

Review on path planning algorithm for unmanned aerial vehicles

Nurul Saliha Amani Ibrahim, Faiz Asraf Saparudin

Department of Electrical Engineering Technology, Faculty of Engineering Technology, University Tun Hussein Onn Malaysia, Malaysia

Article Info

Article history:

Received Feb 24, 2021

Revised Aug 30, 2021

Accepted Sep 6, 2021

Keywords:

Algorithm

Drone

Path planning

Unmanned aerial vehicles

ABSTRACT

The path planning problem has been a crucial topic to be solved in autonomous vehicles. Path planning consists operations to find the route that passes through all of the points of interest in a given area. Several algorithms have been proposed and outlined in the various literature for the path planning of autonomous vehicle especially for unmanned aerial vehicles (UAV). The algorithms are not guaranteed to give full performance in each path planning cases but each one of them has their own specification which makes them suitable in sophisticated situation. This review paper evaluates several possible different path planning approaches of UAVs in terms optimal path, probabilistic completeness and computation time along with their application in specific problems.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Nurul Saliha Amani Ibrahim

Faculty of Engineering Technology

Universiti Tun Hussein Onn Malaysia

Pagoh Higher Education Hub, 84600 Pagoh, Muar, Johor, Malaysia

Email: salihaamani96@gmail.com

1. INTRODUCTION

Over the past decades, unmanned aerial vehicles (UAVs) have increasingly been applied in different areas with wide range of applications, such as communication, surveillance, photogrammetry, disaster management, and structure supervision [1]. Intelligent vehicles such as UAVs have advanced their capabilities for highly and, even fully, automated driving under controlled environments [2]. However, path planning remains one of the primary issues that must be addressed before vehicles can traverse in complex environments independently [3]. Indeed, one of the most difficult problems is generating an efficient path from a given initial destination to a final destination in real time [4]. Thus, multiple algorithms are being introduced and improved in order for it to be able to choose the path that takes less time and that presents less costs to accomplish the intended tasks [5].

– Path planning approach

This review paper evaluates several possible path planning algorithms of UAVs in terms optimal path, probabilistic completeness and computation time along with their application in specific problems. These properties are important in path planning algorithm, when a search algorithm possesses the property of optimality, it guarantees that it will locate the best possible solution. When a search algorithm has the property of probabilistic completeness, it means that the algorithm will return a solution if one is available. Path planning approaches reviewed in this paper can be classified into three categories; Graph search; Sampling-Based; and Biological-Inspired Path Planning as illustrated in Figure 1.

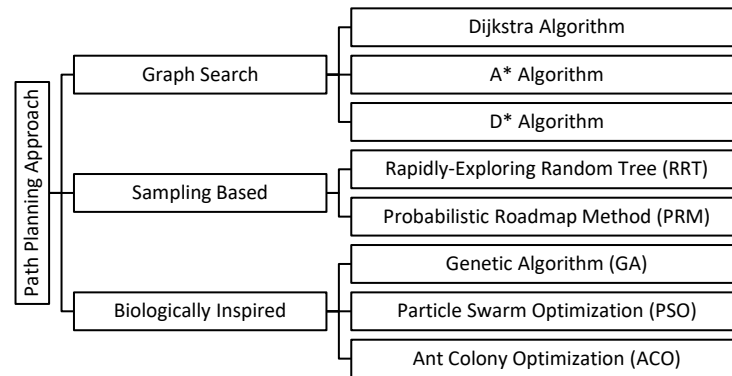


Figure 1. Path planning approach

2. GRAPH SEARCH

In graph search algorithm, the basic idea is to move across from point A to point B in some sort of space. This state space is commonly described as an occupancy grid or lattice which shows where objects are located in the environment. Graph based algorithms generally implemented in sparse and discrete environment [6]. According to [2], some of the automated vehicle's development has been applying this method in their path planning process. There are several graph search algorithms such as A*, Dijkstra's, and D* Algorithm.

2.1. Dijkstra algorithm

Dijkstra algorithm was first introduced in 1956 by Edsger W. Dijkstra [7]. It is based on graph search algorithm that suitable in finding single-source shortest path between one node and every other node in the graph using greedy method [3]. In fact, in [8], Dijkstra algorithm is viewed and presented as Greedy algorithm. The most distinguishing aspect of Dijkstra is that the generated path will start at the center and then extended to the end point in the environment. The formulation of the shortest path between that vertex and every other vertex in the given environment is determined by the vertex (nodes) and edges. The varying weight of the given area will influence the choice of the shortest path in the environment. It is non-heuristic approach. It can provide shortest path but cannot promise optimal result in terms of travel time as in [9]. Although that, results in [10], which compares Dijkstra's, A*, and ant colony optimization (ACO) shows that Dijkstra is still able to give fair performance in real time path planning by having the least run time. By integrating Dijkstra's with Voronoi diagram and visibility algorithm in [11], it has been proven to save up to 21% of energy, which is energy efficient, and the path can keep the vehicle from collision. In [12], Dijkstra's algorithm being improved to only consider the nearest node in a given environment, thus reducing overall time consumption. According to [13], the classical Dijkstra algorithm only capable of finding one shortest path, while skipping over the other paths with the equal distance. Thus, to address this issue, a new enhanced Dijkstra algorithm is presented that able to find all shortest paths. The ideal path with the shortest distance and time is found by adding the running time in the path planning evaluation.

