

The Shortest Path with Intelligent Algorithm

Surachai Panich

Srinakharinwirot University, 114 Sukhumvit 23, Bangkok 10110, Thailand

Abstract: Problem statement: Path planning algorithms need to be developed and implemented in a suitable manner to give better understanding about the intelligent system and also stimulates technological supply to enormous demands in an intelligent vehicle industry. **Approach:** This study concerned with intelligent path planning using A* search algorithm. **Results:** This study introduced intelligent path planning with A* search algorithm, which use to generate the most efficient path to goal. The algorithm was tested on simulator. **Conclusion:** This study is an implementation of a path planning for an intelligent path planning. The implementations are tested and verified with the simulation software. The path planning algorithms were selected for the implementation and to verify them.

Key words: A search algorithm, path planning, Hough transformation

INTRODUCTION

For a number of differences navigational control strategies have been adopted by various parties. Consider a highly automated factory where mobile robots pick up parts and deliver them to assembly robots. The robots must find their way to parts, pick them up and move to the assembly stations. An automatic motion planner will relieve the operators from this tedious job and enable them to control at a supervisory level. In turn, this increases efficiency by eliminating human errors. All these motions have to be executed without colliding with Objects and other robots. Without a motion planner for the robots and arms, human operators have to constantly specify the motions. The need for collision avoidance and efficient motions leads to the problem of motion planning. There are two approaches for the path planning, global path planning and local path planning (Czarnecki and Rotten, 1995). In real world applications of mobile robots both global and local maps are used. Global maps should decompose hierarchically into local maps to allow easy movement between the two. One obvious advantage of this structure is a reduction in the use of local memory without loss of detail because the computer in the robot has to store only the global map and a detail local map for the area it is in not a detailed map of its whole universe. This assumes that local maps can be retrieved from secondary storage, or a host computer, at any time.

Global path planning: Thy roadmap method consists of capturing the connectivity of the robot s free space.

In a network of one dimensional curve, called the roadmap, a sub path combined in the roadmap and a sub path connector the roadmap to the goal configuration (Sotelo and Rodriguez, 2004). A visibility roadmap is obtained by generating all line segments between pairs of obstacle region vertices. Any line segment that lies entirely in the free space is added to the roadmap. When a path planning is given the initial position and goal position are also treated as vertices. This generates a connectivity graph that can be searched for a solution. The sweep-line principle can be applied to yield a more efficient algorithm. This diagram is the set of ah the free configurations whose minimal distance to the obstacle region. The advantage of this diagram is that it yields free paths which tend to maximize the clearance between the robot and the obstacles. Cell Decomposition methods are perhaps the motion planning methods which have been the most extensively studied so far. They consist of decomposing the robot's free space into simple region called cell, such that a path between any two configurations in a cell can be easily generated. A non directed graph representing the adjacency between the cells in then constructed and searched. This graph is called the connectivity graph. Its node is the cell extracted from the free space and two nodes are connected by a link if and only if the two corresponding cells are adjacent. The outcome of the search is a sequence of cells caned a channel.

Local path planning: The reduction of the calculation time for planning is decisive for the controlling concept (Everett, 1995). Between the planning steps a

discrepancy occurs between the present obstacle knowledge and last calculated global plan. These discrepancies have to be bridged by the local planner until the new global plan is calculated. This leads to a different consideration of the online in comparison to the off-line concept. Demand can be made for the local planner depending on the discrepancy. This study assumes that the discrepancy is relatively small. This leads to the demand that the local planner should not have a complex navigation strategy when having a fast execution in mind.

MATERIALS AND METHODS

This study concerns with intelligent path planning using A* search algorithm.

A* search algorithm: The environment has to be represented in a grid, the size and resolution of which are usually determined by speed and memory limitations. Each goal square is given a distance transform value of 0. The minimum cost of moving to each square from the minimum of its neighbors is calculated in a series of forward and reverse raster operations. This algorithm terminates when the values in each square no longer change after a raster operation in each direction. The required path is then a simple gradient descent from the robot's current square to the nearest goal. First the program sets all the cells in the grid to initial values. Goal and obstacle cells have constant values throughout the computation.

Table 1: The solution X to Y

Path 1	From	X→A = 5 m
	From	A→S = 2 m
	From	S→Y = 3 m
Path 2	From	X→A = 5 m
	From	A→B = 4 m
	From	B→Y = 6 m
Path 3	From	X→B = 3 m
	From	B→A = 4 m
	From	A→S = 2 m
	From	S→Y = 3 m
Path 4	From	X→B = 3 m
	From	B→Y = 6 m

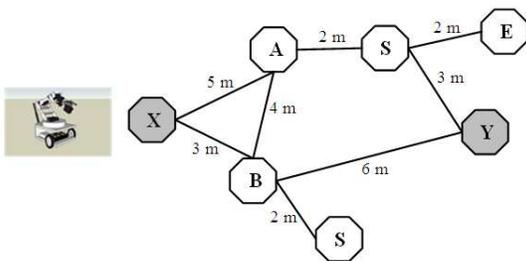


Fig. 1: Robot path planning with A* search algorithm

Since the cost of traveling from the goal to the goal is zero, the goal cell is always set to 0.0. Since traveling through obstacles is not normally desire. We discourage this by setting the cost at obstacle cells to an arbitrarily large value called Big-cost. After setting the cells to initial values, the program repeatedly scans through the grid looking for cells it can reset to lower values. The lower cost is computed by looking at neighboring cells and using an estimate of travel cost from the adjacent cells to the current cell. For orthogonal adjacent cells the estimate is 1.0. This is just the distance from the center of one cell to the center of the next, assuming each cell measures one unit on a side Note that if the cost estimates were multiplied by the resolution of the grid the values at each cell would reflect the true distance to the goal. The paths used in this system are called virtual paths. Most path planner abstracts the search space to a graph of possible paths. This graph is then search to find the shortest path. This approach arises naturally in a learning environment, where a robot may have traversed several paths to map the world.

