International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

# Selection of Shortest Path & Validation for Motion of Multiple Robots

**K Prasanna Lakshmi**

***Abstract : When a robot moves from the initial position to the desired position, one or more obstacles having curved periphery are likely to be encountered. When the robot encounters any static obstacle, to avoid the collision, it deviates its path. When the divergence occurs, the robot has more than one direction to reach its destination. Using recursive algorithm, the shortest path is taken from all these alternative paths. The efficiency of the path obtained from alternate paths and decision tree values, validation is done using chi square test. Multiple robots are used in medical and industrial fields where they are likely to move between people and static obstacles when distributing food services, medical supplies, transporting goods from one location to another and many such applications.***

***Keywords: Alternative path, Chi square test, Collision, Curved obstacles, Recursive algorithm, Shortest path, Validation.***

## I. INTRODUCTION

Due to human intervention, the precision, the punctuality, the pressure, the targets have been at stake over the past decade. Yet as many of the tasks are replaced by the robots, the goals are likely to reach quickly. Replacing a human being with a robot is desperate to have a human's intellect. So many complex tasks become much simpler with efficient programming and intelligent interface between the programming and the robot. In this paper, a small attempt is made to program the motion of multiple robots while simultaneously moving n number of robots. Only 7 robots are considered for illustration in this article, which can be expanded to n robots in addition. If the robots are required to move from one position to another, they may have to encounter the human beings and static obstacles in their path. The robot must ensure that it completes its mission without colliding with any of the obstacle in its path that are stationary or moving. If it faces any obstacle, it must deviate from its path and move along the prescribed path. The recommended path is the generated path by following the recursive algorithm.

If the robot deviates from its course, it may have a lot of alternate paths to travel to its destination. The shortest path must be calculated and prescribed for the robot to travel among all the available alternative paths. The shortest path must be calculated and prescribed for the robot to move among all the available alternative paths. The function assigned to it is accomplished once the robot moves in the shortest path, which is free from collision. Whatever method is followed to determine the alternative path, the shortest path with curved boundaries and curved obstacles, a measure of consistency is required to validate the data used. The testing tool used in this paper is the chi-square check, which validates the data used to find the shortest path using recursive algorithm[5] & decision tree efficiencies.

Earlier research on path planning are referred for implementing the simultaneous motion of multiple robots with curved obstacles. [1] is a graphical user interface developed based on MATLAB GUI to find the optimal path between two points so that robots can follow the shortest path and consume the least energy, using image processing with a single ceiling camera. [2] deals with a path planning problems where turning point algorithm handles two different objectives which are the path safety and path lengths. [3] presents an algorithm based on the concept of reciprocal orientation that guarantees smooth trajectories and collision free paths.

Manuscript published on November 30, 2019.
* Correspondence Author
**Dr. K Prasanna Lakshmi***, Associate Professor in Dept. of Mechanical Engineering. Presently working at JNTUH College of Engineering Manthani. prasannakaujala@gmail.com



## II. METHODOLOGY

Rectangular workspace was considered with rectangular obstacles. As obstacles increases, the scope of alternate paths also increases. Since the robot is made to complete the assigned task in minimum time, the robot will follow the minimum distances amongst all the alternate paths, assuming the same velocity under any condition. The generation of alternate paths is following the recursive algorithm[4].
The total distance of any path is given by the equation:
Total Distance (D) of the of the AP=
$Rn_i$(i=1 to n) =

$$\sum_{p=1}^{m}\{dij_{p1}\, z[path(p-1)] + dij_{p2}\, z(p-2)]\} \quad - \quad \text{Eq- 1}$$

Where p is the path and m is the number of obstacles, RNi is the indexed number assigned to the robot, dij is the distances of alternate paths at various conjunctions, z is the function defined below.