2.2. A* algorithm

A* is an expansion of Dijkstra's graph search algorithm [2] in addressing the shortcoming of Dijkstra's algorithm and adopt the optimum priority search method [14]. A* algorithm can deliver a general heuristic approach in the process of searching for the ideal path [7]. Compared to Dijkstra algorithm, A* have higher path search efficiency [3] but it has longer computation time [10] but both of them has fair performance and can be implemented to accomplished real-time path planning. The search algorithm that used by A* is best-first search [15]. The difference between Dijkstra's and best-first search is that Dijkstra's algorithm favors to search for nodes near the initial point, while a best-first search favors node near the destination point. A* acts to balance the two solutions to ensure that the node with the lowest traverse cost is selected at every level. According to [2], A* cannot achieve continuous path but it ensures that the shortest route is always followed in the direction of the target node [3]. The modified A* algorithm has been implemented in various path planning application [15]-[19].

2.3. D* algorithm

Stentz [20] is the first to introduce dynamic A*(D*). It's a modified version of A* that's been programmed to swiftly repair solutions when the structure changes. At each state in the traverse, an optimal path to the goal is achieved, providing all known information at each step is correct [21]. When the nodes in

the graph change, only the new costs of the nodes are updated, allowing the prior path to be exploited. Because it does not have to re-plan the entire path through the end, D* extends fewer nodes as compared to A*. The analysis of a node's neighbors is used to determine the cost of motion from the current node to the neighbor [22]. D* is most efficient when these changes of the nodes are detected near the current starting point in the search space making it suitable for robots having on board sensor. The algorithm can provide optimal and efficient path as well as managing the full spectrum of map information, from complete and accurate map information to little or no map information. [23]. In [24] D* focused was proposed with the goal of improving the characteristics of D*. This algorithm improved the expansion by minimizing the number of nodes that needed to be analyzed as well as the computing time. Author in [25] presented a 3D grid D* algorithm in which demonstrated that the change could deliver real-time performance at a lower cost. A modified approach of D* for terrain-based path planning is proposed in [26]. Besides the distance to be travelled, the terrain slope estimate is also considered in the cost function computation to plan the path. When compared to the D* method, the modified D* algorithm generates more efficient results since it is able to avoid peaks and so reach the end destination in more efficient way.

3. SAMPLING BASED PATH PLANNING

Sampling-based approaches are implemented throughout the search within configuration space where information is acquired from a collision detector. The path depends on the possible configuration and checks collision so that the configuration's validity can be verified and produce results that match with the target configuration. While this random approach has advantages in providing quick result to difficult problems [27], the algorithm lacks information on the existence of the object in the configuration space since collision testing is only performed when necessary [28]. The improvement, description, application, and improvement of sampling-based algorithm are being reviewed thoroughly in [27]. According to [6], [29], In a complex and realistic setting, sampling-based algorithms are more promising compared to graph-based algorithms because it is simpler in aspects of representation and computation. In sampling-based path planning, there are two common methods; rapidly exploring random tree (RRT) and probabilistic roadmap (PRM).

3.1. Rapidly-exploring random tree (RRT)

RRT is a tree-growing algorithm that grows a tree from the initial point to the target point, or vice versa. A point is chosen at random from the configuration space, and if it is in free space, a connection to the closest vertex in the tree is attempted, resulting in a rapidly exploring random tree [27], as shown in Figure 2. RRT have been proved to be effective at addressing complicated path-planning problems in high dimensional environments. In addition, RRT is conceptually simple and able to attain probabilistic completeness [28]. On the other hand, the traditional RRT are not guaranteed to achieve optimality or even producing high-quality path. In [30], It has been shown that under moderate space conditions, the cost of the best route in the RRT approaches a non-optimal value, thus the algorithm has been modified to obtain optimal result. Researchers in [31] has improved the algorithm to achieve optimal path planning in a cost space while describes an anytime motion algorithm that based on the RRT* which able to quickly discover an initial possible solution and then converges to an optimal solution. Improved algorithm in [32] suitable for handling the path planning with threat region and dynamic constraint. The old RRT algorithm was modified in this approach by deleting unnecessary nodes and constructing a transition trajectory, which increased the UAV's safety and maneuverability.

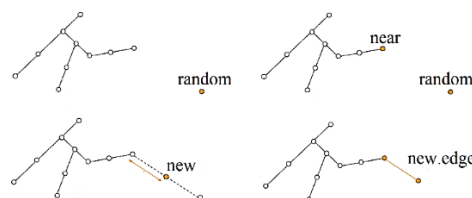


Figure 2. Procedure of extending RRT [27]

3.2. Probabilistic roadmap method (RPM)

PRM consist of several steps. The first step is configuration. By selecting coordinates at random, configurations are sampled. Then, in phase two, the sampled configurations are checked for collision to avoid obstacles. The collision-free for start and target configurations are kept as milestones, and each milestone is

connected to its closest neighbors by straight paths. From the linked milestone, the collision-free links are retained as local paths as new configuration to form the PRM. This is shown in Figure 3.

