Engineering and Technology Journal
Journal homepage: engtechjournal.org

A Review on Path Planning Algorithms for Mobile Robots

Mustafa S. Abed [a], **Omar F. Lutfy** [b] , **Qusay F. Al-Doori** [c*]

[a] Control and Systems Engineering Department, University of Technology-Iraq, Baghdad, Iraq, cse.19.07@grad.uotechnology.edu.iq
[b] Control and Systems Engineering Department, University of Technology-Iraq, Baghdad, Iraq, 60157@uotechnology.edu.iq
[c] Control and Systems Engineering Department, University of Technology-Iraq, Baghdad, Iraq, Qusay.f.hasan@uotechnology.edu.iq

* Corresponding author.

Submitted: 02/12/2020 Accepted: 20/02/2021 Published: 25/05/2021

KEYWORDS

Artificial-intelligent, Classic approach ,Heuristic Methods, Intelligent Methods, Meta Heuristic Methods, Mobile Robot, ,Path planning , Randomized Methods.

ABSTRACT

Mobile robots use is rising every day. Path planning algorithms are needed to make a traveler of robots with the least cost and without collisions. Many techniques have been developed in path planning for mobile robot worldwide, however, the most commonly used techniques are presented here for further study. This essay aims to review various path planning strategies for mobile robots using different optimization methods taken recent publisher's paper in last five year.

How to cite this article: M. S. Abed, O. F., Lutfy, and Q. F. Al-Doori "Path Planning Algorithms For Mobile Robots: A Survey," Engineering and Technology Journal, Vol. 39, Part A, No. 05, pp. 804-820, 2021.
DOI: https://doi.org/10.30684/etj.v39i5A.1941

1. INTRODUCTION

Navigation, a significant factor in mobile robotics, is known as a process of accurately specify the position of the robot, planning the path, and following the planned path. The navigation issue of the MR was divided into three sub-problems, shown in Figure 1 Where is the initial point? (i.e., the localization problem), Where is the target point? (i.e., the object recognition problem or Mapping), and How to travel from the initial point to the target? (i.e., the path planning issue) [1].

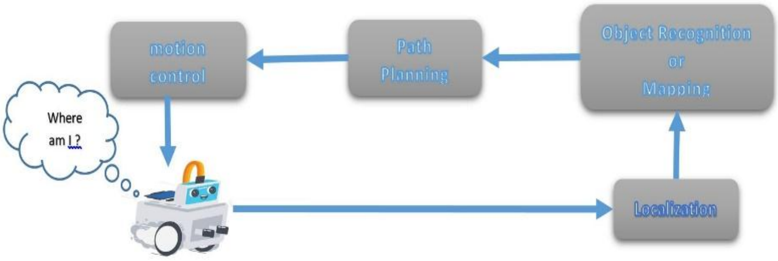

Figure 1: Mobile Robot Problem

Localization: the robot must localize its position in the environment;[2] **Mapping or Recognition**: The robot needs a map of its environment to determine where it has been moving around so far. The map helps the robot understand directions and locations;[1] and **Path planning:** To find a route for the mobile robot, where the target direction must be identified in advance by the robot requiring an effective robot adjustment scheme [1], and **Motion Control**: the robot must modulate its motor output to reach the desired route,[2].

Path planning is a process of obtaining a reasonable, collision-free route between start and goal points [3]. Path planning becomes an important issue for a fully or partially automated process and can be considered as one of the essential problems in the robotics world. Because of the high demand for using robots in different fields, path planning becomes an interesting field for researchers [4]. An environment of robots can be divided into two categories dynamic and static environments. Usually, the dynamic environment contains moving and non-moving obstacles, while the static environment contains only non-moving obstacles. Also, the environment can be divided into known and unknown environments according to how much information is known about the environment. Usually, the obstacle avoidance term is used with unknown environment, which is mean the robots move in the free space without any collision with the environment's obstacles [5]. Due to the different purposes and functions of the same functional robot as shown in Figure 2, many problems must be considered in planning the path of a mobile robot [6].

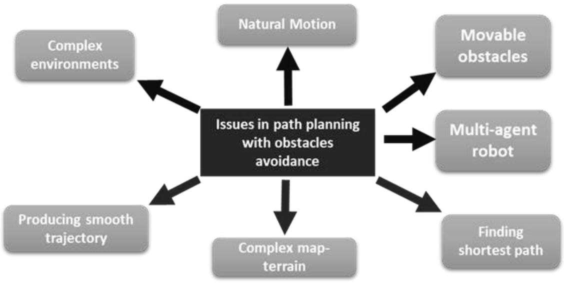

Figure 2: Issues in path planning

Figure 2 illustrates the issues in path planning with hurdles avoidance, including the following points:

- **Movable obstacles:**

While moving obstacles, the issues of path planning is a practical one, as the robot is permitted to move the obstacles if it blocks a robot path from the start of the target position [7].

- **Multi-agent robot:**

Multi-agent robot device control issues have taken on considerable significance. Any multi-agent robot system has a certain transport subsystem, consisting of many mobile robots. The issue of controlling such a mobile robot community can be broken down into two main parts:

- The major task is divided into subtasks and the optimal distribution among individual group robots.
- Per movable robot, path planning, control, and movement correction [8].

- **Finding the shortest path:**

An effective way to save time and power is to find the shortest path. There are two major groups of algorithms with the shortest path:

- Finding distances for any pair of points.
- Finding the shortest paths from a given point of origin [9].

- **Complex map-terrain:**

The problem of determining the terrain's characteristics, detecting the attitude of the robot on top of the surface, establishes the traversability of the mobile robot over the field [10].

- **Producing a smooth trajectory:**

The algorithm can produce smooth paths of the robot and the modified exploration rate increases the minimum expected return. The exploration rate may not be sufficiently general or difficult to implement in the actual robot manually or automatically [11].

- **Complex environments**

The issue of planning the route for mobile robots in complex environments was analyzed and tuned for disparate working areas with obstacles in different numbers, sizes, and shapes [12].

Some of the suggested methodologies focus on searching for the smallest route from the start to the target. Newly, researchers have been focusing on solving time constraints in a complex, multi-obstacle environment [13]. Other researchers focus on configuration control and obstacle bypass problems for multi-agent systems in an environment with unknown hindrances [14]. Several others consider the transferred obstacles and navigation of robots [15].

2. Methodology

The relevant work collection was performed between 2017 and 2021, resorting to the scientific search engine Google Scholar. The search was open to any path planning approach in the robotic field for mobile robots. The analysis of the related work intends to answer the following issues: i) planning approach ii) on-line or off-line capability; iv) dynamic or static; v) path optimality; vi) optimization criteria; viii) computational complexity and processing time.

