Multiple Mobile Robot Path Planning in 3D Environment Using Tangent Tree Algorithm

Ahmed Fahem Albaghdadi*1,2, Abduladhem Abdulkareem Ali2

1 Asst Lecturer, Najaf Technical Institute, Al-Furat Al-Awsat Technical University
31001 Al-Najaf, Iraq.
2 Faculty of Engineering, University of Basrah, Basrah, Iraq
*Email: ahmedfa91m@gmail.com

Abstract. Due to the effective applications of multi-agent systems, there is an importance to develop efficient methods to manage these systems. One of the most important methods is path planning. This paper proposes a new algorithm for the path planning of multiple mobile robots in 3D environment named Tangent Tree Algorithm. It based on the assumption that the robots have spherical shapes and can move in a 3D space like drones, other moving robots considered as obstacles for the robot. The robot path performed using two modes: In the first mode, the robot moves in a straight line to the tangent point of assumption sphere. In the second mode, it moves around the assumption sphere. The assumption sphere locates at the calculated collision 3D point with the other robots. A number of possible paths constricted using these modes. A Tangent tree algorithm for multi-robot formed and the minimum path is determined based on minimizing this tree. Simulation results indicate that the new algorithm is superior to other algorithms in terms of both the path cost and computation required. Furthermore, this algorithm does not change the velocity of the robots; it only depends on the orientation.

Key words: 3D path planning, Multi-robot, search tree, Collision free path, and Tangent Tree Algorithm

1. Introduction

Recently, and due to the importance of the multi-agent system and their increasing applications, Researchers interest also increases. One of the main problems in the area is motion planning. Each robot must complete its mission without collision with any other robot. This problem not only robotics. There are many other sides like video gaming, architecture design, and traffic engineering. Many kinds of research study the path planning of multi-robot. The vast majority of them deal with this problem only in the 2D space. Some of these researches are summarized as follows.

[1] Proposed an algorithm that deals with multiple robot navigation named: reciprocal orientation algorithm. The reciprocal orientation concept grantsee smooth path. They use differential drive robot to simulate their work in different scenarios. The main idea of this algorithm is keeping all robots in same velocities without the need to change their velocities. There is only change in direction to avoid collision. They depend only on change angular velocities. They solve the deadlock problem by this principle "change angular velocity based on change of distance between the robots.

The shortest distance algorithm [2] is an improvement of previous work. The main modification is that robots rotate to the shortest side from their location to avoid collision. This work is also limited to two dimensions space and uses differential drive robot to simulate the algorithm.
Another study made some experiment about complexity in a population of Artemia depending on its response to a spot of light. And drive a mathematical model of the motion [3]. This model used to drive a mathematical model of a guided Artemia swarm dynamics in an attempt to implement in a multi-robot system. This model based on the principle of potential field algorithm. They use spot of light to control a dynamic of a swarm [4]. The derived model can implemented to control a dynamic robot swarm in two-dimensional space.

Path planning is very important and essential for advance robots in modern life especially in 3D space such applications include multi-rotor aerial vehicles. Lopez et al [5] proposed an algorithm of path planning for multirotor aerial vehicles in dynamic environment. The probabilistic graph used to sample admissible free space without obstacles. A* algorithm with an artificial field map as cost function is utilized to explore the generated probabilistic graph.

The importance of energy is studied by Araki et al, [6]. They made a path-planning algorithm to minimize path and energy consumption.

3D path planning in automated aerial videography has been studied in [7]. They test their algorithm by multiple challenging shots, also multiple drones and actors at same time.

Jose and Pratihar [8] studied path planning and task allocation of centralized multi robots. They use three robots in the common workspace in their tests. Task allocation achieved by using a genetic algorithm (GA). Path planning is performed by A* algorithm. There is another usage of genetic algorithm in this work. They used the GA to assign optimal number of tasks to each robot.

An algorithm for multi-robot Path Planning on Graphs is proposed in [9]. The algorithm used to find the optimal path based on four objective functions. Weighted sum principle employed to combine different objective functions in one equation. The algorithm depends on minimization of maximum travelled distance of each robot, last arrival time, total distance and total arrival time.

3D path planning in automated aerial vehicles has been studied in [10]. It deals with multi mobile robot in swarm. The authors develop Particle Swarm Optimization to be Angle Encoded Particle Swarm Optimization. The developed algorithm is faster than PSO. It used to find a collision-free path for multi-robot in 3D and maintain shape of the UAV formation as well as minimize path and maintain altitude.