PRM generally applied in large and complex configuration space. It has been shown in [33], [34] that PRM is probabilistically complete. Based on [35], the rate of convergence of PRM on the other hand, is slow, and paths generated are not optimal. According to [29] the solutions from PRM can lead the device to fail and move very close to the obstacles in the configuration, making it impractical. Various modification has been done to the original PRM for better result. The resulting roadmap from modification made in [29] can then be applied to produce much practical paths. In [36] with some improvement of the path efficiency and cost, RPM has been implemented to solve path planning problems under unknown environment. In [37], the author proposes a method for quadrotor UAVs to fly in formation with collision avoidance using PRM.

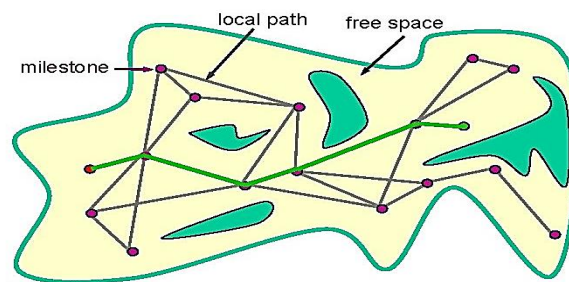


Figure 3. PRM process

4. BIOLOGICALLY INSPIRED PATH PLANNING

Biologically inspired path planning is one of the major subsets of natural computation. It is described as the combination of connectionism, social behavior, and emergence. With the use of computers, this method is being implemented to model living phenomenon, and at the same time it attempts to enhance the use of computer for a better future. Natural inspired path planning algorithm can be categorized into three methods [38], which evolutionary, swarm intelligence and neurodynamic. In this paper, algorithms from two types of method are being discussed which genetic algorithm using evolutionary method, also particle swarm optimization (PSO) and ACO which used swarm intelligence method. Research article [39], [40] discuss thoroughly on the swarm optimization technique such as PSO, ACO and others.

4.1. Genetic algorithm (GA)

A genetic algorithm (GA) resides to meta-heuristic search algorithms [4], [7] that is motivated by the principle of natural evolution of Charles Darwin. GA is based on natural genetics, which benefits from processes such as natural selection, crossover, and mutation [41]. This method used the natural selection system in which the most suitable individuals are chosen for reproduction to create next-generation offspring. This biological evolution can be applied to solve both constrained and unconstrained optimization problems. GA can rapidly obtain any solution but it could result in local optimum solution if the algorithm operates in an improperly-defined fitness function [38] as the convergence speed will reduce when it approaches the optimal solution [3], thus making GA computationally expensive and practically incomplete [28]. Algorithm improvement in [42] shows that their method can boost the global search ability of genetic algorithm, as well as improving the quality and accuracy of UAV flight path. GA also being combined with PRM in [43] to solve mobile robot path because as compared to other methods, GA has the potential to look for optimal solutions in a larger search space. In [44], the crossover, selection and mutation of GA helps to improve energy optimization path planning for near optimal or optimal solution. To deal with scenarios involving obstacles and buildings, a coverage path planning approach based on 3D structure mapping is proposed in [45]. The coverage path is calculated using a GA, and only the free spaces and areas with target below the height flying are considered.

4.2. Particle swarm optimization (PSO)

Particle swarm optimization (PSO) is a traditional meta-heuristic practically used to address global optimization issues using the swarming characteristic of biological populations. It was created in the mid-1990s as a means of recreating the well-choreographed motion of a flock of birds [46]. Figure 4 shows how each particle find its next location towards the target. Each particle in the algorithm changes its conditions to find target based on velocity value. As shown in (1) velocity value indicates how much distance, position and speed of a particle can be modified and it is affected by three factors; its own inertia, particle memory

influence that pulls particle toward personal best (pBest), and swarm influence that pulls particles towards swarm best (gBest) [40]. A pBest value specifies how close the particle's data has ever come to the target. When the neighborhood of a particle forms a swarm, the best position in the neighborhood, gBest is obtained.

$$V_i^{t+1} = V_i^t + c_1 r_{1i} (pBest_i^t - X_i^t) + c_2 r_{2i} (gBest_i^t - X_i^t) \tag{1}$$

With the balance the pBest and gBest operations in PSO, it eases in the process of generating an optimum path [4], [28]. According to [47], because of its straightforward implementation theory and ability to provide gBest for all particles, PSO is well suited for use in UAV route planning and other optimization tasks. It also is easy to be implemented [38], high precision and fast convergence [3], but if the environment become complicated, it can lead convergence speed issues [38]. PSO is able to give path completeness according to review in [28]. In [48], a new PSO-based technique called Adaptive Particle Swarm Optimization is developed, and it is compared to PSO in terms of path length and time in static settings, and it successfully avoids obstacles and reaches the goal in less time than traditional PSO. The comprehensively improved PSO proposed and analyzed by author in [47] capable of producing faster convergence and optimal solution.

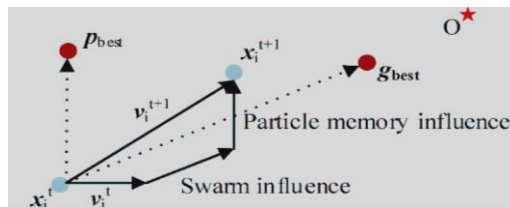


Figure 4. Particle swarm optimization process

4.3. Ant colony optimization (ACO)

Marco Dorigo introduced ant colony optimization (ACO) in 1992. ACO, like PSO, is a meta-heuristic and probabilistic technique focused on ant colony activity in searching for food and forming paths after finding its source. Ant, pheromone, daemon action, and decentralized control are four key components of ACO [39]. Ants release pheromones as they move through the search area, and the quantity of these pheromones indicates the trail's intensity. Daemon acts are used to collect global information to decide if additional pheromones need to be introduced in order to promote convergence. Decentralized control is then implemented to make it more robust and flexible in a complex setting. Figure 5 shows ant colony optimization Algorithm processes where early process, the ants start find a path between nest and food and lay pheromone. The ants then went through all possible paths and lastly majority of them opting for the one with the highest pheromone.

