IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Journal Européen des Systèmes Automatisés
Vol. 55, No. 3, June, 2022, pp. 405-412
Journal homepage: http://iieta.org/journals/jesa

Online Path Planning of Mobile Robots Based on African Vultures Optimization Algorithm in Unknown Environments

Mustafa S. Abed*, Omar F. Lutfy, Qusay F. Al-Doori

Control and Systems Engineering Department, University of Technology-Iraq, Baghdad 10002, Iraq

Corresponding Author Email: cse.19.07@grad.uotechnology.edu.iq

https://doi.org/10.18280/jesa.550313

Received: 15 March 2022
Accepted: 20 April 2022

Keywords:
APSO, AVOA, path planning, mobile robot, obstacle avoidance

ABSTRACT

Autonomous mobile robots developed using metaheuristic algorithms are increasingly becoming a hot topic in control and computer sciences. Specifically, finding the shortest root to the goal and avoiding hurdles are current subjects of autonomous mobile robots. The main drawbacks of classic methods are the incapacity to move the robot in a dynamic and unknown environment, deadlock in a local minimum and complicated environments, and incapacity to foretell the speed vector of obstacles and non-optimality of the route. This article exhibits a recent path planning approach that utilizes the African Vultures Optimization (AVOA) for navigation of the mobile robot in static and dynamic unknown environments with a dynamic target. The proposed online optimization approach is used in three different environments including an environment with unknown static obstacles, an environment with unknown dynamic obstacles, and an environment with a dynamic target. The proposed approach can solve a local minima problem in the environment with static obstacles. The online optimization method is performed using two phases which are the sensors' reading phase and the path calculation phase and the results are given based on computer simulation in different unknown environments. A comparative study was conducted between the suggested algorithm and two other algorithms and the results showed that the AVOA algorithm was better in avoiding obstacles successfully including the local minima situation. Finally, the average enhancement rates in the path length compared with the Adaptive Particle Swarm Optimization (APSO) and the Hybrid Fuzzy-Wind Driven Optimization (WDO) are 2.21% and 1.02207%, respectively.

1. INTRODUCTION

Recently, the motion planning issue of a mobile robot is an important topic of study in the area of mobile robots in current research [1, 2]. Route planning is a process to obtain a sensible and collision-free path between the beginning point and the destination, where the need to plan the route becomes important for a fully or partly automated process [3, 4]. Generally, planning the movement of a mobile robot in an unfamiliar environment has been divided into three categories. The first one is based on information about the obstacles that might be known or unknown [5]. The second category is according to time, including online and offline planning [6]. The third category depends on the nature of obstacles inside the environment, where the obstacles might be stationary or moving [7, 8]. Many researchers have used different approaches in the navigation of Differential Wheel Mobile Robot (DWMR). For instance, Wang et al. [9] utilized a dynamic pathway designing method for mobile robots based on an improved genetic algorithm in an unknown environment, using a serial number coding method and chromosome length shortening. However, the drawback of this method is that the time taken for processing was high. Akka and Farid [10] compared the performance of a modified ant colony algorithm and the classic ant colony algorithm in grid maps for optimal static navigation paths. This improves the algorithm's convergence speed and search efficiency, but the drawback is a lack of initial pheromone and a weakening of guidance in a complex environment. Rao et al. [11] employed a hybrid method of Intelligent Water Droplet (IWD) and Differential Evolution (DE) technology for multi-robot navigation in static and dynamic environments. The authors indicated that there was a problem with deadlock situations for the robots and that this algorithm needs to be improved to avoid this problem. Chhillar and Choudhary [12] improved the whale optimization algorithm, which was used to find a collision-free path. The proposed process has been tested in various search spaces. However, the algorithm does not guarantee a collision-free path in crowded environments. Oleiwi et al. [13] applied fuzzy logic control for collision avoidance using only dynamic obstacles in partially unknown and known environments and they used the A-star algorithm to find the route in an offline manner. The drawback of this method is that it cannot work in fully unknown or maze environments. In another work in Ref. [14], the authors proposed an intelligent Adaptive Particle Swarm Optimization (APSO) algorithm for robot path planning in uncertain environments. The drawback of this method is that it needs to be modified to work in dynamic or multi-robot systems and to avoid the early convergence in local minima.

The main drawbacks to previous studies of the methods are the inability to plan movement in a dynamic and unknown environment, failure in crowded and complex environments, the unpredictability of the velocity vector of obstacles, and non-optimization of the complex path.

From previous studies, it was seen that many optimization methods produce low-quality solutions or suffer from an early convergence in the primary stages. These issues are observed in PSO [14], Ant colony [7], GWO [15]. etc., therefore there is a need to study and improve these methods to overcome their limitations.