3. Classification of Path Planning

Path planning is studied extensively by researchers because of its importance in robot navigation. It is important to understand path planning classification because it leads to find a suitable solution for the path planning problem. Robot path planning can be classified into different kinds depending on different aspects such as obstacles type, environment type, planning type, space type, and time as shown in Figure 3,and stated in the following points [16]:

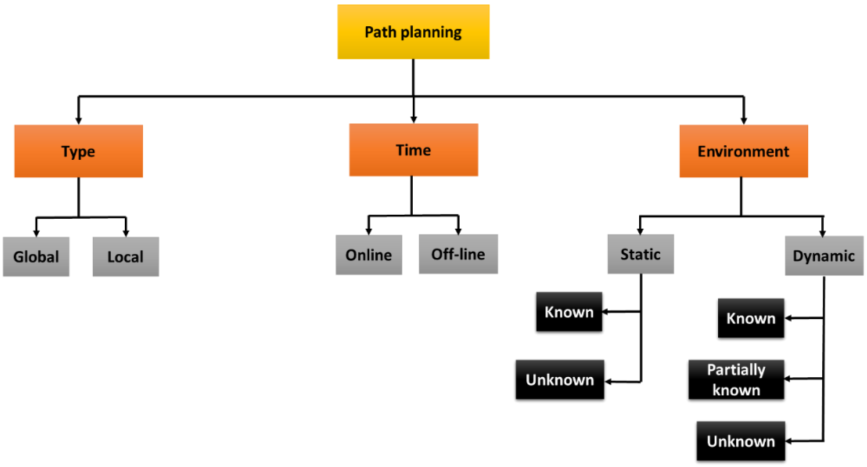

Figure 3: path planning Classification

I. Path planning based on plan type

a) Global Planning: it assumes that information about the environment is completely known. Therefore; the path is completely planned before the robot starts to move.

b) Local Planning: it is also called sensor-based planning because the information is unknown or partially known so, the robot depends on sensors to collect information. The path is planned step by step during robot motion. The difference between these methods is summarized in table (I) [17]–[19].

Table I: The difference between local and global path planning

No.	Global path planning	Local path planning
1	Work off-line	Work on-line
2	Robot Map-based	Robot Sensor-based
3	Deliberative navigation	Reactive navigation
4	workspace area is fully known and the terrain should be fixed.	workspace is not necessarily fully known or unknown
5	The algorithm produces a whole path from the initial point to the target point before the robot begins its movement.	The algorithm produces a new pathway in echo to environmental moveable.
6	Approximately slower response	Fast response

II. Path planning based on Time [20]

a) On-line: the path from the start to the goal points is planned during the robot movement depending on information coming from the robot's sensors.

b) Off-line: the path is fully planned before the robot starts to move.

III. Path planning based on the environment type [20]

a) Dynamic environment: the environment of the robot contains static and dynamic obstacles.

- Known dynamic environment.
- Partially known dynamic environment.
- Unknown dynamic environment

b) Static Environment: the environment of the robot contains only static obstacles.

- Known static environment.
- Unknown static environment.

4. Path Planning Mobile Robots Techniques and Theories

This section presents the path planning theories and techniques as seen in Figure 4.

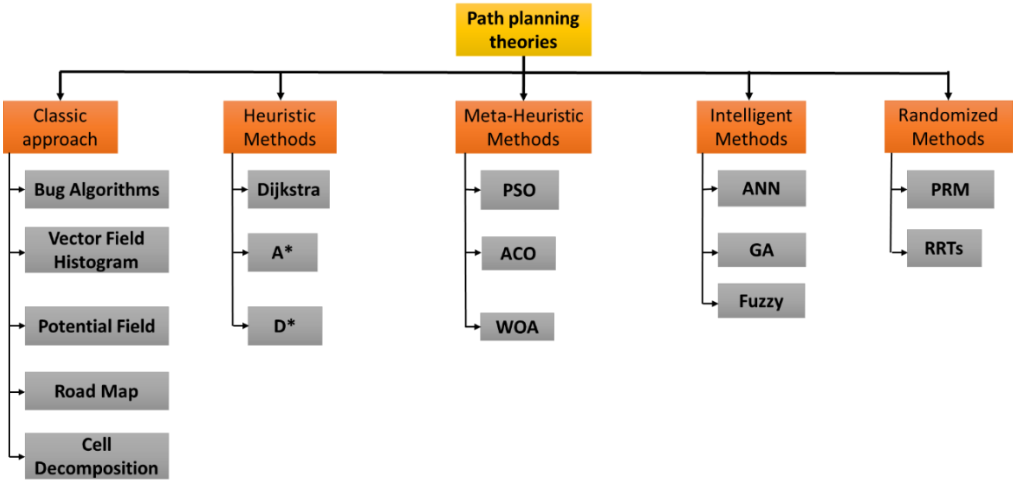

Figure 4: path planning theories Classification

I. Path planning classic approach

Methods such as Cell Decomposition (CD), Potential Field (PF), Bug Algorithm, and Vector Field Histogram are commonly used in motion mapping problems in the classical category [21],[22].as seen in Table II the recent publish paper for each classic approach between 2017 and 2021 In their basic formulation, these approaches are found to be less capable of handling unknown, partially known, or dynamic environments and are known to be computer-intensive. Moreover, in order to create a feasible path between the starting and the destination points, they are mostly found to be dependent on complete prior knowledge of the environment.

Table II: Recent Paper on Path Planning using classic approach algorithms:

No	Approach	Authors	Ref.	Year	Kinematic model	Online or Off-line	Static obstacle	Dynamic obstacle	Optimal Path	Simulation	Real system
1	Cell Decomposition (CD)	Samaniego etal	[23]	2019	N	Offline	Y	Y	Y	Y	N
2	Potential Field (PF)	F. A. Raheem and M. M. Badr	[24]	2017	N	Offline	Y	N	Y	Y	N
3		Bayat etal	[25]	2018	N	Offline	Y	N	Y	Y	N
4	Bug Algorithm	Xu et al	[26]	2017	Y	Offline	Y	N	Y	Y	N
5		Das et al	[27]	2020	N	Offline	Y	N	N	Y	N
6	Vector Field Histogram	Chen etal	[28]	2019	Y	Online	Y	N	N	N	Y

- **Cell Decomposition (CD)**

Cell Decomposition(CD) is widely used by literature in path planning Issues. It is the representation of the search space of the robots in the form of individual units called cells. The correlation will then be calculated between each neighboring cell within the goal creating a collision-free pathway from the starting point to the target. First, the barrier cells are split into two new cells, then the pure cells (non-blocking cells) are added to the sequence. [29] using Global path planning. Figure 5 shows the cell decomposition method. Samaniego et al in 2019 modified the Cell Decomposition in an adaptive grid methodology in 3D environments applied to flight path planning. The Cell Decomposition using to Create the node of the path.

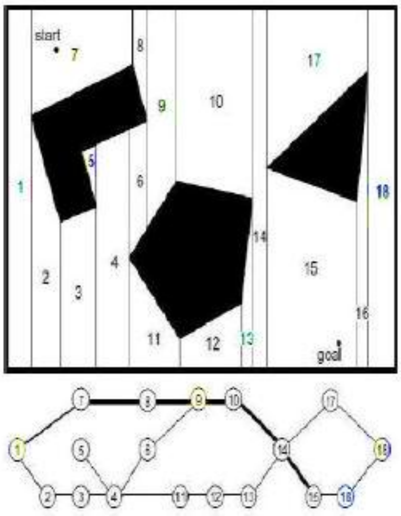

Figure 5: the cell decomposition method [22]

- **Potential Field (PF)**

For the first time, the concept of using artificial potential fields (APF) in autonomous robots to avoid obstacles was suggested by Khatib. O [30] using Global path planning. In this method, the robot reaches the target by generating attractive and repulsive forces in the robot's environment. Forces of attraction are assigned to targets and repulsion forces are assigned to obstacles. Figure (5) show the Mobile robot navigation by APF Method. F. A. Raheem and M. M. Badr in 2017 [22]

Surface Vessel based on APF is provided and using PSO discovered the best values of Artificial Potential Field (APF) factors in order to make an iterative enhancement till reaching the shortest path. Bayat et al in 2018 [23] using a potential surface is constructed representing the resultant potential fields of all obstacles and the objective is that the robot avoids the high potential areas and finds an optimal and collision-free path to the target through the potential surface. Figure 6 show the Mobile robot navigation by APF Method.