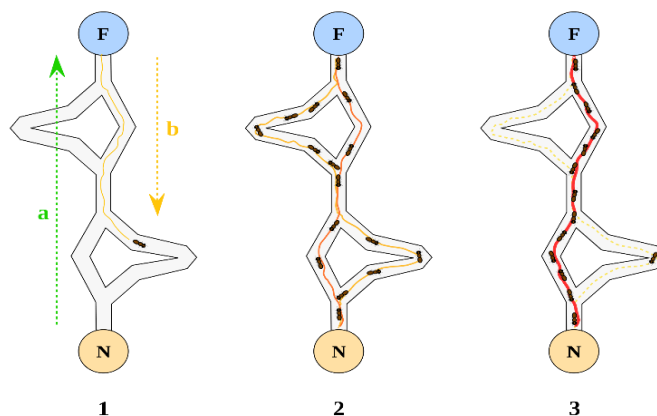


Figure 5. Ant colony optimization algorithm processes [39]

Problems with specific and clearly predefined source and destination are the most fit for ACO implementation [41] and more applicable for problems that requires crisps results. The benefits of ACO

include rapid exploration of good solution finding and distributed computing, which prevents premature convergence [39]. ACO can adapt to new changes making it suitable to be implemented in dynamic applications. ACO, has disadvantages, such as slow convergence speed as compared to other metaheuristic approaches. The convergence is guaranteed, but as the complexity of the search space increases, the convergence time becomes uncertain [38], [40]. As in [10], the comparison result proves that ACO has the longest simulation run time. Ant colony optimization is being enhanced in [49]. Experimental result of the implementation in complex situation with dense obstacles shows that the enhanced algorithm is able to provide satisfactory path planning and meet the computational time requirement. Research in [50] shows that in both simple and complex environments, ACO can discover a near optimal path and avoid obstacles in a timely manner. The new dynamic algorithm proposed in [51] incorporates ACO with potential field. It uses an artificial field to simulate the environment for collision-free path planning for the UAV while taking into account on both static and dynamic threats. In [52], path planning integrates immune network optimization with ant colony optimization to improve the ability of a multi UAV system to find the shortest path.

5. SUMMARY ON PATH PLANNING TRAITS

Table 1 summarizes the observation of traits in terms of optimal path, probabilistic completeness, and application areas for each algorithm. Three graph search algorithms discussed in this paper, all applicable in finding shortest path planning and has fair performance and can implemented to complete online real-time path planning. Dijkstra's algorithm cannot always provide optimal path and not suitable for vast and high dimensional area in terms of run time as its dependency on the number of nodes. A* being introduced to address the shortcoming of Dijkstra's, make used the advantages of heuristic method. D* is then introduced to swiftly repair solutions when the environment changes. For sampling-based method, it is being found that both algorithms, RRT and PRM are both suitable in solving complex path planning problem in high-dimensional spaces. They are able to achieve probabilistic completeness but having slow computation time and path produces are not optimal. In biologically inspired algorithm, GA is inspired from evolutionary method while PSO, and ACO are inspired by swarm. GA is suitable to be applied in finding solutions in wide search space and rapidly obtain any solution efficiently. PSO in other hand having fast convergence speed but it becomes problem as the environment becomes complicated but it still able to provide path optimality and completeness making it fit for problems with dynamically changing landscapes and to find multiple solution. Lastly, ACO ideal for problems with predefined source and destination. It able to give path completeness which suitable for problems that require firm and clean result.

Table 1. Path planning algorithm summary

	Optimal Path	Probabilistic Completeness		Application	Implemented in
Dijkstra's	No	No	Graph Search	Finding shortest single source path planning	[11]-[13]
A*	Yes	No		Finding shortest path planning	[15]-[19]
D*	Yes	No		Solve graph-based cost optimization problem for which arc costs change during the traverse of the solution path.	[25], [26]
RRT	No	Yes	Sampling Based	Solve complex path planning problem in high-dimensional spaces	[30]-[32], [53]
RPM	No	Yes		Solve complex path planning problem in high-dimensional spaces	[29], [36], [37]
GA	No	No	Biologically Inspired	Search for optimum solutions in a wider search space	[42]-[45], [54]
PSO	Yes	Yes		Solve problems with dynamically changing landscapes, and to find multiple solutions	[47], [48], [55], [56]
ACO	No	Yes		Solve problems with predefined source and destination. Applicable for problems that requires crisps results	[49]-[52], [57]

5.1. PSO as efficient path planning

In the context of expending and vast network of internet of things (IoT), UAVs are being utilized to collect uplink data from ground IoT devices by acting as aerial gateway (AG). Using aerial gateway can reduce the energy used by IoT devices as the IoT devices (ID) cannot transmit data over long distance, but UAV also has its own drawback in terms of battery capacity, which then can reduce the flight time. In order for the UAV to cover all the desired location in limited flight time, it requires a sophisticated path planning that gives the shortest path for the AG to fly and at the same instant can reduce the time travel. In order for the AG to obtain data from all of those ID, instead of visiting each ID individually, one method has been

proposed which is K-PSO Method. In this method, ID is grouped based on their distance from one another, AG only need to visit center of each cluster, as illustrated in Figure 6. K-PSO consists of K-means method to group the ID into clusters and determined the location of cluster center, known as centroid on the ground level. The stop point of AG on the air level is perpendicular to the location of the centroid. PSO method is then integrated to find the shortest and optimal path to connect all those stop points as route for the AG.