This article exhibits a recent path planning approach that utilizes the African Vultures Optimization Algorithm (AVOA) for navigation of the mobile robot in static and dynamic unknown environments with a dynamic target. The proposed online optimization approach is used in three different environments including an environment with unknown static obstacles, unknown dynamic obstacles, and a dynamic target. The proposed approach can solve a local minima problem in the environment with static obstacles and has fast convergence in finding the shortest path taking less time compared with other methods.

2. PROBLEM STATEMENT

Finding the best route between the starting position of the robot and the goal is one of the most important topics in the field of search and rescue applications that use robots or quadcopters. In this work, the African Vultures Optimization Algorithm (AVOA) is employed in an online procedure to adaptively handle three different environments that include unknown static obstacles, unknown dynamic obstacles, and a dynamic target.

3. METHODOLOGY AND MODELLING

In this section, the methodology of the work and the modeling of environments are described considering a two-dimensional (X, Y) square map. In particular, the modeling stage includes the following parts.

3.1 Obstacles

This section describes the architecture of the obstacles. There are N obstacles in the environment, $O_1, O_2, \ldots, O_N$. The obstacles' coordinates are shown as (X_{O1}, Y_{O1}), (X_{O2}, Y_{O2}), ..., (X_{ein}, Y_{ein}) and this presentation includes static and dynamic obstacles. In particular, when the obstacle is stationary, its speed is zero, and when it moves, its velocity (v) is along the X and Y axes. Each obstacle's speed is determined at random and can be greater than, less than, or equal to the speed of the robot. The robot is unaware of the obstacle vectors' speed and position (velocity and direction).

3.2 Robot and sensing

In general, in the case of local path planning problems, the robot gradually tries to set the next collision-free position in an indefinite environment from the start location to the end location in steps. The robot location which is used in this work is based on a relative location, where each robot calculates its next place based on its current position, corner, and speed over an interval of time, as shown in Figure 1. In particular, the next position of the robot is formulated in Eqns. (1) and (2) [16-18].

$$X_{next} = X_{current} + Vln * Cos\ (\theta_i) \tag{1}$$

$$Y_{next} = Y_{current} + Vln * sin\ (\theta_i), \tag{2}$$

where, $X_{current}\ and\ Y_{current}$ are the coordinates of the current position and $X_{next}\ and\ Y_{next}$ are the coordinates of the next position to the nth robot in the Cartesian coordinate system. Furthermore, the speed of the robot is specified by *Vln* and θ_i that indicates the corner to direct the nth robot.

The robot, which recognizes the obstacle within its range, determines the space between the obstacle and the robot and estimates the direction of the moving obstacle [19, 20]. If the position of the obstacle does not change, the obstacle is static. Otherwise, the obstacle is dynamic. Figure 2 shows the path-finding procedure of the mobile robot.

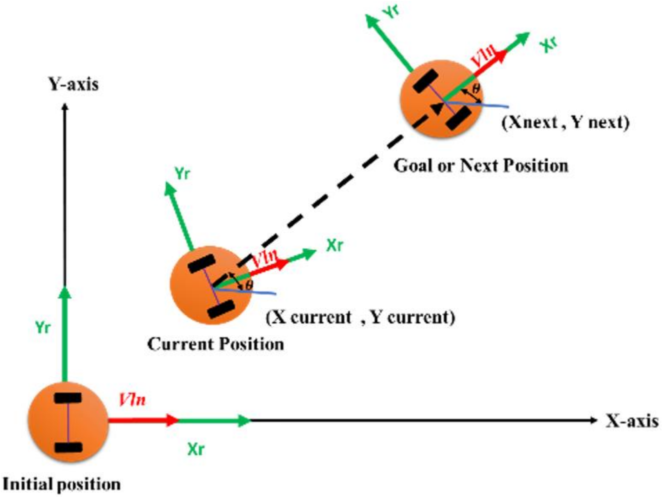

Figure 1. The current and the next positions of the mobile robot

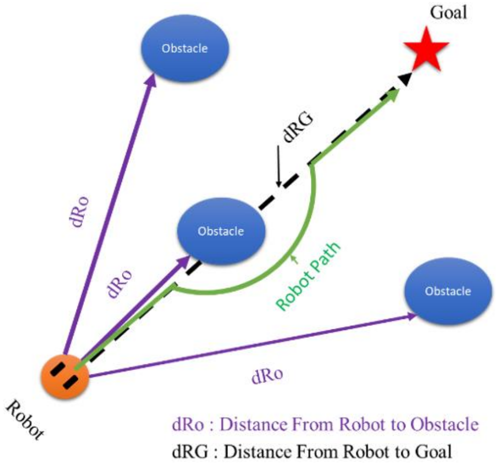
Goal
Obstacle
dRG
dRo
Obstacle
dRo
Robot Path
Obstacle
dRo
Robot
dRo : Distance From Robot to Obstacle
dRG : Distance From Robot to Goal