**Z - Function**

Z (x) = 1 If x = 0

= 0 If x ≠ 0

**Path Array (3 x 2)**

**Table I : Table depicting the alternate paths for every added obstacle.**

| Path No. | 1 | 2 |
|---|---|---|
| Obstacle (m) | | |
| 1 | 1 | 2 |
| 2 | 1 | 2 |
| 3 | 1 | 2 |

Various alternate paths are generated based on this algorithm by undergoing various iterations for a robot to move from its original location to final destination for varying paths. According to number of obstacles, every obstacles gives rise to two alternate paths as depicted in the path array in Table I. The figure1 shows the various alternate paths for a rectangular obstacle obtained from recursive algorithm with rectangular boundary.

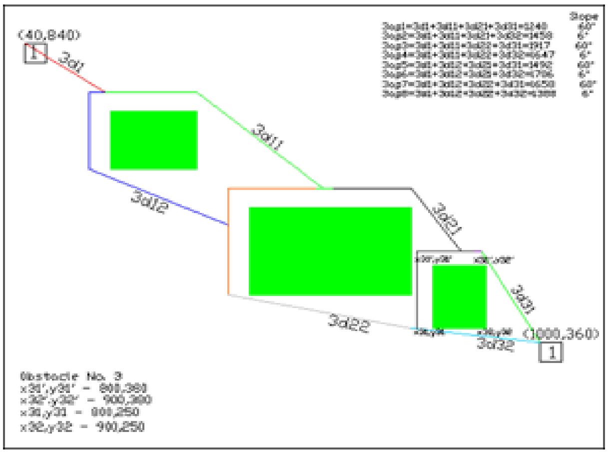


**Fig 1: Figure shows alternate paths for every obstacle**.

## III. RESULTS & ANALYSIS

### A. *Calculation Of Distances*

The distances of the robot for all the alternate paths are calculated taking the initial and intermediate positions. The shortest distance is the straight line path when the obstacle doesn't exist. The following figure shows the straight line path without any obstacle.

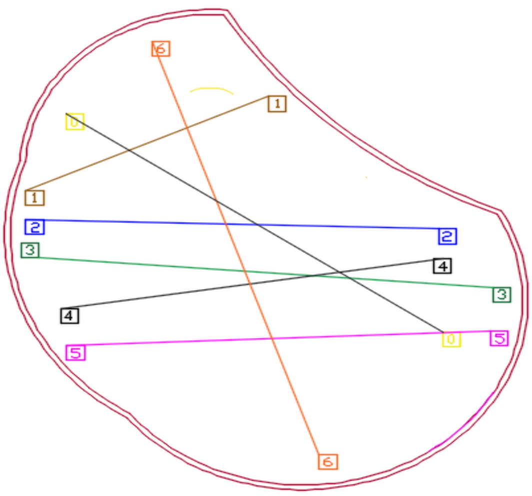

**Fig 2 : Figure shows the paths for 7 robots without any obstacle.**

The intermediate position may come across the curved path. The equation of the curved path [5] can fetch the distance moved on it. Likewise all the small distance in the intermediate positions are taken and added, resulting in the minimum distance for that particular robot. Likewise all the distances of the other robots are also calculated.

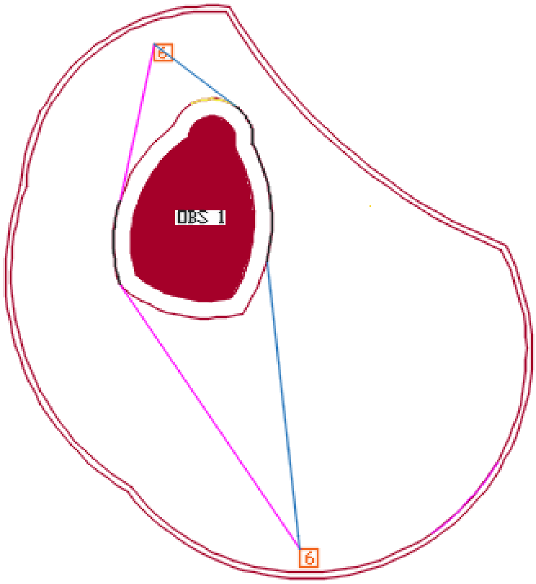


**Fig 3 : Figure shows two alternate paths for One obstacle.**

While the robots are moving, the distance may not change but the time may vary if the robots have to trace the same path. In such a case, the priority is assigned to the robot having low index number, and the lower indexed numbered robot will move first and then with the predefined gap, the second robot will move on the same path and thus reaching their destinations. For an illustration, Robot no. 6 is taken for finding the alternate paths according to the path array. The following figure shows the increase in alternate paths as the obstacle size increases.