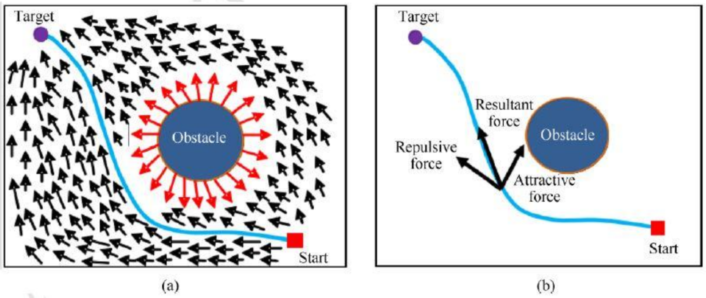

Figure 6: Mobile robot navigation by APF Method [31].

- **Bug Algorithms:**

Xu et al. [26] resolved the autonomous navigation in the unknown static environment in the robot process, Transition from starting position to goal (end) position, where the path was planned that met the kinematics parameters of the robot, by combining the bug algorithm and Dubins path. However, this method took a long time when the unknown environment is more complex and it does not work with dynamic obstacles. Based on the bug algorithm, Das et al.[27] proposed a new approach called Modified Critical Point Bug (MCPB) algorithm, which avoids run-time obstacles. This algorithm only takes into account the heads of those obstacles that generate collisions and does not take into account all obstacles. Moreover, the authors did not consider the shortest distance between the starting and target modes.

- **Vector Field Histogram**

Borenstein and Koren initially introduced this method and considered it to be an improved version of the PF method. VFH is an approach to prevent obstacles in real-time, which enables a robot to identify and avoid unknown obstacles while continuing to pursue the goal [29]. Chen et al in 2019 [26] are present a local path planning method based on VFH* the drawback did not find the optimal path.

- **Road Map method:**

Road maps are created with a set of paths in this approach where each route consists of collision-free zone connections. For route planning, these road maps are used later. Thus, route planning is reduced to searching for a series of roads from the initial location to the road network target connecting the primary and target points [32] as shown in Figure 7, for a given static environment. In 2013 the Elbanhawi et al [33] were improving algorithms that allow robots to navigate unknown environments independently is a widely studied field of robotics. This method is a set for controlling the distance from obstacles between the waypoints, without increasing its computational complexity.

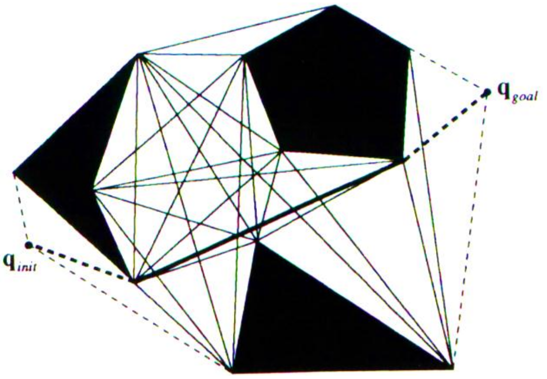

Figure 7: Roadmap [2]

II. Path plan Heuristic Methods:

The Heuristic Path Planning Approaches, although recently used in comparison to the Classical Approaches, have gained a great deal of importance due to their human-like behavior-based learning. As seen in table III the recently published paper for each Heuristic Methods between 2017 and 2021.

Table III: Recent Paper on Path Planning using Heuristic Methods:

No	Approach	Authors	Ref.	Year	Kinematic model	Online or Off-line	Static obstacle	Dynamic obstacle	Optimal Path	Simulation	Real system
1	A*	Zeng et al	[34]	٢٠١٩	N	offline	Y	N	N	Y	N
2		RAHEEM &. ABDULKAREEM	[35]	2020	N	offline	Y	N	Y	Y	N
3	D*	Hasan et al	[36]	2018	N	online	Y	N	N	Y	N
4		Raheem and Hameed	[3]	2018	Y	offline	Y	Y	Y	Y	N

- **Dijkstra method:**

The Dijkstra algorithm proposes In 1959, by E.W. Dijkstra. It is known as the shortest path algorithm where it is used to solve path planning problems. It starts from starting point to the target point. Thru the execution of the algorithm, it is always ensuring that the distance from the starting point to the current point is minimized [37].

- **A star (A*) Algorithm:**

In 1968 Hart et al. propose A* algorithm where A* can be considered as the development of the Dijkstra algorithm. The main feature of the A* method is it has a fitness function used to evaluate the found solution, where A* starts at a specific point then it chooses the next depending on fitness value. The advantage of A* over the Dijkstra method is that A* considers the goal point during the search process [38]. Zeng et al in 2019 [34] A* algorithm commonly used in mobile robot path planning game development. Although the A* algorithm is characterized by optimity, simplicity, and flexibility and has been widely used in path planning, it also has some disadvantages, such as redundant points, large overhead memory, and long computing time. F. A. RAHEEM and M. I. ABDUL KAREEM in 2020 [35] The A* heuristic method is used to find the shortest path within the constructed roadmap.

- **D star (D*)**

The D* algorithm is proposed by Stentz in 1994. It can be considered as a dynamic A* algorithm because it is used in Online path planning with a dynamic environment, therefore; D* has the ability for autonomous navigation [32].Hasan et al in 2018 [36] are present A new centralized and competitive multi-robot method in a dynamic environment for online path planning. In the same complex dynamic environment that has a variant number (from one to five) of dynamic obstacles, it finds the optimal paths to the variant number of robots that compete with each other. Raheem and Hameed at 2018 [3] use the technique of PSO optimization to get the final optimal path. In addition, by adding a stop case and return backward case, a modification to the D* algorithm was made to ensure the path solution interactivity reaction to the position of obstacles and continuous changes in time during robot motion.

III. Path planning Meta-Heuristic Methods

A metaheuristic method helps in solving the optimization problem. Problems in optimization can be found in many daily life aspects. The kinds of metaheuristic method are various which are ant colony optimization (ACO) ,particle swarm optimization (PSO) and the recent algorithm is whale optimization algorithm (WOA) [39]. the metaheuristic algorithms have been developed to reduce the drawbacks of classical approaches as they take more time to calculate the optimal path [40]. As seen in table IV the recently published paper for each Meta- Heuristic Methods between 2017 and 2021.