Figure 2. Path planning model of the mobile robot

3.3 Target- pursuing behavior

This section explains how the mobile robot detects the position of the goal, as illustrated in Figure 3. We can calculate the minimum distance between the robot and the goal (dRG) which is defined by Eq. (3). Specifically, if the robot reaches the target, this means the distance (dRG) is equal to or near to zero, in which case the robot must be stopped [14].

$$dRG = \sqrt{(Xg - Xr)^2 \quad + \quad (Yg - Yr)^2} \tag{3}$$

where, Xg and Yg are the coordinates of the goal in the environment, and Xr and Yr are the current robot's coordinates in the environment.

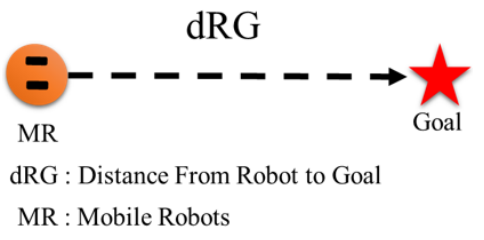

Figure 3. Distance between the robot and the goal

3.4 Obstacle- pursuing behavior

When an obstacle is found using the sensors, as depicted in Figure 4, the online optimization method can calculate the maximum distance between the robot and the goal (dRo), which is defined by Eq. (4). Particularly, if the robot detects an obstacle, it tries to avoid it and then continues to reach the target [14].

$$dRo = \sqrt{(Xo - Xr)^2 \quad + \quad (Yo - Yr)^2} \tag{4}$$

where, Xo and Yo are the obstacle's coordinates in the environment, and *Xr* and *Yr* are the coordinates of the current robot's position in the environment.

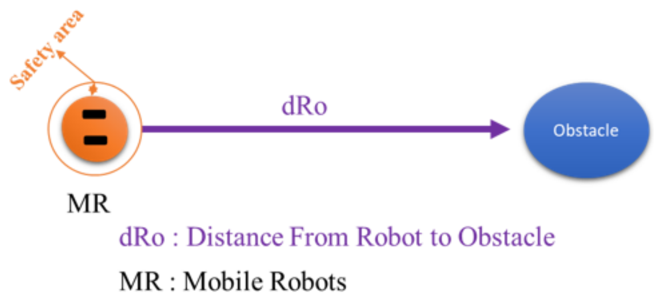

Figure 4. Distance between the robot and the obstacle

3.5 Objective function (cost function)

For robot safety reasons, the space between the robot and the obstacle should be maximum, but to achieve the shortest route, the space from the robot to the target should be minimized, which is based on the two perspectives, the cost function needs to guarantee the optimal route. In particular, the cost function of Eq. (5) is utilized in this thesis [14]:

$$\text{Cost Function}=[\alpha_1 \times dRG + \alpha_2 \div dRO + \alpha_3 \times \theta] \tag{5}$$

Here, the parameter α_1 is set so that the robot covers the minimum distance and reaches the target point as possible [14]. Similarly, α_2 controls the space between the robot and obstacles [14]. α_3 is the path smoothing parameter to avoid sharp turning [14].

dRO and dRG are used to calculate the space between the robot and the obstacle, and the space between the robot and the target point, respectively using the Euclidean distance formula (4) and (5). Moreover, θ in the corner of the difference is required by the robot to detect the next iteration location in the environment. Specifically, θ is given by Eq. (6).

$$\theta = \tan^{-1} \frac{Yg - Yr}{Xg - Xr} \tag{6}$$

4. THE PROPOSED PATH PLANNING ALGORITHMS

This section explained the detail of the African Vultures Optimization Algorithm (AVOA). The AVOA is a news algorithm presented by Abdollahzadeh et al. [16] in 2021. To explain the works: Vulture refers to two types of hunting birds. The majority of vultures have no regular feathers and are bald. Vultures do not exist in Australia or Antarctica. Unlike most other birds, they do not build nests and prefer to live on the ground [16]. The algorithm work by calculating the suitability function of all initial population to be divided the best solution is chosen as the best and first vulture, while the second solution is chosen as the second-best vulture. Another form of the population is either replacing or moving one of the top two vultures in each iteration. The standard AVOA is described below with necessary conditions and points [16].

STEP ONE: DETERMINING THE BEST VULTURE

The best solution is chosen by using Eq. (7), and After each capability repetition, the population is recalculated [16].

$$C(i) = \begin{cases} BestVulture1 if\ pi = L1 \\ BestVulture2 if\ pi = L2 \end{cases} \tag{7}$$

where, L1 and L2 are the parameters that have to be measured before the search operation and have values between (0 and 1) and the summation of both parameters is 1.