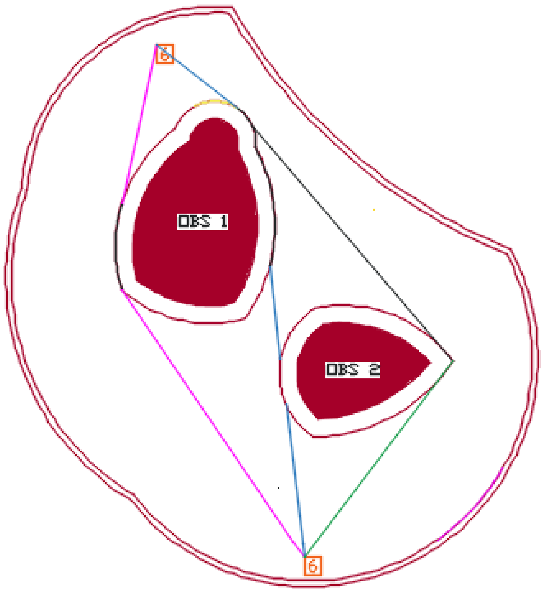


**Fig 4 : The figure shows the 3 alternate paths for 2 robots, since the 1st alternate path doesn't find any obstacle.**

The figure depicts the increase in alternate paths in geometric progression form. Robot No. 6,, if encountered with 3 obstacles, will have total of 8 paths.

Retrieval Number: D9023118419/2019©BEIESP
DOI:10.35940/ijrte.D9023.118419
Journal Website: www.ijrte.org



Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

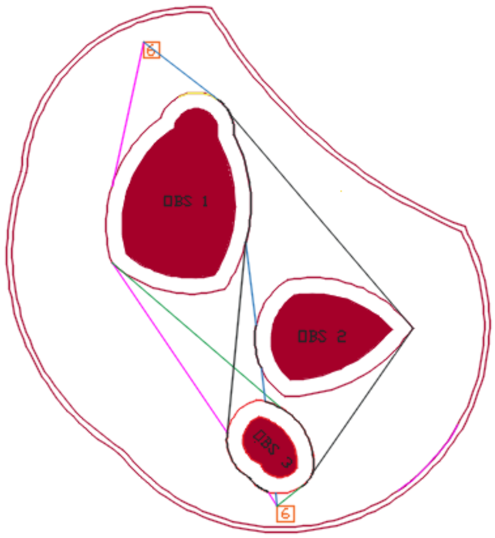


**Fig 5 : The figure shows total possible alternate paths for three obstacles.**

The shortest of these distances are measured and the path is traced in this shortest path. So a clear relation can be mentioned between the number of obstacles and number of paths as shown the table II.

| N (no. of Obstacles) | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| P - No. of Paths | 1 | 2 | 4 | 8 | 16 |

**Table II : It depicts the pattern of geometric progression for the increase in the number of obstacles.**

According to the path array and according to the distance measured of all the alternate paths. The shortest path is calculated from Eq 1. The Figure 6 shows all such shortest path on to which the robot traces its journey to reach their destinations.

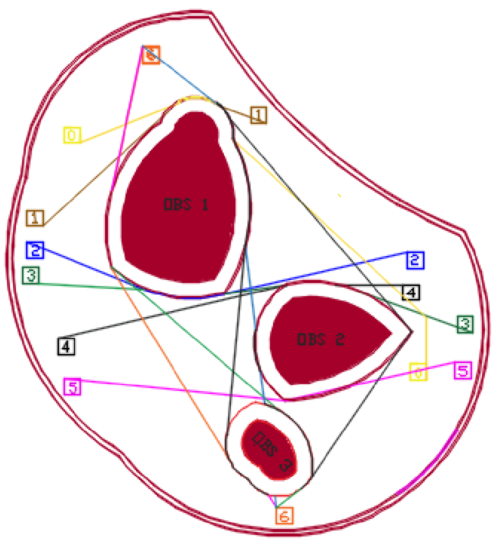


**Fig 6 : The figures shows the shortest distance for all the 6 robots.**

The figure 6 shows the path(s) to be traced by all the robots, after finding the shortest distance from the recursive algorithm.