Table IV: Recent paper on Path Planning using Meta-Heuristic Methods:

No	Approach	Authors	Ref.	Year	Kinematic model	Online or Off-line	Static obstacle	Dynamic obstacle	Optimal Path	Simulation	Real system
1	PSO	Setyawan etal	[41]	2017	N	offline	Y	N	Y	Y	N
2		Dewang etal	[42]	2018	N	offline	Y	N	Y	Y	N
3		Krell	[43]	2019	N	online	Y	N	Y	Y	N
4		Tang et al	[44]	2020	N	offline	Y	N	Y	Y	N
5	ACO	Rajput and Kumari	[45]	2017	N	offline	Y	N	Y	Y	N
6		WANG, Tao, et al.	[46]	2018	N	offline	Y	N	Y	Y	N
7		Luo	[47]	2020	N	offline	Y	N	Y	Y	N
8		Pu	[48]	2020	N	offline	Y	N	Y	Y	N
9	WOA	Chhillar & Choudhary	[49]	2020	N	offline	Y	N	Y	Y	N

- **The Particle Swarm Optimization (PSO)**

Eberhart and Kennedy [50] developed the original PSO, which used an equation to calculate the particle velocity relative to the previous velocity and direction. The best position is a better swarm and then updated the particle position. Setyawan et al. [41]in 2017 Formulated the path planning problem with three objective functions, namely path distance, risk degree, and smoother path, as an optimization problem. Then, they analyzed the initial PSO's work for the optimization algorithm. However, the drawback of the work is that they did not find the time and the iterations number. The authors in [42] suggested an adaptive PSO algorithm (APSO) to define the solution for route planning in MR situations that were handled precisely. Krell et al. [43]used the PSO and the Gazebo simulator to create an ARN system that can navigate in an unknown environment and reach the predefined target without collisions. The findings demonstrated that in multiple environments with different features, the robot can generate and follow efficiently. Tang et al. [44]solved the multi-robot path planning issue by proposing a new-PSO-based approach. To cope with the multi-robot path-planning problem, a coevolution-based particle swarm optimization technique was designed in an attempt to amend the global and local search ability fully and address the problem of the PSO recession. The steps of the standard PSO [51][52] were itemized in Algorithm B, while Figure 8 illustrates the corresponding flowchart.

Algorithm B: Pseudo-code of PSO algorithm:

```
1- Initialize the PSO parameters
2- For t equals 1 until t equals the maximum generation
3- For I equals 1 until I equal the pop- size
4- If f (xi,d (t) ) is less than f (pi (t) ) then pi (t) equals xi,d (t)
5- Let f(Pg (t) ) equals min (f (Pi(t))
6- End
7-  For d equlas1 until d equals dimension
8- Vi,d (t+1) =vi,d (t)+c1r1(pbesti-xi,d(t))+c2r2(pgbest-xi,d(t))
9- X i,d(t+1)=xi,d(t)+vi,d(t+1)
10- If Vi,d (t+1) is bigger than  Vmax then Vi,d (t+1)equals Vmax, Else if Vi,d (t+1)is less than
    Vmin then Vi,d (t+1) equals Vmin
11- End
```

12- If X i,d(t+1) is bigger than Xmax then Xi,d (t+1) equals Xmax, Else if X i,d(t+1) is less than x min then X i,d(t+1)=X min

13- End

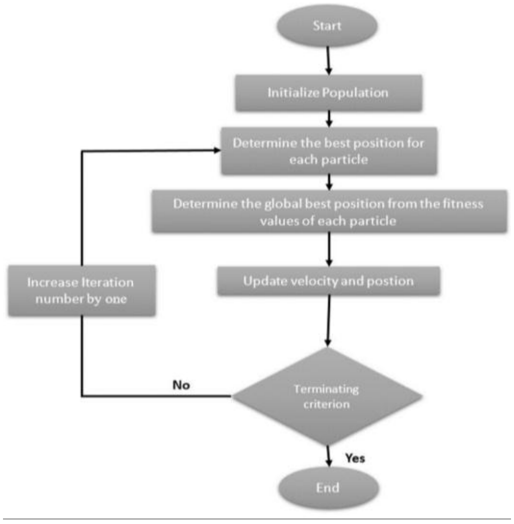
Start
Initialize Population
Determine the best position for each particle
Determine the global best position from the fitness values of each particle
Increase Iteration number by one
Update velocity and postion
No
Terminating criterion
Yes
End

Figure 8: The PSO algorithm Flowchart

Form the flowchart above; the steps of the PSO algorithm are given below:

1- Randomly configure the velocity and position of all particles in N-dimensional coordinates.
2- Estimate the suitability of each particle.
3- Match each value of particle fitness with its value of P_{best}. If the new value is better than the old best value, use the current value as the best value.
4- calculate p_{gbest} between all the particles. This p_{gbest} has the finest fitness value.
5- Recalculate the position and velocity by the following equations:
   - The velocity: $V_{i,d\,(t+1)} = v_{i,d\,(t)} + c_1 r_1 (p_{best} - x_{i,d(t)}) + c_2 r_2 (p_{gbest} - x_{\,i,d(t)})$
   - The Position: $X_{i,d(t+1)} = x_{i,d(t)} + v_{i,d(t+1)}$

where the variable t is the iterations number, r1 and r2 are random variables inside [0, 1], c1 and c2 are the velocity Change agents, Vi,d (t) is the inertia velocity, Pbest is the particle best position and Pgbest is the swarm best position .

6- Terminating criterion If the robot reaches his gold position go to the end (P_{gbest}) else go to the next iteration.

- **Ant colony optimization algorithms (ACO)**

The ACO was proposed in 1992 by M.Dorigo. The principle of ACO was that each ant releases an excretion along the path it has taken and it will also understand the excretion that other ants release while foraging for food, [53]. Rajput and Kumari [45] modified versions of the optimization of ant colonies were used. The robot's directional motion history on a grid has been incorporated into a vector. As a multiplication probability factor that helps to achieve faster convergence and prevent unnecessary movements, e.g. looping. They have developed a novel pheromone update scheme for this work. In addition to this, They have used path smoothing to minimize the number of turns on the optimal path of the filter. The drawback of the work was not given the path length. WANG, Tao, et al,[46]. The ACO enhanced long search time, low efficiency, and easy landing into local optimum for the main problem. That means a more efficient algorithm and a shorter path. Luo; Pu [47],[48] in this paper improve ACO to solve the problem of slow convergence and low search efficiency. itemized ACO in the equation and flowchart algorithm in Figure 9 show down, [54].

$$p_{ij}^{k} = \frac{\{\tau_{ij}\}^{a}}{\sum_{j\varepsilon N_i^k}^{m}\{\tau_{ij}\}^{a}} \qquad (1)$$

Where τ_{ij} is pheromone trail, α is a positive constant termed as weight values, m is the number of ants, p_{ij}^{k} is the transition probability in which ant k will move from node i to j[54] :

$$\tau_{ij}= (1-\rho) * \tau_{ij} \qquad (2)$$

where the range (0<p≤1) is the pheromones dissipation rate and change of pheromone represents in eq.3:

$$\tau_{ij} = \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij} \qquad (3)$$

Where $\Delta\tau_{ij}$ is the quantity of pheromones ant k charge on the visited paths, represents in an eq.4:

$$\Delta\tau_{ij} = \frac{1}{C^k} \qquad (٤)$$

Where C^k is the path length build by the k th

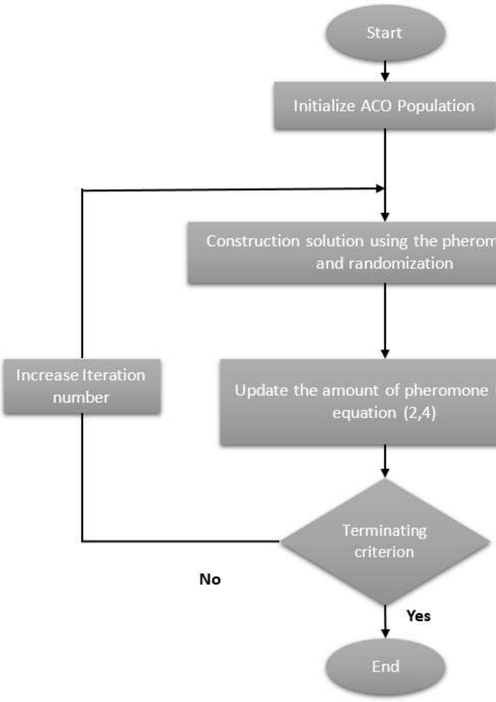
Start
Initialize ACO Population
Construction solution using the pheromone trail and randomization
Update the amount of pheromone using equation (2,4)
Increase Iteration number
Terminating criterion
No
Yes
End