STEP TWO: THE RATE OF STARVATION OF VULTURES

Eq. (8) is used to model this behavior mathematically, which shows that vultures become aggressive towards one another if they are hungry or have less energy to fly for long distances [16].

$$Fo = (2 * rand1 + 1) * z * \left(1 - \frac{iterationi}{maxiterations}\right) + t \tag{8}$$

where, ***F*** indicates to the vultures are satiated, ***iteration*** is the current repetition number, ***max iterations*** are the total number of repetitions, and ***z*** is a random number between -1 and 1. ***rand*₁** is a variable with a value ranging from 0 to 1.

STEP THREE: EXPLORATION

In the exploration phase, it is done by the Eqns. (9)-(12) [16].

$$Po(i+1) = \begin{cases} equation\ (10) if\ Po1 \geq \text{randPo1} \\ equation\ (12) if\ Po1 < \text{randPo1} \end{cases} \tag{9}$$

$$Po(i+1) = C(i) - D(i) * Fo \tag{10}$$

$$D(i) = |X * C(i) - Po(i)| \tag{11}$$

$$\begin{aligned} Po(i+1) = C(i) - Fo + rand2 * ((ub - lb) \\ * rand3 + lb) \end{aligned} \tag{12}$$

where, ***Po*(*i*+1)** is the position vulture in the next repetition, and ***F***o is the saturation rate of the vulture calculated using Eq. (8). *C(i)* is one of the best vultures, and ***X*** is where the vultures are moving randomly to protect food from other vultures by using the formula *X* = 2× *rand*, and the random value in ***rand*, randp1, rand2, rand3** is a random number between (0) and (1). *Po*(*i*) is the existing vector position of the vulture. *lb* and *ub* demonstrate the higher and the lesser bound of the variables [16].

STEP FOUR: EXPLOITATION (FIRST PHASE)

The AVOA is entering the first stage in the utilization stage when the value |Fo| is in the middle of (1) and (0.5).

$$Po(i+1) = \begin{cases} equation\ (14)\ if\ Po2 \geq \text{randPo2} \\ equation\ (16)\ if\ Po2 < \text{randPo2} \end{cases} \tag{13}$$

$$Po(i+1)=D(i)*(F+rand4)-d(t) \tag{14}$$

$$d(t)=C(i)-Po(i) \tag{15}$$

$$Po(i+1)=C(i)-(S1+S2) \tag{16}$$

$$\left.\begin{aligned} S1 = C(i) * ((rand5 + Po(i))/2\pi) * cos\ (Po(i)) \\ S2 = C(i) * ((rand6 + Po(i))/2\pi) * sin\ (Po(i)) \end{aligned}\right\} \tag{17}$$

where, *rand*4 is a random number between 0 and 1.

EXPLOITATION: (SECOND PHASE)

The AVOA enters the second stage in the utilization stage when the value |Fo| is smaller than 0.5 [16].

$$Po(i+1) = \begin{cases} equation\ (19)\ if\ Po3 \geq \text{randPo3} \\ equation\ (21)\ if\ Po3 < \text{randPo3} \end{cases} \tag{18}$$

$$Po(i+1) = \frac{AS1 + AS2}{2} \tag{19}$$

$$\left.\begin{aligned} AS1 = BestVulture1(i) - \frac{BV1(i) * Po(i)}{BV1(i) - Po(i)} * Fo \\ AS2 = BestVulture2(i) - \frac{BV2(i) * Po(i)}{BV2(i) - Po(i)} * Fo \end{aligned}\right\} \tag{20}$$

$$Po(i+1) = C(i) - |d(t)| * Fo * Levy(d) \tag{21}$$

***BV*1(i) and *BV*2(i)** are the best vultures of the first and second groups in the current iteration. ***AS*1 and *AS*2** are The accumulation of several. *d*(*t*) is representing the space between the vulture and one of the best vultures in the two groups. **Levy(d)** was the mean Levy Flight (LF) calculated using Eq. (22) [16].

$$\text{Levy(d)=0.01} * \frac{r2*\sigma}{|r1|^{\frac{1}{\beta}}},\ \sigma = \left(\frac{\lceil(1+\beta)*\sin\left(\frac{\pi\beta}{2}\right)}{\lceil(1+\beta2)*\beta*2\left(\frac{\beta-1}{2}\right)}\right)^{\frac{1}{\beta}} \tag{22}$$

r1 and **r2** are random numbers in the middle of (0) and (1), and **β** is a fixed and default number of (1.5).