Figure 7(a) shows an environment where 20 ID being placed randomly. Figure 7(b), those ID are grouped into eight clusters using K-Means method, and the stop points each cluster is represented by number 0,1,2,3,4,5,6,7 respectively and point 8 indicates the take-off and landing point of the AG. The shortest path connecting them is obtained using PSO method and the distance between each stop point is shown in Table 2. The result shows that PSO able to connect multiple stops points with rules of visiting only visiting each stop point once during the operation.

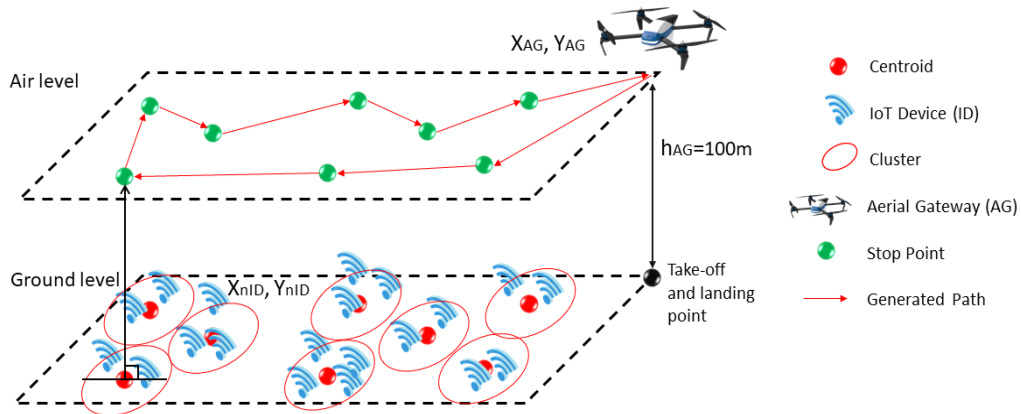


Figure 6. Network topology

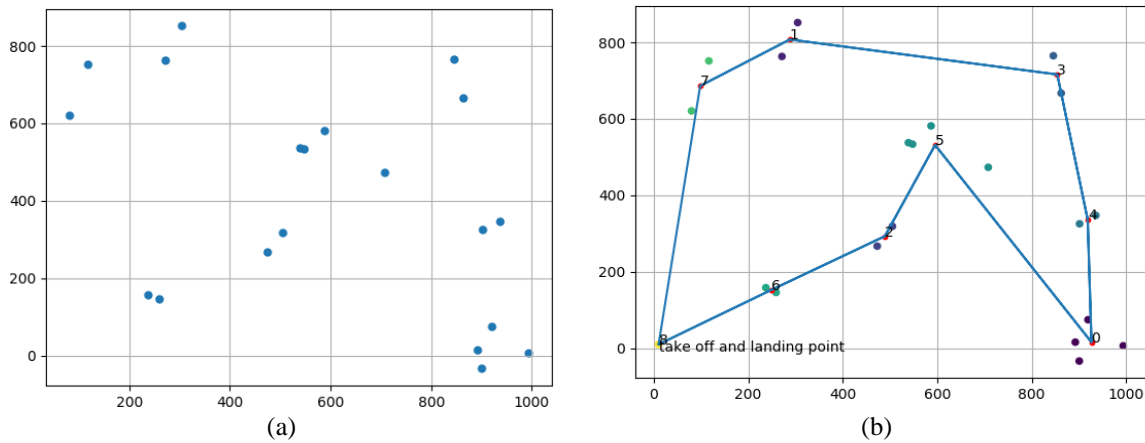


Figure 7. Shows; (a) simulation environment with random ID, (b) map of PSO connecting multiple stop points

Table 2. Path planning algorithm summary

Shortest Route	Distance (Meter)
8	277.196
6	279.248
2	261.309
5	613.025
0	320.859
4	385.069
3	573.77
1	225.261
7	381.938
Total Distance	3617.675

6. CONCLUSION

This paper presented some algorithms used in the path planning of UAV. The algorithm discussed being classified into its category and by what it is inspired by. Also, brief operation of each one is summarized for better understanding on the algorithm. The advantages and disadvantages of each algorithm also briefly described. Evaluation of different path planning algorithm in terms optimal path, probabilistic completeness and computation time along with their application in specific problems has been represented in this paper. It was possible to conclude that each algorithm has their own traits making it applicable in different type of path planning problems. The original or primary algorithm may be lack in certain characteristics but with the improvement of the algorithm and integration with other techniques may result in more efficient solution in solving sophisticated problems. The brief result on using PSO to connect multiple stop points is also being represented.