Figure 9: Flow Chart of ACO Algorithm

1- **Initialization**: Two parts are involved in an ACO initialization: The first mainly consists of initializing the pheromone pathway. Secondly, on the randomly selected nodes, an arbitrary number of ants are placed. Then each of the distributed ants, by creating a path according to the rule of transmission of the node, makes a round of the graph.
2- **Solution construction**: According to the rule of probabilistic state transmission, each ant creates a whole solution to the issue. The state transition ruling is primarily based on a pheromone state and the ants sighting.
3- **Pheromone updating**: The intensity of the pheromone pathways at every edge is updated by the pheromone update referee when each ant has built a solution. In two stages, Equation 2 and Equation 4 are applied to the pheromone update rule.
4- **Terminating criterion**: steps 2 and 3 are repeated until the termination criterion.

- **The Whale Optimization Algorithm (WOA)**

It is the new a novel nature-inspired meta-heuristic optimization algorithm, called Whale Optimization Algorithm (WOA), which mimics the social behavior of humpback whales. The algorithm is inspired by the bubble-net hunting strategy.Developed by Mirjalili and Lewis in 2016 [55]. This algorithm is new recent application on path planning in 2020 as seen in [49] The author proposed that the modified whale optimization algorithm ensures an optimal collision-free path. The fitness of any whale will be calculated by taking into account the target location and the obstacles in the search space in the whale optimization algorithm (WOA).

The Mathematical model and optimization algorithm:

the WOA assumes that the current best candidate solution is the target prey or is close to the optimum. The effort is made to identify the best search agent, while the other search agents will update their positions near to the best search agent. The behavior is expressed by the following equations as stated by [56]:

$$\vec{D} = \left|\vec{C}.\overrightarrow{X^*}(t) - \vec{X}(t)\right| \qquad (1)$$

Spiral updating position the helix-shaped movement of whales is simulated by the spiral equation. The spiral equation is as follows [55]:

$$\vec{X}(t+1) = \overrightarrow{D'}.e^{bl}.\cos(2\pi l) + \overrightarrow{X^*}(t) \qquad (2)$$

Search For Prey update the position using the encircling prey method [57] :

$$\vec{X}(t+1) = \overrightarrow{X_{rand}}(t) - \vec{A}.\vec{D} \qquad (3)$$

Where $\vec{A}$ coefficient vector $X^*$is the position vector of the best solution obtained so far, $\vec{X}$is the position vector

Algorithm: Pseudocode Whale Optimization algorithm:

```
Initialize the whales population Xi(i = 1, 2, ..., n)
Calculate the fitness of each search agent
X*=the best search agent
while(t < maximum number of iterations)
    for each search agent
        Update a, A, C, l, and p
        if1(p<0.5)
            if2(|A|< 1)
                Update the position of the current search agent by the
            else if2(|A|≥1)
                Select a random search agent ($X_{rand}$)
                Update the position of the current search agent by the Eq (2)
            end if2
        elseif1(p≥0.5)
            Update the position of the current search by the Eq. (3)
        end if1
    end for
    Check if any search agent goes beyond the search space and amend it
    Calculate the fitness of each search agent
    Update X* if there is a better solution
    t=t+1
end while
return X*
```

IV. Intelligent method

- **Artificial Neural Networks (ANNs)**

An ANN is a bio-inspired artificial model of a human brain capable of imitating learning based on behavior. A neuron has the ability to store and reproduce experiential information similar to the human brain is called the basic computational unit of an ANN. Due to their ability to produce simple and optimal solutions in complex situations to maintain the integrity of the specifications, these have been widely used in many search optimization, learning and pattern recognition problems [58]. Rath et al. [59] designed a crossbred controller using GA and ANN for the path planning of a humanoid robot.

- **Genetic Algorithm (GA):**

primarily developed by J. Holland in 1960 in Germany, Ingo Reichenberg [60]. The authors,[61] studied the effectiveness of two approaches, precisely the probabilistic roadmap (PRM) and the genetic algorithm (GA). To compare their performances, two maps, one simple, and one complex, were used. Lamini et al.[62] improved the crossover operator to figure out the path planning issue using GA. The simulation consequences showed that average iterations numbers and turns values are optimal but the authors did not give the value of the time and the path length. Utami et al. [63] utilized a mutated genetic algorithm method for collision-free paths planning with static obstacles. The consequence of the test showed that it still got a fast time for an ideal solution. The steps of the standard GA [64],[65] were itemized in Algorithm A, and Figure 10 depicts a flowchart of the Algorithm.

Algorithm A: Pseudocode genetic algorithm:

```
Input parameter: N: the size of the population; Pc: Crossover rate; Pm: Mutation rate.
Output: Best Chromosome.
t←0
Initialize arbitrarily the initial population P(t)
While (not termination condition) do
    Evaluate P(t) using a fitness function
    Select P(t) from P(t − 1)
    Rearrangement of P(t)
    Mutate P(t)
    Replace P(t − 1) by P(t)
    t←t+1
End
```

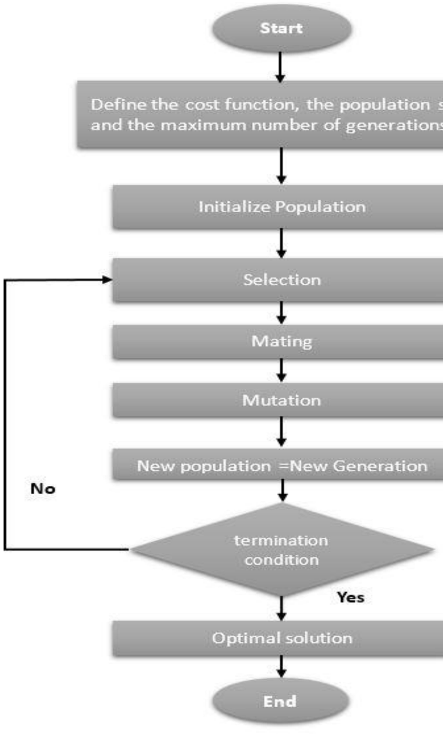
Start
Define the cost function, the population size, and the maximum number of generations
Initialize Population
Selection
Mating
Mutation
New population =New Generation
No
termination condition
Yes
Optimal solution
End

Figure 10: The flowchart of the Genetic algorithm

From the diagram above, the theoretical work steps are as follows:

1- The GA begins, like any other optimization algorithm, by defining the cost function, the size of the population, and the generation's maximum number.

2- -The initial pop of the GA is obtained randomly.

3- The selection operation is to locate which chromosomes participate in propagation to create the next population (next generation) according to the values of their fitness in the current population.
4- A mating operator is used to generate new chromosomes for the next offspring by randomly combining two chosen chromosomes from the previous step.
5- A mutation is a genetic factor that preserves the diversity of heredity from one generation of a genetic chromosome population to the next and reduces the chance of early convergence by randomly modifying chromosomes.
6- The new generation is the same as the new population of candidate solutions that will then be used in the next algorithm iteration.
7- A termination condition of the GA is influential in determining when the GA run will complete.
8- An optimal solution means the pest solution they found it then end the GA.