The algorithmic flowchart of the proposed algorithm navigation algorithm in an unknown environment is illustrated in Figure 5.

In this context, the steps involved in the AVOA for mobile robot navigation are explained below:

1. Build the robot, target, and obstacle position.
2. If the sensor does not detect any obstacles in the detection area, the robot will move directly to the target position, otherwise, the proposed AVOA algorithm starts working to avoid obstacles, and calculate a new trajectory toward the target.
3. If the robot reaches a target, the program stops and calculates the distance and time taken, otherwise, the program decides that the target is a dynamic one, and the algorithm should search for it.
4. The best trajectory calculated for the robot is based on Eq. (5).
5. Repeat steps 2 to 4 until the robot avoids all the obstacles and reaches its target.

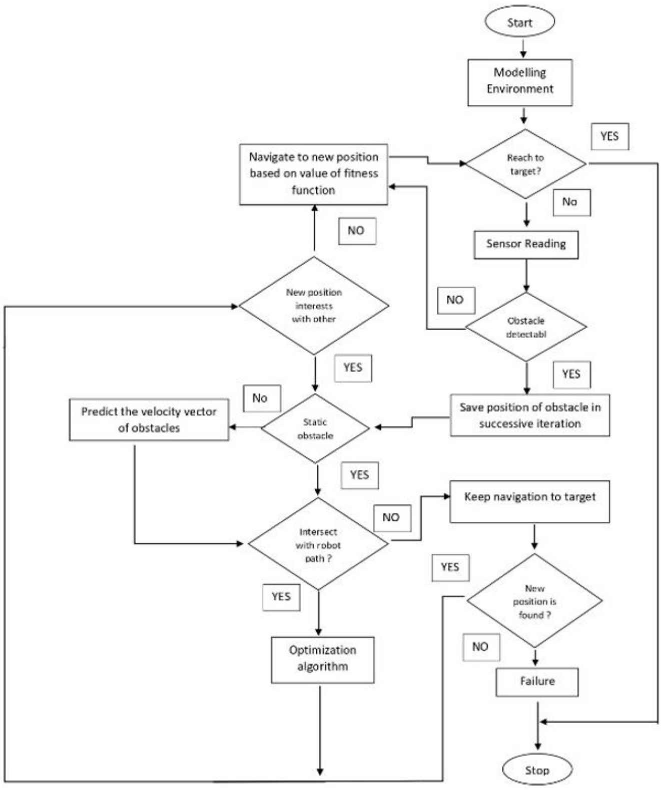
Start
Modelling Environment
Reach to target?
YES
No
Navigate to new position based on value of fitness function
NO
Sensor Reading
NO
Obstacle detectabl
New position interests with other
YES
YES
Save position of obstacle in successive iteration
No
Predict the velocity vector of obstacles
Static obstacle
YES
Keep navigation to target
NO
Intersect with robot path ?
YES
YES
New position is found ?
NO
Optimization algorithm
Failure
Stop

Figure 5. The flowchart of the proposed navigation algorithm in an unknown environment

5. SIMULATION RESULTS AND DISCUSSIONS

Using MATLAB R2016 b, the simulation tests were performed on PC, Windows 10 OS, Intel(R) Core (TM) i7-8550 U processor, 1.80 GHz CPU, and 20 GB RAM. The simulation parameters considered for the proposed algorithm are listed in Table 1. In particular, these parameters were selected to obtain the best optimization performance and they were sufficient in achieving the desired path planning objective.

Table 1. Parameter settings for the simulation

Parameters used in the simulation	
Number of search agents	10
Maximum number of iterations	10
Dimension	1
Lower bound	-5
Upper bound	5

For more tests on the performance of the AVOA algorithm on specific problems, we use three cases to test without obstacle, and (obstacle static, dynamic) unknown environment, the last case for the dynamic target. Comparing the proposed method with previous works. The simulation results for the static unknown map are summarized in Table 1, and the Enhancement rates in the path lengths of the objective function for the utilized techniques are explained in Table 2.

Table 3 illustrates the enhancement rates in the path length for the modified algorithms compared to the APSO and Hybrid Fuzzy-WDO.