2. Tangent Tree Algorithm

Advance technology in engineering science needs many aspects of algorithms to Compatible the new ideas. One of the most attractive ideas is working of multi smart robots in same workspace especially in 3D environment. Path planning in 3D environment is very important to provide a collision-free path and make it short as possible. There are some algorithms of path planning in 3D environment as described above. In our algorithm, we suppose each robot has spherical shape and other information like radius, source point, and target point are given. This algorithm produces a smooth path for each robot consists of arcs and straight lines. These two parts of each path used multiple times to avoid each obstacle or another robot. A number of arcs and straight lines are proportional to the number of obstacle face the robot. This algorithm based on the behavior of human. If we imagine a human behavior when he wants to avoid any obstacle like building. Normally he will go to the nearest edge of the obstacle by direct path then move around the building then go directly to the target when he starts seeing it. To apply this behavior and learn it to the robots, we need complex equations. This algorithm can summarized according to the following assumption and procedure.

Assumptions are:
- All robot has a spherical shape
- The radius of each robot is given
- Source and target points in the 3D space of each robot are given.

The procedure of tangent tree algorithm in 3D environment with existence of other robots are:

Step 1: Sense and obtain information. The radius and 3D position of each robot are represented by r, px, py, and Pz, respectively, while the target position is represented by (fx, fy, and fz).

Step 2: construct a straight-line path between source and goal points for each robot. This equation given as:
\[ x = (f x - px)t + px \]
\[ y = (f y - py)t + py \]  
\[ z = (f z - pz)t + pz \]

Step 3: Calculate collision points based on the given velocity and direction vector for each robot.

Step 4: Construct a dummy fix spherical obstacle centered at each collision point with a radius proportional with size and number of collision robots. The center and radius of supposed obstacle are Sx, Sy, Sz, and R respectively.

Step 5: Determine n paths around the dummy obstacle for each collide robot according to the following sub-steps.

1- Depending on the source point and obstacle information (position and radius), n points around the dummy obstacle are determined. When the robot moves in a straight line to these points, it will be in touch with this obstacle.

When n approaches infinity, a 3D circle-shaped, as illustrated in Figure 1.

![Figure 1](image)

Figure 1. Visible circle by the robot around the obstacle when n=inf.

2- Select n points from the circle. Firstly, a circle determined by the intersection of a 2D plane with the supposed obstacle after increasing its radius by r to compensate for the robot diameter. The result of the intersection is a circle in 3D according to equations 2

\[(X - Sx)^2 + (Y - Sy)^2 + (Z - Sz)^2 = (R + r)^2 \]
\[(2Px - 2Sx)X + (2Py - 2Sy) Y + (2Pz - 2Sz) Z + d = 0 \]

![Figure 2](image)

Figure 2. Possible paths to the n selected point around the obstacle
3- Derive n equations of straight lines between the target point and tangent of obstacle previously determined. Each line has two points: the target point and point belong to the n points selected from the determined circle around the obstacle seen from the target side in equations 3.

\[
(X - Sx)^2 + (Y - Sy)^2 + (Z - Sz)^2 = (R + r)^2
\]
\[
(2Fx - 2Sx)X + (2Fy - 2Sy)Y + (2Fz - 2Sz)Z + d = 0
\]

……………………………………………………………………………….(3)

4- Draw curves between the points of each obstacle sides, according to the short segment of the great circle principle, as illustrated in Figure 3. The great 3D circle can be determined by intersecting a plane with the obstacle after increasing its radius by r. We have center point and two points that we wish to connect, so by using a cross product between vectors from the origin point and two points that we wish to connect, we determine a perpendicular vector in order to identify the plane. The resulting intersection is a great circle that includes two points that we wish to connect. These two points divide the great circle into two parts with different lengths. The shortest part is what we need. Equation 4 describes a great circle by intersecting sphere and plane. Where P and Q are the points we wish to connect and C is the center of the obstacle.

\[
(X - Cx)^2 + (Y - Cy)^2 + (Z - Cz)^2 = (R + r)^2
\]
\[
((Py - Cy)(Qz - Cz) - (Pz - Cz)(Qy - Cy))(X - Cx) -
\]
\[
((Pz - Cz)(Qx - Cx) - (Pz - Cz)(Qy - Cy))(Y - Cy) +
\]
\[
((Pz - Cz)(Qy - Cy) - (Py - Cy)(Qx - Cx))(Z - Cz) = 0
\]

……………………………………………………………………………….(4)

Step 6: Chose a random path of the calculated paths. The random choice to avoid dead look in some special scenarios.

Step 7: Check about any collision. If it exists repeat from step three after checking the following two cases

Case 1: if collision happen before point calculated in substep 3 of step 5 then just go to step 3
Case 2: if a collision happens after point calculated in substep 3 of step 5 then replace source point by point that calculated in substep 3 and go to step 3.