- **The fuzzy logic (FL)**

Lotfi Zadeh introduced the concept of fuzzy logic in 1965 is based on the idea that human thinking is not about crisp numbers, but about ideas instead. There are many vulnerability factors in the portrayal of the environment in the self-ruling mobile robot path arranging, it cannot easily group the situation to a particular situation, and a fuzzy logic concept is embraced in this type of situation,[66]. J. Guo, C. Li, and S. Guo [67] used fuzzy logic in a novel path planning technique. Firstly, the configuration and rectify of the ultrasonic sensors (HC-SR04) were done by analyzing the motion model of the spherical mobile robot. Then, a fuzzy controller for the spherical mobile robot was designed using multi-sensor fusion technology and the D-H parameter method. Finally, in an unknown environment, the proposed fuzzy control method was applied to the path planning of the spherical mobile robot. A novel variant of bio-inspired planning algorithms are presented in [68]. The method was implemented without prior information on the robot environment. Robot collision-free path planning in dynamic environments was proposed. In some difficult scenarios, such as the boundary map and the tight pathway map, the traditional neural dynamic model almost always returns a sub-optimal choice, the first contribution of mild technical analysis. Second, a topologically organized network with connections between neighboring neurons is the proposed planning algorithm, namely, the filling means neural dynamic model, and is suitable for pervasion nerve impulses such as waves without conjugation effects. There is a nice mix between FL and ACO by TaYen [69]The suggested the method of fuzzy ant colony optimization (FACO) to minimize the iterative learning error of the ant colony optimization (ACO) algorithm with the use of Fuzzy control. This algorithm locates the smallest path, and uses ultrasonic transducers to detect any obstructions ahead of the mobile robot, and adjusts the mobile robot's turning angle to avoid obstacles.

V. Randomized Methods:

There are two categories: single-question and multi-query randomized path planning. In a single-query, a single path planning problem must be resolved without any pre-processing, while in a multi-query, many path planning problems must be resolved in the same environment[70].

- **Rapidly Exploring Random Trees (RRTs):**

RRT algorithm is a good algorithm to solve the single-query task and probabilistic based which develop by Lavalle and Kuffner. It is a search tree used in real-time path planning, where it is either regrow The whole tree or only the previous iteration. The disadvantage of the RRT is a slow convergence rate which means that they consume a lot of memory and time to find the optimal path.[71],[72]. The authors [73] in 2020 , propose the NRRT* to achieve non-uniform sampling in the path planning process by learning quantities of successful planning cases from the A* algorithm. RRT algorithm is presented in Pseudocode down[74] ,and Figure 11 the a Schematic diagram of the RRT algorithm operation principle[72].

Algorithm: Pseudocode RRTs algorithm:

T=(V,E) ← RRT(Z_{init})

```
T ← $Iniailize\ tree()$
T ← $InsertNode(\emptyset, Z_{init}, T)$
For i=0 to i=N do
$Z_{rand} \leftarrow Sample(i)$
$Znearest \leftarrow Nearest\ (T, Z_{rand})$
$(Z_{new}, U_{new}) \leftarrow Steer(Z_{nearest}, Z_{rand})$
If obstacle free ($z_{new}$) then
T ← $InsertNode(Z_{nearest}, Z_{new}, T)$
Return T
```

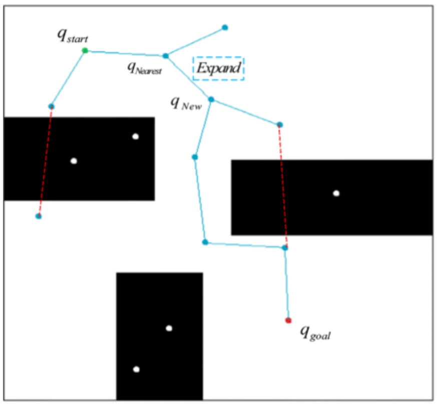

Figure 11:Schematic diagram of RRT algorithm operation principle [72]

- **Probabilistic Roadmap (PRM) Algorithm:**

This algorithm is used for multi-query tasks and used to solve the path planning problem in complex static environment, while sometimes used with simple dynamic environment. The main two elements in roadmap construction are edges and nodes, where edges are straight lines connecting nodes together. These edges are found by not powerful planner but very fast called local planners. On the other hand, nodes can be distributed using different techniques such as the normal distribution, medial axis, bridge test, …, etc. All these techniques have one common disadvantage which is neglecting the locations of both the start and the goal points. PRM can calculate all paths possible, but at the same time it cannot find the best possible path from these possibilities[75]. Because of the previous the author in [35] using Ant colony optimization and heuristic A* method can be combined with PRM as a path search method.

5. Conclusions

The issue of path planning is an important area of research. A mobile robot caught the attention of researchers. Good mobile robot path planning technology can not only memorize a great amount of time but also reduce mobile robot wear and principal development. In this paper, various path planning strategies are reviewed. The classical approaches had major drawback such as high temporal complexity in high dimensions and longer time to find the optimal pathway and they will be struck into the local minima which makes them inefficient in practice. The metaheuristic and heuristic algorithms have been developed to reduce the drawbacks of classical approaches and to fix the local minima problem. Each algorithm can be use depending on the application of mobile robots and the type of environment for example if we have known dynamic or static environment we can be using heuristic and meta-heuristic (D*, A* and PSO, ACO, WOA) to re-path and find the optimal path Consecutively.

ACKNOWLEDGMENT

I would like to express my deepest thanks to my supervisor Dr. Omar Farouq Lutfy and Dr.Qusay Fadhel Hasan Al-Doori for encouragement and useful discussion through the progress of this work. I would like to thank all the staff of Control and Systems Engineering Department for their efforts. Finally, all the words in the world are not enough to express my Acknowledgment to my family.

References

[1] X. Wang, X. Wang, and D. M. Wilkes, “An Automatic Scene Recognition Using TD-Learning for Mobile Robot Localization in an Outdoor Environment,” in *Machine Learning-based Natural Scene Recognition for Mobile Robot Localization in An Unknown Environment*, Springer, 2020, pp. 293–310.

[2] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, “Introduction to Autonomous Mobile Robots (Intelligent Robotics and Autonomous Agents series).” The MIT Press Cambridge, 2011.

[3] F. A. Raheem and U. I. Hameed, “Interactive heuristic D* path planning solution based on PSO for two-link robotic arm in dynamic environment,” *World J. Eng. Technol.*, vol. 7, no. 1, pp. 80–99, 2018.

[4] M. Virt, S. Siroki, Á. Nyerges, and V. Tihanyi, “An Online Path Planning Algorithm for Automated Vehicles for Slow Speed Maneuvering,” in *2019 International IEEE Conference and Workshop in Óbuda on Electrical and Power Engineering (CANDO-EPE)*, 2019, pp. 97–102.

[5] A. A. Hussain, “New Approaches for C-Space Map Construction for Heuristic Path Solutions Based on PSO Techniqe,” *Univ. Technol. Master Thesis*, 2017.

[6] A. Koubaa *et al.*, “Introduction to mobile robot path planning,” in *Robot Path Planning and Cooperation*, Springer, 2018, pp. 3–12.