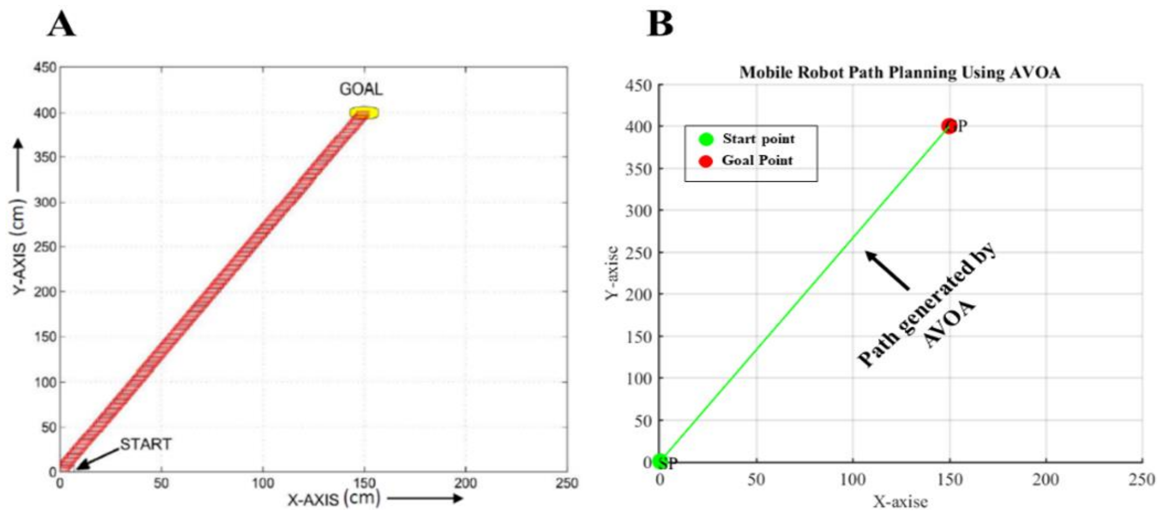

Figure 6. Map (1) simulation graph robot path planning with a starting point at (0,0) and an endpoint at (400,150): (A) Fuzzy-Wind Driven Optimization [21] Vs AVOA

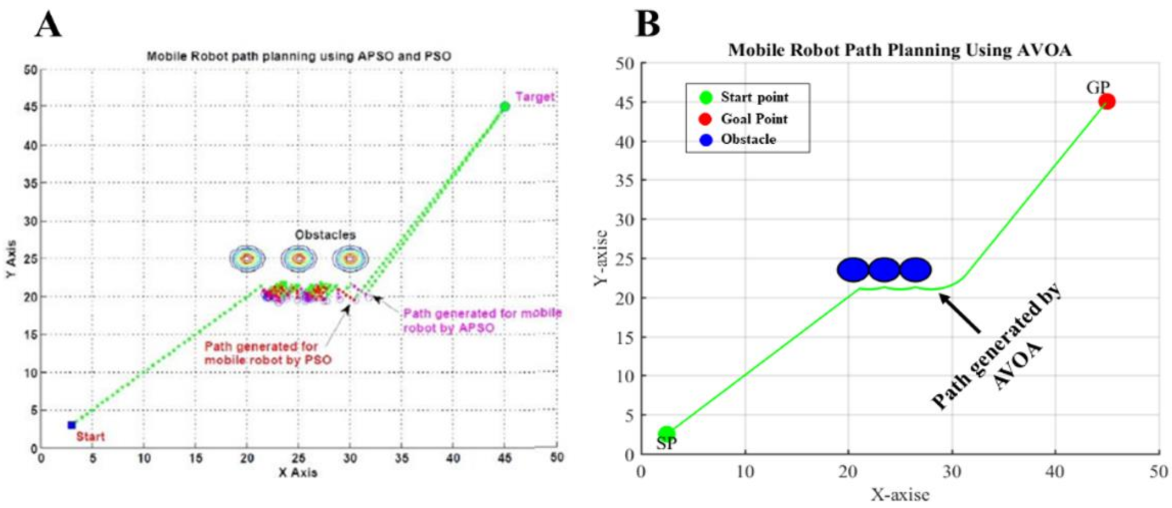

Figure 7. Map (2) simulation graph robot path planning with a starting point at (0,0) and an endpoint at (45,45): (A) APSO [14] Vs AVOA