ACKNOWLEDGEMENTS

This research was supported by Universiti Tun Hussein Onn Malaysia (UTHM) through Geran Penyelidikan Pascasiswazah (GPPS) (Vot H594). Communication of this research is made possible through monetary assistance by Universiti Tun Hussein Onn Malaysia and the UTHM Publisher's Office via Publication Fund E15216.

REFERENCES

- [1] T. Cabreira, L. Brisolaro, and P. R. Ferreira, "Survey on Coverage Path Planning with Unmanned Aerial Vehicles," *Drones*, vol. 3, no. 1, p. 4, Jan. 2019, doi: 10.3390/drones3010004.
- [2] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A Review of Motion Planning Techniques for Automated Vehicles," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135-1145, April 2016, doi: 10.1109/TITS.2015.2498841.
- [3] H. Zhang, W. Lin, and A. Chen, "Symmetry Path Planning for the Mobile Robot: A Review," *Symmetry*, vol. 10, no. 10, p. 480, 2018, doi: 10.3390/sym10100450.
- [4] O. Souissi, R. Benatitallah, D. Duvivier, A. Artiba, N. Belanger, and P. Feyzeau, "Path Planning: A 2013 Survey," *2013 International Conference on Industrial Engineering and Systems Management*, pp. 1-8, 2013.
- [5] M. M. Costa and M. F. Silva, "A Survey on Path Planning Algorithms for Mobile Robots," *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2019, pp. 1-7, doi: 10.1109/ICARSC.2019.8733623.
- [6] L. Ou, W. Liu, X. Yan, Y. Chen, and J. Liang, "A Review of Representation, Model, Algorithm and Constraints for Mobile Robot Path Planning," *2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)*, 2018, pp. 563-569, doi: 10.1109/ITOEC.2018.8740620.
- [7] M. Radmanesh, M. Kumar, P. H. Guentert, and M. Sarim, "Overview of Path-Planning and Obstacle Avoidance Algorithms for UAVs: A Comparative Study," *Unmanned Systems*, vol. 6, no. 2, pp. 95-118, 2018, doi: 10.1142/S2301385018400022.
- [8] M. Sniedovich, "Dijkstra's Algorithm Revisited: The Dynamic Programming Connexion," *Control and Cybernetics*, vol. 35, no. 3, pp. 599-620, 2006.
- [9] Q. Li, Z. Zeng, B. Yang, and T. Zhang, "Hierarchical route planning based on taxi GPS-trajectories," *2009 17th International Conference on Geoinformatics*, 2009, pp. 1-5, doi: 10.1109/GEONFORMATICS.2009.5293532.
- [10] Z. He and L. Zhao, "The Comparison of Four UAV Path Planning Algorithms Based on Geometry Search Algorithm," *2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 2017, pp. 33-36, doi: 10.1109/IHMSC.2017.123.
- [11] H. Niu, Y. Lu, A. Savvaris, and A. Tsourdos, "Efficient Path Planning Algorithms for Unmanned Surface Vehicle," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 121-126, 2016, doi: 10.1016/j.ifacol.2016.10.331.
- [12] S. Julius Fusic, P. Ramkumar, and K. Hariharan, "Path planning of robot using modified dijkstra Algorithm," *2018 National Power Engineering Conference (NPEC)*, 2018, pp. 1-5, doi: 10.1109/NPEC.2018.8476787.
- [13] G. Qing, Z. Zheng, and X. Yue, "Path-planning of automated guided vehicle based on improved Dijkstra algorithm," *2017 29th Chinese Control And Decision Conference (CCDC)*, 2017, pp. 7138-7143, doi: 10.1109/CCDC.2017.7978471.
- [14] Y. Chen and Y. Chen, "Path planning in large area monitoring by drones," *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, 2018, pp. 295-299, doi: 10.1109/ICACI.2018.8377472.
- [15] Z. Zhao and R. Liu, "An optimized method for A* algorithm based on directional guidance," *2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 2015, pp. 986-989, doi: 10.1109/ICSESS.2015.7339219.
- [16] X. Chen, X. Chen, and J. Zhang, "The Dynamic Path Planning of UAV Based on A * Algorithm," *Applied Mechanics and Materials*, vol. 494-495, pp. 1094-1097, 2014, doi: 10.4028/www.scientific.net/AMM.494-495.1094.
- [17] J. Stastný, V. Skorpil, and L. Cizek, "Traveling Salesman Problem optimization by means of graph-based algorithm," *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*, 2016, pp. 207-210, doi: 10.1109/TSP.2016.7760861.