[7] J. Van Den Berg, M. Stilman, J. Kuffner, M. Lin, and D. Manocha, “Path planning among movable obstacles: a probabilistically complete approach,” in *Algorithmic Foundation of Robotics VIII*, Springer, 2009, pp. 599–614.

[8] F. A. Kolushev and A. A. Bogdanov, “Multi-agent optimal path planning for mobile robots in environment with obstacles,” in *International Andrei Ershov Memorial Conference on Perspectives of System Informatics*, 1999, pp. 503–510.

[9] C. Rao and L. Yu, “A particular method for the shortest path finding of mobile robots,” in *2010 The 2nd International Conference on Industrial Mechatronics and Automation*, 2010, vol. 2, pp. 221–224.

[10] D. Amorim and R. Ventura, “Towards efficient path planning of a mobile robot on rough terrain,” in *2014 ieee international conference on autonomous robot systems and competitions (icarsc)*, 2014, pp. 22–27.

[11] S. Li, C.-M. Chew, and V. Subramaniam, “Smooth and efficient policy exploration for robot trajectory learning,” in *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2018, pp. 1087–1092.

[12] R. Uriol and A. Moran, “Mobile robot path planning in complex environments using ant colony optimization algorithm,” in *2017 3rd international conference on control, automation and robotics (ICCAR)*, 2017, pp. 15–21.

[13] Y. Xiu-xia, C. Wei-yi, Z. Yi, F. Guo-wei, and Y. Xuan, “Mobile robot path planning in complex environment,” in *2019 IEEE International Conference on Unmanned Systems (ICUS)*, 2019, pp. 426–431.

[14] Y. Xu, P. Liu, X. Zhang, C. Zha, and Z. Tian, “Formation Control and Obstacle Avoidance for Multi-agent Systems in Unknown Environment,” in *2019 IEEE International Conference on Unmanned Systems (ICUS)*, 2019, pp. 925–930.

[15] N. Sun, E. Yang, J. Corney, Y. Chen, and Z. Ma, “Semantic enhanced navigation among movable obstacles in the home environment,” in *UK-RAS19 Conference*, 2019, pp. 68–71.

[16] M. M. Badr, “Development of a Robot Path Planning Algorithm Based on a Modified Artificial Potential Field,” *Univ. Technol. Master Theses*, 2017.

[17] K. H. Sedighi, K. Ashenayi, T. W. Manikas, R. L. Wainwright, and H.-M. Tai, “Autonomous local path planning for a mobile robot using a genetic algorithm,” in *Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753)*, 2004, vol. 2, pp. 1338–1345.

[18] A. T. Ismail, A. Sheta, and M. Al-Weshah, “A mobile robot path planning using genetic algorithm in static environment,” *J. Comput. Sci.*, vol. 4, no. 4, pp. 341–344, 2008.

[19] M. Dirik, A. F. Kocamaz, and O. Castillo, “Global Path Planning and Path-Following for Wheeled Mobile Robot Using a Novel Control Structure Based on a Vision Sensor,” *Int. J. Fuzzy Syst.*, pp. 1–12, 2020.

[20] S. H. Shwail and A. Karim, “Probabilistic roadmap, A*, and GA for proposed decoupled multi-robot path planning,” *Iraqi J. Appl. Phys.*, vol. 10, no. 2, 2014.

[21] A. and D. P. “Review of classical and heuristic-based navigation and path planning approaches. . (2013). Atyabi, “Review of classical and heuristic-based navigation and path planning approaches,” *Int. J. Adv. Comput. Technol.*, 2013.

[22] M. Gnanaprakash, “Study on Mobile Robot Path Planning–A Review,” *Int. J. Appl. Eng. Res.*, vol. 10, no. 57, p. 2015.

[23] F. Samaniego, J. Sanchis, S. García-Nieto, and R. Simarro, “Recursive Rewarding Modified Adaptive

Cell Decomposition (RR-MACD): A Dynamic Path Planning Algorithm for UAVs," *Electronics*, vol. 8, no. 3, p. 306, 2019.

[24] F. A. Raheem and M. M. Badr, "Development of Modified path planning algorithm using artificial potential field (APF) based on PSO for factors optimization," *Am. Sci. Res. J. Eng. Technol. Sci.*, vol. 37, no. 1, pp. 316–328, 2017.

[25] F. Bayat, S. Najafinia, and M. Aliyari, "Mobile robots path planning: Electrostatic potential field approach," *Expert Syst. Appl.*, vol. 100, pp. 68–78, 2018.

[26] Q.-L. Xu, T. Yu, and J. Bai, "The mobile robot path planning with motion constraints based on Bug algorithm," in *2017 Chinese Automation Congress (CAC)*, 2017, pp. 2348–2352.

[27] S. K. Das, K. Roy, T. Pandey, A. Kumar, A. K. Dutta, and S. K. Debnath, "Modified Critical Point–A Bug Algorithm for Path Planning and Obstacle Avoiding of Mobile Robot," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, 2020, pp. 351–356.

[28] W. Chen, N. Wang, X. Liu, and C. Yang, "VFH* Based Local Path Planning for Mobile Robot," in *2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)*, 2019, pp. 18–23.

[29] S. Campbell, N. O'Mahony, A. Carvalho, L. Krpalkova, D. Riordan, and J. Walsh, "Path Planning Techniques for Mobile Robots A Review," in *2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)*, 2020, pp. 12–16.

[30] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous robot vehicles*, Springer, 1986, pp. 396–404.

[31] B. K. Patle, L. B. Ganesh, A. Pandey, D. R. K. Parhi, and A. Jagadeesh, "A review: on path planning strategies for navigation of mobile robot. Defence Technol 15 (4): 582–606." Elsevier, pp. 582–606, 2019.

[32] M. A. H. Ali and I. H. Shanono, "Path Planning Methods for Mobile Robots: A systematic and Bibliometric review," *Elektr. Electr. Eng.*, vol. 19, no. 3, pp. 14–34, 2020.

[33] M. Elbanhawi, M. Simic, and R. Jazar, "Autonomous robots path planning: An adaptive roadmap approach," in *Applied Mechanics and Materials*, 2013, vol. 373, pp. 246–254.

[34] Z. Zeng, W. Sun, W. Wu, M. Xue, and L. Qian, "An Efficient Path Planning Algorithm for Mobile Robots," in *2019 IEEE 15th International Conference on Control and Automation (ICCA)*, 2019, pp. 487–493.

[35] F. A. RAHEEM and M. I. ABDULKAREEM, "DEVELOPMENT OF A* ALGORITHM FOR ROBOT PATH PLANNING BASED ON MODIFIED PROBABILISTIC ROADMAP AND ARTIFICIAL POTENTIAL FIELD," *J. Eng. Sci. Technol.*, vol. 15, no. 5, pp. 3034–3054, 2020.

[36] A. H. Hasan and A. M. Mosa, "Multi-robot path planning based on max–min ant colony optimization and D* algorithms in a dynamic environment," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 110–115.

[37] H. Zhang, W. Lin, and A. Chen, "Path planning for the mobile robot: A review," *Symmetry (Basel).*, vol. 10, no. 10, p. 450, 2018.

[38] M. N. S. T. A Badmos, P. O Omolaye, S. O Adio, "Path planning techniques for mobile robot : A review," *IOSR J. Comput. Eng.*, vol. 20, no. 6, 2018.