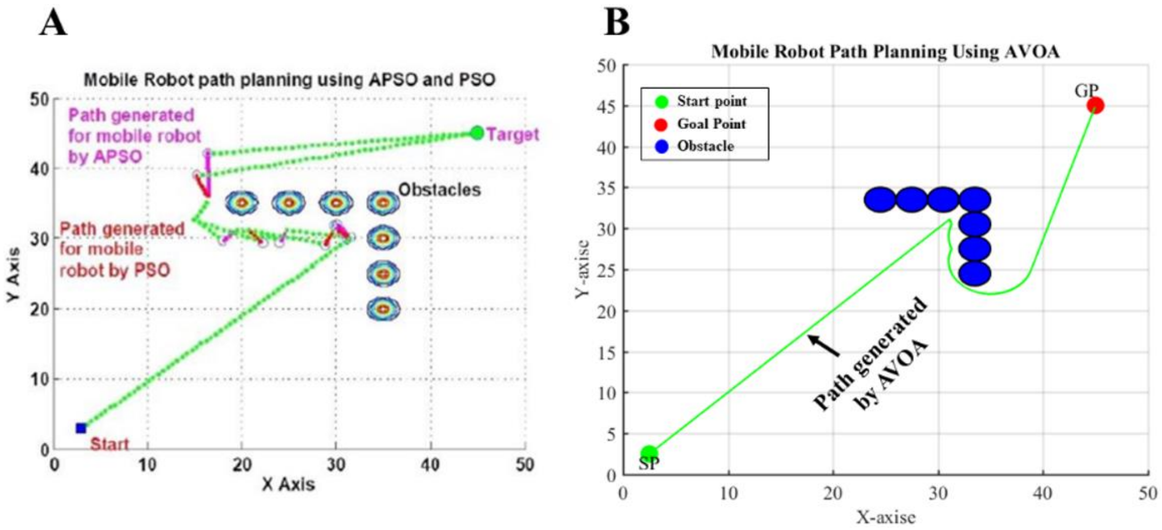

Figure 8. Map (3) simulation graph robot path planning with a starting point at (0,0) and an endpoint at (45,45): (A) APSO [14] Vs AVOA

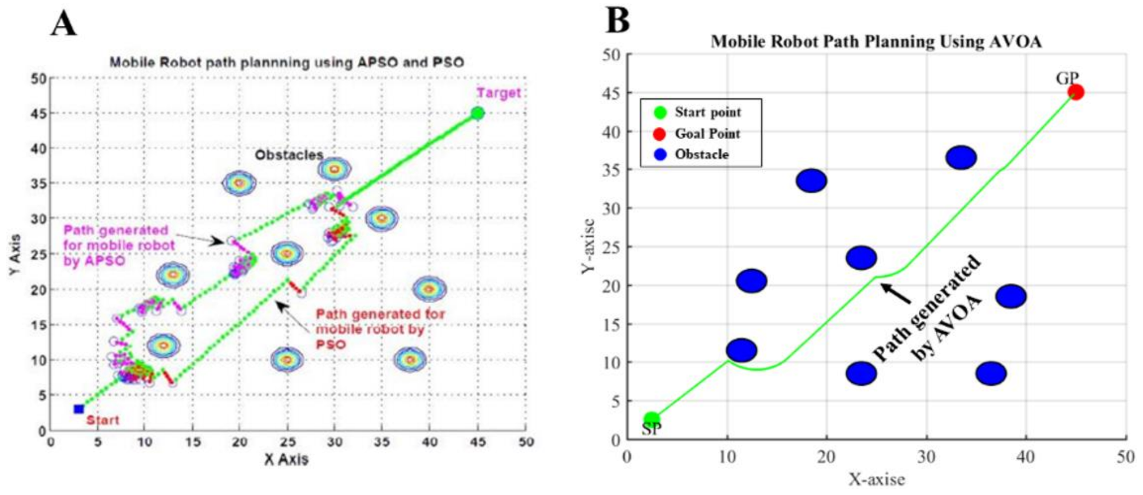

Figure 9. Map (4) simulation graph robot path planning with a starting point at (0,0) and an endpoint at (45,45): (A) APSO [14] Vs AVOA

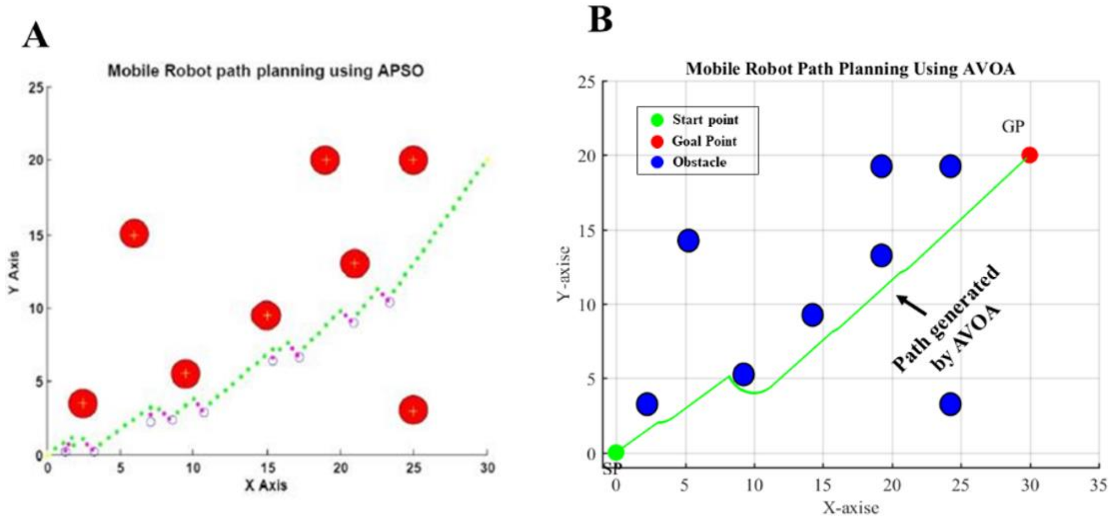

Figure 10. Map (5) simulation graph robot path planning with a starting point at (0,0) and an endpoint at (30,20): (A) APSO [14] Vs AVOA