- [18] F. H. Tseng, T. T. Liang, C. H. Lee, L. D. Chou, and H. C. Chao, "A Star Search Algorithm for Civil UAV Path Planning with 3G Communication," *2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2014, pp. 942-945, doi: 10.1109/IIH-MSP.2014.236.
- [19] F. A. Raheem and A. A. Hussain, "Applying A * Path Planning Algorithm Based on Modified C-Space Analysis," *Al-Khwarizmi Engineering Journal*, vol. 13, no. 4, pp. 124-136, 2017, doi: 10.22153/kej.2017.03.007.
- [20] A. Stentz, "Optimal and efficient path planning for partially-known environments," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, vol. 4, 1994, pp. 3310-3317, doi: 10.1109/ROBOT.1994.351061.
- [21] A. Stentz, "The D * Algorithm for Real-Time Planning of Optimal Traverses," Carnegie-Mellon Univ Pittsburgh Pa Robotics Inst., p. 30, 1994.
- [22] L. De Filippis, G. Guglieri, and F. Quagliotti, "Path planning strategies for UAVS in 3D environments," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 65, pp. 247-264, 2012, doi: 10.1007/s10846-011-9568-2.
- [23] A. Stentz, "Optimal and Efficient Path Planning for Partially-Known Environments," *Intelligent unmanned ground vehicles*, Springer, Boston, MA, pp. 3310-3317, 1994.
- [24] A. Stentz, "The Focussed D * Algorithm for Real-Time Replanning," *Proceedings of 14th International Joint Conference on Artificial Intelligence*, 1995, pp. 1652-1659.
- [25] J. Carsten, D. Ferguson, and A. Stentz, "3D Field D: Improved Path Planning and Replanning in Three Dimensions," *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3381-3386, doi: 10.1109/IROS.2006.282516.
- [26] C. Saranya, M. Unnikrishnan, S. A. Ali, D. S. Sheela, and V. R. Lalithambika, "Terrain Based D* Algorithm for Path Planning," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 178-182, 2016, doi: 10.1016/j.ifacol.2016.03.049.
- [27] M. Elbanhawi and M. Simic, "Sampling-Based Robot Motion Planning: A Review," in *IEEE Access*, vol. 2, pp. 56-77, 2014, doi: 10.1109/ACCESS.2014.2302442.
- [28] S. K. Debnath, R. Omar, and N. B. A. Latip, "A Review on Energy Efficient Path Planning Algorithms for Unmanned Air Vehicles," *Computational Science and Technology*, pp. 523-532, 2019, doi: 10.1007/978-981-13-2622-6_51.
- [29] J. Yang, P. Dymond, and M. Jenkin, "Practicality-Based Probabilistic Roadmaps Method," *2011 Canadian Conference on Computer and Robot Vision*, pp. 102-108, 2011, doi: 10.1109/CRV.2011.21.
- [30] S. Karaman and E. Frazzoli, "Incremental Sampling-based Algorithms for Optimal Motion Planning," *Robotics: Science and Systems*, vol. 6, pp. 267-274, 2011, doi: 10.7551/mitpress/9123.003.0038.
- [31] D. Devaurs, T. Siméon, and J. Cortés, "Optimal Path Planning in Complex Cost Spaces with Sampling-Based Algorithms," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 415-424, 2016, doi: 10.1109/TASE.2015.2487881.
- [32] D. Devaurs, T. Siméon and J. Cortés, "Optimal Path Planning in Complex Cost Spaces With Sampling-Based Algorithms," in *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 415-424, April 2016, doi: 10.1109/TASE.2015.2487881.
- [33] D. Hsu, J.-C. Latombe, and H. Kurniawati, "On the Probabilistic Foundations of Probabilistic Roadmap Planning," *The International Journal of Robotics Research*, vol. 25, no. 7, pp. 627-643, 2006, doi: 10.1177/0278364906067174.
- [34] J. Barraquand, L. Kavraki, J. C. Latombe, R. Motwani, T. Y. Li, and P. Raghavan, "A Random Sampling Scheme for Path Planning," *International Journal of Robotics Research*, vol. 16, no. 6, pp. 759-774, 1997, doi: 10.1177/027836499701600604.
- [35] Z. Kong and B. Mettler, "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance," *Journal of Intelligent & Robotic Systems*, vol. 57, pp. 65-100, 2010, doi: 10.1007/s10846-009-9383-1.
- [36] Zhiye Lee and Xiong Chen, "Path planning approach based on probabilistic roadmap for sensor based car-like robot in unknown environments," *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, vol. 3, 2004, pp. 2907-2912, doi: 10.1109/ICSMC.2004.1400774.
- [37] M. U. Farooq, Z. Ziyang and M. Ejaz, "Quadrotor UAVs Flying Formation Reconfiguration with Collision Avoidance Using Probabilistic Roadmap Algorithm," *2017 International Conference on Computer Systems, Electronics and Control (ICCSEC)*, 2017, pp. 866-870, doi: 10.1109/ICCSEC.2017.8446781.
- [38] J. Li, S. X. Yang and Z. Xu, "A Survey on Robot Path Planning using Bio-inspired Algorithms," *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2019, pp. 2111-2116.
- [39] M. Nadhir, A. Wahab, S. Nefti-meziari, and A. Atyabi, "A Comprehensive Review of Swarm Optimization Algorithms," *PLoS ONE*, pp. 1-36, 2015, doi: 10.1371/journal.pone.0122827.
- [40] V. Selvi and S. Tamilnadu, "Comparative Analysis of Ant Colony and Particle Swarm Optimization Techniques," *International Journal of Computer Applications*, vol. 5, no. 4, pp. 1-6, 2010, doi: 10.5120/908-1286.
- [41] T. T. Mac, C. Copot, D. T. Tran, and R. De Keyser, "Heuristic Approaches in Robot Path Planning: A survey," *Robotics and Autonomous Systems*, vol. 86, pp. 13-28, 2016, doi: 10.1016/j.robot.2016.08.001.
- [42] J. Tao, C. Zhong, L. Gao, and H. Deng, "A Study on Path Planning of Unmanned Aerial Vehicle Based on Improved Genetic Algorithm," *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 2016, pp. 392-395, doi: 10.1109/IHMSC.2016.182.
- [43] R. M. C. Santiago, A. L. De Ocampo, A. T. Ubando, A. A. Bandala, and E. P. Dadios, "Path planning for mobile robots using genetic algorithm and probabilistic roadmap," *2017 IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2017, pp. 1-5, doi: 10.1109/HNICEM.2017.8269498.