### *B. Decision Tree Analysis*

When a robot is moving and it comes across a static obstacle, it will have two paths to travel to its destination deviating its path from the 1st obstacle. If the first path again encounters a 2nd static obstacle, it will also have two alternate paths as depicted in path array. So every obstacle prompts the robot to have two alternate paths. Now the decision tree [6], takes the path which has least distance in every deviation. All such distances are measured and is taken as the final path to be traced by the robot. The following figure shows the decision tree analysis for every such added robots for varying paths.

The decision tree distances are measured as deviation from the straight line path. These distances are preplanned and doesn't give the shortest path but computation effort is very less. So a comparison is done between the decision tree efficiency and maximum efficiency found by the alternate path.

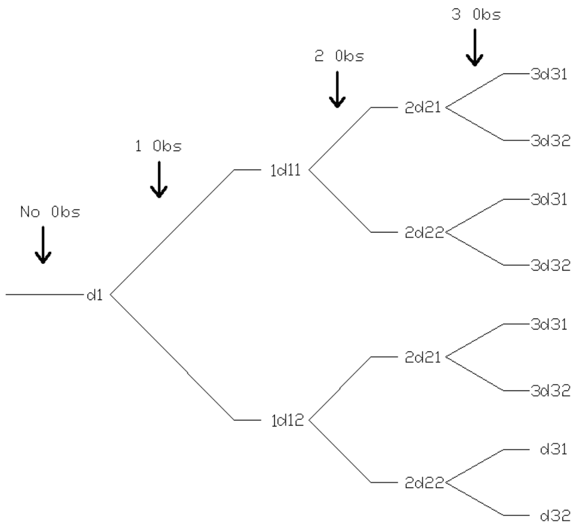


**Fig 7 : Decision tree analysis**

### *C. Validation*

Theoretical formulas based data, experimented data, survey data and all such data needs to be checked for correctness, meaningfulness and truthfulness.

This paper checks such a validation of the data used in the process undergone for motion of multiple robots among curved obstacles and curved boundaries. Validation of the data acquired is done by using chi square test[7]. The data is measured for all the 6 robots. Based on the results obtained for all the robots with three obstacles, the statistical analysis is carried-out considering the maximum efficiencies obtained from the research work and maximum efficiencies obtained from Decision Tree analysis.

The values of $\chi 2$ obtained is 3.74 as shown in the TableIII, which is less than the table value of 95% confidence level(p=0.05) of 3.84 for one degrees of freedom, in which case null hypothesis is accepted and hence the validation.

*Retrieval Number: D9023118419/2019©BEIESP*
*DOI:10.35940/ijrte.D9023.118419*
*Journal Website: www.ijrte.org*



*Published By:*
*Blue Eyes Intelligence Engineering*
*& Sciences Publication*

| Robot No. | Maximum efficiency obtained with three obstacle | | | | |
|---|---|---|---|---|---|
| | Simulated Value (O) | Decision tree values (E) | (O-E) | $(O-E)^2$ | $(\chi^2) = \frac{(O-E)^2}{E}$ |
| 0 | 73.33 | 88.05 | -14.72 | 216.6 | 2.45 |
| 1 | 89.41 | 90.85 | -1.44 | 2.07 | 0.022 |
| 2 | 95.93 | 99.31 | -3.38 | 11.42 | 0.115 |
| 3 | 96.29 | 97.5 | -1.21 | 1.46 | 0.015 |
| 4 | 98.18 | 99.25 | -1.07 | 1.14 | 0.011 |
| 5 | 92.36 | 97.65 | -5.29 | 27.98 | 0.286 |
| 6 | 92.5 | 101.75 | -9.25 | 85.56 | 0.841 |
| | | | | $(\chi^2)$ =Total = 3.74 | |