[39] N. Gunantara and I. Nurweda Putra, "The characteristics of metaheuristic method in selection of path pairs on multicriteria ad hoc networks," *J. Comput. Networks Commun.*, vol. 2019, 2019.

[40] M. Saraswathi, G. B. Murali, and B. Deepak, "Optimal path planning of mobile robot using hybrid cuckoo search-bat algorithm," *Procedia Comput. Sci.*, vol. 133, pp. 510–517, 2018.

[41] N. Setyawan, R. E. A. Kadir, and A. Jazidie, "Adaptive Gaussian parameter particle swarm optimization and its implementation in mobile robot path planning," in *2017 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2017, pp. 238–243.

[42] H. S. Dewang, P. K. Mohanty, and S. Kundu, "A robust path planning for mobile robot using smart particle swarm optimization," *Procedia Comput. Sci.*, vol. 133, pp. 290–297, 2018.

[43] E. Krell, A. Sheta, A. P. R. Balasubramanian, and S. A. King, "Collision-free autonomous robot navigation in unknown environments utilizing pso for path planning," *J. Artif. Intell. Soft Comput. Res.*, vol. 9, no. 4, pp. 267–282, 2019.

[44] B. Tang, K. Xiang, M. Pang, and Z. Zhanxia, "Multi-robot path planning using an improved self-adaptive particle swarm optimization," *Int. J. Adv. Robot. Syst.*, vol. 17, no. 5, p. 1729881420936154, 2020.

[45] U. Rajput and M. Kumari, "Mobile robot path planning with modified ant colony optimisation," *Int. J. Bio-Inspired Comput.*, vol. 9, no. 2, pp. 106–113, 2017.

[46] T. Wang, L. Zhao, Y. Jia, and J. Wang, "Robot path planning based on improved ant colony algorithm," in *2018 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, 2018, pp. 70–76.

[47] Q. Luo, H. Wang, Y. Zheng, and J. He, "Research on path planning of mobile robot based on improved ant colony algorithm," *Neural Comput. Appl.*, vol. 32, no. 6, pp. 1555–1566, 2020.

[48] X. Pu, C. Xiong, L. Ji, and L. Zhao, "3D path planning for a robot based on improved ant colony algorithm," *Evol. Intell.*, pp. 1–11, 2020.

[49] A. Chhillar and A. Choudhary, "Mobile robot path planning based upon updated whale optimization algorithm," in *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 2020, pp. 684–691.

[50] A. H. Gandomi, X.-S. Yang, S. Talatahari, and A. H. Alavi, *Metaheuristic applications in structures and infrastructures*. Newnes, 2013.

[51] S. S. Rao, *Engineering optimization: theory and practice*. John Wiley & Sons, 2019.

[52] H.-P. Dai, D.-D. Chen, and Z.-S. Zheng, "Effects of random values for particle swarm optimization algorithm," *Algorithms*, vol. 11, no. 2, p. 23, 2018.

[53] H. Zhang, W. Lin, and A. Chen, "Path planning for the mobile robot: A review," *Symmetry (Basel).*, vol. 10, no. 10, p. 450, 2018.

[54] M. N. Zafar and J. C. Mohanta, "Methodology for path planning and optimization of mobile robots: A review," *Procedia Comput. Sci.*, vol. 133, pp. 141–152, 2018.

[55] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016.

[56] N. Rana, M. S. Abd Latiff, and H. Chiroma, "Whale optimization algorithm: a systematic review of contemporary applications, modifications and developments," *Neural Comput. Appl.*, pp. 1–33, 2020.

[57] A. M. Mosaad, M. A. Attia, and A. Y. Abdelaziz, "Whale optimization algorithm to tune PID and PIDA controllers on AVR system," *Ain Shams Eng. J.*, vol. 10, no. 4, pp. 755–767, 2019.

[58] A. S. H. H. V Injarapu and S. K. Gawre, "A survey of autonomous mobile robot path planning approaches," in *2017 International conference on recent innovations in signal processing and embedded systems (RISE)*, 2017, pp. 624–628.

[59] A. K. Rath, D. R. Parhi, H. C. Das, P. B. Kumar, and M. K. Mahto, "Design of a hybrid controller using genetic algorithm and neural network for path planning of a humanoid robot," *Int. J. Intell. Unmanned Syst.*, 2020.

[60] J. H. Holland, "Genetic algorithms," *Sci. Am.*, vol. 267, no. 1, pp. 66–73, 1992.

[61] 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," in *2017IEEE 9th international conference on humanoid, nanotechnology, information technology, communication and control, environment and management (HNICEM)*, 2017, pp. 1–5.

[62] C. Lamini, S. Benhlima, and A. Elbekri, "Genetic algorithm based approach for autonomous mobile robot path planning," *Procedia Comput. Sci.*, vol. 127, pp. 180–189, 2018.

[63] N. S. Utami, A. Jazidie, and R. E. A. Kadier, "Path Planning for Differential Drive Mobile Robot to Avoid Static Obstacles Collision using Modified Crossover Genetic Algorithm," in *2019 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2019, pp. 282–287.

[64] F. H.-F. Leung, H.-K. Lam, S.-H. Ling, and P. K.-S. Tam, "Tuning of the structure and parameters of a neural network using an improved genetic algorithm," *IEEE Trans. Neural networks*, vol. 14, no. 1, pp. 79–88, 2003.

[65] S. Baluja and R. Caruana, "Removing the genetics from the standard genetic algorithm," in *Machine Learning Proceedings 1995*, Elsevier, 1995, pp. 38–46.

[66] A. Pandey, S. Pandey, and D. R. Parhi, "Mobile robot navigation and obstacle avoidance techniques: A review," *Int Rob Auto J*, vol. 2, no. 3, p. 22, 2017.

[67] J. Guo, C. Li, and S. Guo, "A Novel Step Optimal Path Planning Algorithm for the Spherical Mobile Robot Based on Fuzzy Control," *IEEE Access*, vol. 8, pp. 1394–1405, 2019.

[68] Y. Chen, J. Liang, Y. Wang, Q. Pan, J. Tan, and J. Mao, "Autonomous mobile robot path planning in unknown dynamic environments using neural dynamics," *Soft Comput.*, pp. 1–17, 2020.

[69] C.-T. Yen and M.-F. Cheng, "A study of fuzzy control with ant colony algorithm used in mobile robot for shortest path planning and obstacle avoidance," *Microsyst. Technol.*, vol. 24, no. 1, pp. 125–135, 2018.

[70] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, 2000, vol. 2, pp. 995–1001.

[71] M. Korkmaz and A. Durdu, "Comparison of optimal path planning algorithms," in *2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET)*, 2018, pp. 255–258.

[72] X. Xia, H. Zhu, Z. Zhang, X. Liu, L. Wang, and J. Cao, "3D-based multi-objective cooperative disassembly sequence planning method for remanufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 106, no. 9, pp. 4611–4622, 2020.

[73] J. Wang, W. Chi, C. Li, C. Wang, and M. Q.-H. Meng, "Neural RRT*: Learning-based optimal path planning," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1748–1758, 2020.

[74] I. Noreen, A. Khan, and Z. Habib, "A comparison of RRT, RRT* and RRT*-smart path planning algorithms," *Int. J. Comput. Sci. Netw. Secur.*, vol. 16, no. 10, p. 20, 2016.

[75] C. Zhou, B. Huang, and P. Fränti, "A survey of motion planning algorithms for intelligent robotics," *arXiv Prepr. arXiv2102.02376*, 2021.