From Fig. 1 the solution X to Y has cost is shown in Table 1.

So the shortest cost from X-Y is path number 4, which is 3+6 (X→B→Y) equal 9 m (Table 1).

RESULTS

This study introduces intelligent path planning with A* search algorithm, which use to generate the most efficient path to goal (Staugaard, 1987; Stentz, 1994; Arikan *et al.*, 2001; Dai, 2005). The algorithm is tested on simulator, which is developed from Gulliveig's source code. The start point is green circle, red point for obstacles and blue point for goal point. The path used to test before algorithm start is shown in Fig. 2. The result after using algorithm is shown in Fig. 3. The shortest path from start to goal point is shown in green line.

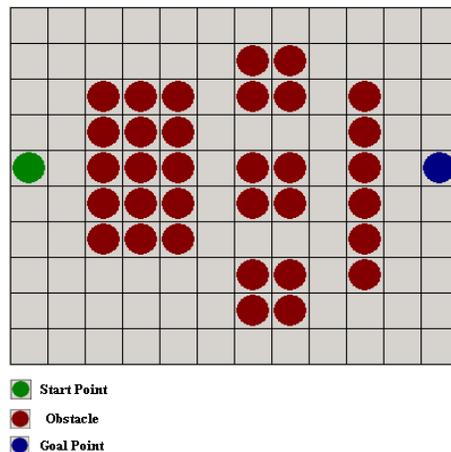


Fig. 2: Path used to test before algorithm start

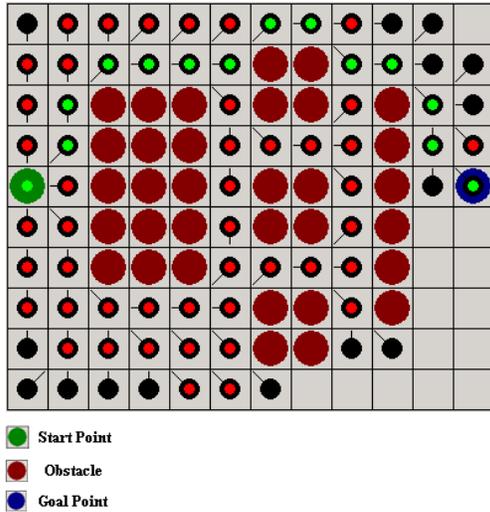


Fig. 3: Result after using algorithm

DISCUSSION

The implementation of local path planning and obstacle avoidance will increase the degree of autonomous of the vehicle and more algorithms like dynamic version of D-Star can be implemented.

CONCLUSION

This study is an implementation of a path planning for an intelligent path planning. The implementations are tested and verified with the simulation software. The path planning algorithms were selected for the implementation and to verify them. From the result, A*Star path planning algorithm performed well, in term of performance and give a correct result. The algorithm is also give the correct result but with more execution time. Both algorithms are suited for path planning navigation task. The output of the simulator gives guide information for vehicle's motion.

ACKNOWLEDGEMENT

This research from Measurement and Mobile Robot Laboratory (M and M-LAB) was supported by Faculty of Engineering, Srinakharinwirot University.

REFERENCES

- Arikan, O., S. Cheney and D.A. Forsyth, 2001. Efficient multi-agent path planning. Proceedings of the Eurographic Workshop on Computer, Sept. 2-3, ACM Press, Manchester, UK., pp: 151-162.
- Czarnecki, C.A. and T.W. Rotten, 1995. Evolutionary motion planning for mobile robots. Proceedings of the ECML Workshop on Intelligent Robotics, (IR'95), Greece, pp: 25-29.
- Dai, L., 2005. Fast shortest path algorithm for road network and implementation. Carleton University. http://www.citeulike.org/user/sven_luzar/article/2967935
- Everett, H.R., 1995. Sensors fur mobile Robot Theory and Application. 1st Edn., AK Peters Ltd., Wellesley, Massachusetts, ISBN: 10: 1568810482, pp: 528.
- Sotelo, M.A. and F.J. Rodriguez, 2004. A color vision-based lane tracking system for autonomous driving on unmarked roads. *Autonom. Robots*, 16: 95-116.
- Staugaard, Jr., A.C., 1987. Robotics and AI: An introduction to Applied machine Intelligence. 1st Edn., Prentice Hall, Inc., Englewood Cliffs, NJ., ISBN: 10: 0137822693, pp: 373.
- Stentz, A., 1994. Optimal and efficient path planning for partially-known environments. Proceedings of the IEEE International Conference on Robotics and Automation, May, 1994, Carnegie Mellon Robotics Institute, USA., pp: 3310-3317.