**Table III : Chi-Square test values for all the 7 robots.**

## IV. CONCLUSION

The major role is of the path traced by robots when they are moving simultaneously. In this paper, the shortest path is found out by finding all the alternate path distances by using recursive algorithm. The algorithm gives rise to alternate paths by the path array. Maximum efficiency of all the robots are obtained as shown under simulated(Observed) values in Table 3. Once all the alternate path distances are calculated, the shortest distance is picked as shown in Fig 6.

Maximum efficiency of all the robots are obtained as shown under simulated(Observed) values in Table 3. The highest efficiency is of the robot No 4 found out to be 98.18%. A generalized layout for the automated working environment, for the multi Robot systems with their origin and target position are designed choosing the target positions. The Z-Function is incorporated considering 3 x 2 Array for considering all the alternate paths. A Robot can overtake the obstacle by choosing the best alternative path, which is the shortest distance using recursive algorithm. Such an alternative path is considered using the Z-Function, incorporated considering 3 x 2 Array. Robot No. 6's paths are traced in Figure 3,4 & 5 which depicts how the alternate paths are formed. The data acquired is validated using chi square test shown in Table3 and is found out that the null hypotheses is accepted.

This paper has all the information regarding the paths without obstacle, with obstacles, its path and then finding the shortest path. But the velocities of the robots is taken almost constant. If the robot is moving in a space where in the nearby zone is free of obstacle, the velocity of the same robot can be increased. Further study of increase in velocity wherever possible can be appreciated. The recursive algorithm proposed in this paper is useful for the designer to design the motion of multiple robots simultaneously in working environment.

## REFERENCES


1. Z Batik Garip, D Karayel, SS Ozkan & G Atali, "Path planning for multiple robots using A* Algorithm", 3rd International Conference on Computational and Experimental Science and Engineering, 2016.
2. Imen Hassani, Imen Maalej & Chokri Rekik, " Robot Path planning with avoiding obstacles in known environment using free segments and turning points algorithm", Mathematical problems in engineering, doi 10.1155/2018/2163278, Jun 2018.
3. Abduladhem A Ali, Abdulmuttaib T Rashid, Mattia Frasca, Luigi Fortuna, "An algorithm for multi robot collision free navigation based on shortest distance", Robotics and Autonomous systems, Vol 75, pp 119-128, Jan 2016.
4. K Prasanna Lakshmi, "Collision avoidance recursive algorithm", International journal of Mechanical engineering and robotics research, Vol 3, Issue 1, Jan 2014.
5. Danny Z Chen, Haitao Wang, "Computing shortest Paths among Curved Obstacles in the plane", in ACM Transactions on Algorithms (TALG) , Vol 11 Issue 4, June 2015.
6. Dr. K Prasanna Lakshmi, "Validation of alternate paths for motion of multiple robots", International Journal of Interdisciplinary Research and Innovations, Vol 7, Issue 2, pp-449-451, 2019.
7. Rakesh Rana and Richa Singhal, "Chi Square Test and its application in Hypothesis Testing', Journal of the practice of Cardio Vascular sciences, 2015, Vol 1, Issue 1, Page 69-71


## AUTHORS PROFILE

Dr. K Prasanna Lakshmi, Associate Professor in Dept. of Mechanical Engineering. Presently working at JNTUH College of Engineering Manthani. Completed B.Tech (Mechanical Engineering) from NBKRIST in 1997, SVU, M.Tech (Production Engineering) from SVU in 2000, PhD(Motion Planning of Multiple Robots) from JNTU Anantapur in 2011. Publications: 2 National Journals, 3 National Conference, 11 International Journals, 6 International Conferences. Awarded best paper presentation in Nov 2009 in Science Congress Presently guiding 6 PhD scholars. MISTE, AMIE, MIIPE are the existing memberships. Acting as Advisor for IEISC(Institute of Engineers India, Students Chapter, Ramagundam Local Centre).