- [44] T. R. Schäfle, S. Mohamed, N. Uchiyama, and O. Sawodny, "Coverage path planning for mobile robots using genetic algorithm with energy optimization," *2016 International Electronics Symposium (IES)*, 2016, pp. 99-104, doi: 10.1109/ELECSYM.2016.7860983.
- [45] M. M. Trujillo, M. Darrah, K. Speransky, B. DeRoos and M. Wathen, "Optimized flight path for 3D mapping of an area with structures using a multicopter," *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2016, pp. 905-910, doi: 10.1109/ICUAS.2016.7502538.
- [46] M. Juneja and S. K. Nagar, "Particle swarm optimization algorithm and its parameters: A review," *2016 International Conference on Control, Computing, Communication and Materials (ICCCCM)*, 2016, pp. 1-5, doi: 10.1109/ICCCCM.2016.7918233.
- [47] S. Shao, Y. Peng, C. He, and Y. Du, "Efficient Path Planning for UAV Formation Via Comprehensively Improved Particle Swarm Pptimization," *ISA Transactions*, vol. 97, pp. 415-430, 2020, doi: 10.1016/j.isatra.2019.08.018.
- [48] H. S. Dewang, P. K. Mohanty, and S. Kundu, "A Robust Path Planning for Mobile Robot Using Smart Particle Swarm Optimization," *Procedia Computer Science*, vol. 133, pp. 290-297, 2018, doi: 10.1016/j.procs.2018.07.036.
- [49] Z. Zhou, Y. Nie, and G. Min, "Enhanced Ant Colony Optimization Algorithm for Global Path Planning of Mobile Robots," *2013 International Conference on Computational and Information Sciences*, 2013, pp. 698-701, doi: 10.1109/ICCIS.2013.189.
- [50] R. Rashid, N. Perumal, I. Elamvazuthi, M. K. Tageldeen, M. K. A. Ahamed Khan and S. Parasuraman, "Mobile robot path planning using Ant Colony Optimization," *2016 2nd IEEE International Symposium on Robotics and Manufacturing Automation (ROMA)*, 2016, pp. 1-6, doi: 10.1109/ROMA.2016.7847836.
- [51] C. Huang *et al.*, "A New Dynamic Path Planning Approach for Unmanned Aerial Vehicles," 2018, doi: 10.1155/2018/8420294.
- [52] W. Hao and X. Xu, "Immune ant colony optimization network algorithm for multi-robot path planning," *2014 IEEE 5th International Conference on Software Engineering and Service Science*, 2014, pp. 1118-1121, doi: 10.1109/ICSESS.2014.6933762.
- [53] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, "Anytime Motion Planning using the RRT*," *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1478-1483, doi: 10.1109/ICRA.2011.5980479.
- [54] M. Paradzik and G. İnce, "Multi-agent search strategy based on digital pheromones for UAVs," *2016 24th Signal Processing and Communication Application Conference (SIU)*, 2016, pp. 233-236, doi: 10.1109/SIU.2016.7495720.
- [55] Y. K. Ever, "Using Simplified Swarm Optimization on Path Planning for Intelligent Mobile Robot," *Procedia Computer Science*, vol. 120, pp. 83-90, 2018, doi: 10.1016/j.procs.2017.11.213.
- [56] X. Wu, W. Bai, Y. Xie, X. Sun, C. Deng, and H. Cui, "A hybrid algorithm of particle swarm optimization, metropolis criterion and RTS smoother for path planning of UAVs," *Applied Soft Computing Journal*, vol. 73, pp. 735-747, 2018, doi: 10.1016/j.asoc.2018.09.011.
- [57] X. Yang and J. Wang, "Application of improved ant colony optimization algorithm on traveling salesman problem," *2016 Chinese Control and Decision Conference (CCDC)*, 2016, pp. 2156-2160, doi: 10.1109/CCDC.2016.7531342.

BIOGRAPHIES OF AUTHORS



Nurul Saliha Amani Ibrahim currently pursuing her study in Master of Engineering Technology at UTHM. She previously received her degree in Electronic Engineering Technology (Communication and Computer) at University Tun Hussein Onn Malaysia (UTHM) in 2019.



Faiz Asraf Saparudin received his Bachelor B.Sc. in Electrical Engineering (Telecommunication) with First Class Honours from Universiti Teknologi Malaysia in 2010 and received WAMY Academic Excellence Award in the same year. Ph.D. degree in Electrical Engineering (Telecommunication) from the Universiti Teknologi Malaysia in 2015. He is currently a Faculty Member in Fakulti Teknologi Kejuruteraan, Universiti Tun Hussein Onn Malaysia. Member of Institute of Electrical and Electronic Engineers (IEEE) and IEEE Communication Society (ComSoc). His current research interests include radio resource management, distributed algorithms, nature-inspired techniques, multiagent system and game theoretic approach for next-generation mobile network.