3. Simulation Results

The simulation implemented using Matlab 2018 on a core i7 2.67 GHZ computer with Windows 10. The simulation tested for multiple cases with different numbers of robots, and various sizes of robots. Some of the simulation results in Fig. 4,5,6 and 7. In Fig. 4, there are two robots move in collide path. Tangent tree algorithm used to find a collision-free path for each robot as illustrated in Fig.4. The same idea used in Fig.5, 6 and 7 to find a collision-free path for each robot. The direction of
moving of each robot illustrated with arrows on the figures. The Tangent tree algorithm compared with Potential Field Algorithm. Fig.8 shows the path lengths for both proposed tangent tree algorithm and potential field algorithm. Though the path difference shows little improvements in the proposed algorithm, the proposed algorithm has advantage that is more important. The new algorithm has a smooth path compared with potential field. To show this advantage one of the robot path for robots in Fig.7 is shown in Fig.9 for both potential field and tangent tree algorithms. See as from this Fig.9. The new algorithm has a much smoother path, which makes the robot movement control much easier in implementation.

4. Conclusions

This paper presents a new algorithm of a 3D path planning with existence of other dynamic robots. This algorithm named Tangent tree algorithm. The robots used in this algorithm assumed to have a spherical shape. This does not mean that the developed algorithm cannot deal with robots of other shapes. It is possible to represent any shape of robot by a sphere with diameter equal or more than longest diameter of real robot. The robot paths have two forms. The first form is straight line began from the current location of the robot, the end of the line is the edge of obstacle or another robot according to the calculations previously explained. The second form is a curve to avoid an obstacle or other robots. In this mode, the great circle principle is used. This form determines a sub-optimal path for all moving robots. This algorithm can find sub minimal path easily and quickly with short computational time. The resulting path is smooth regardless collision angles. When compare this property with other algorithms like potential field algorithm in [11], it is easy to find much difference in the smoothness in paths between this algorithm and the potential field algorithm.
5. References

[1] A. T. Rashid, A. A. Ali, M. Frasca, and L. Fortuna, “Multi-robot collision-free navigation based on reciprocal orientation,” Rob. Auton. Syst., vol. 60, no. 10, pp. 1221–1230, 2012.

[2] A. A. Ali, A. T. Rashid, M. Frasca, and L. Fortuna, “An algorithm for multi-robot collision-free navigation based on shortest distance,” Rob. Auton. Syst., vol. 75, pp. 119–128, 2016.

[3] A. A. Ali, L. Fortuna, M. Frasca, M. T. Rashid, and M. G. Xibilia, “Complexity in a population of Artemia,” Chaos, Solitons and Fractals, vol. 44, no. 4–5, pp. 306–316, 2011.

[4] M. T. Rashid, M. Frasca, A. A. Ali, R. S. Ali, L. Fortuna, and M. G. Xibilia, “Artemia swarm dynamics and path tracking.” Nonlinear Dyn., vol. 68, no. 4, pp. 555–563, 2012.

[5] J. L. Sanchez-Lopez, M. Wang, M. A. Olivares-Mendez, M. Molina, and H. Voos, “A Real-Time 3D Path Planning Solution for Collision-Free Navigation of Multirotor Aerial Robots in Dynamic Environments,” J. Intell. Robot. Syst. Theory Appl., pp. 1–21, 2018.

[6] B. Araki, J. Strang, S. Pohorecky, C. Qiu, T. Naegeli, and D. Rus, “Multi-robot path planning for a swarm of robots that can both fly and drive,” Proc. - IEEE Int. Conf. Robot. Autom., pp. 5575–5582, 2017.

[7] T. Nägeli, L. Meier, A. Domahidi, J. Alonso-Mora, and O. Hilliges, “Real-time planning for automated multi-view drone cinematography,” ACM Trans. Graph., vol. 36, no. 4, pp. 1–10, 2017.

[8] K. Jose and D. K. Pratihar, “Task allocation and collision-free path planning of centralized multi-robots system for industrial plant inspection using heuristic methods,” Rob. Auton. Syst., vol. 80, pp. 34–42, 2016.

[9] J. Yu and S. M. LaValle, “Optimal Multi-Robot Path Planning on Graphs: Structure and Computational Complexity,” vol. 32, no. 5, pp. 1163–1177, 2015.

[10] V. T. Hoang, M. D. Phung, T. H. Dinh, Q. P. Ha, and R. O. Dec, “Angle-Encoded Swarm Optimization for UAV Formation Path Planning.” Dec 2018

[11] Ahmed Fahem Albaghdadi, Abduladhem Abdulkareem Ali, March 2019. 3D Path planning of fixed and mobile environments using potential field algorithm with Genetic algorithm 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON); pp. 115-119. Jaipur, India.