Table 2. Results of the average Path length covered by the latest literature and AVOA

No. Map	**No. Figure**	**Type of algorithm**	**Average path length**	
			Pixel	**CM**
MAP 1	Figure 6	Hybrid Fuzzy-WDO [21]	2237.31	59.19549375
		AVOA	2214.4432	58.590476333
MAP 2	Figure 7	APSO [14]	1285.56	34.013775
		AVOA	1247.4514	33.005484958
MAP 3	Figure 8	APSO [14]	1784.71	59.19549375
		AVOA	1704.088	45.087328333
MAP 4	Figure 9	APSO [14]	2237.31	59.19549375
		AVOA	2226.30095	33.005484958
MAP5	Figure 10	APSO [14]	470.45	12.447322917
		AVOA	466.4037	12.340264563

As seen in Figure 6 the simulation results in without obstacle environment, and in Figures 7-10 obstacle avoidance in (a narrow escaping environment, local minima environment, trap condition environment, and a maze environment, respectively. Table 2 shows the path length to reach the goal of the proposed algorithm in different maps with the latest literature on APSO, and Hybrid Fuzzy-WDO. From Table 3, it is obvious the enhancement rate in the path length. We can notice that the best result algorithm is based on the enhancement rate in the route length. The best algorithm result was achieved by AVOA.

By using AVOA which is based on navigational strategy mobile robots can easily avoid the two moving obstacles in the dynamic environment, as shown in Figure 11, the unbeknown environment and the moving goal has been tested successfully as shown in Figure 12.

The results that are shown in Figures 11-12 for the dynamic obstacle and the dynamic target the time took were 8.3775,3.6448 with a path length of 30.427,38.3 cm.

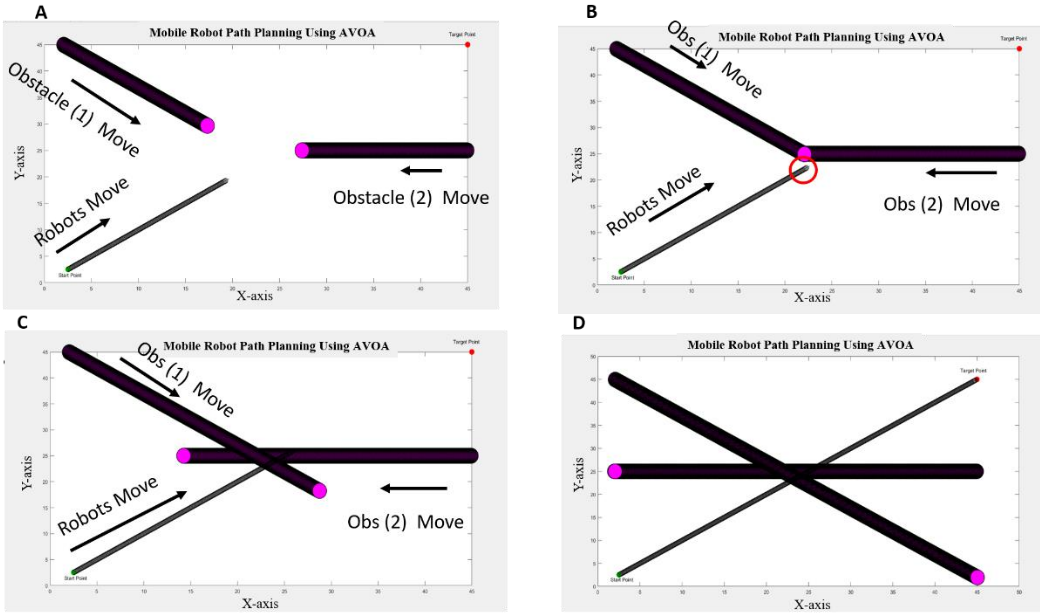

Figure 11. Simulation graph robot path planning with dynamic obstacle unknown environment test of proposed AVOA: A) Start Move (B) Obstacle near to robot (C) Obstacle far to the robot (D) Robots reach to the goal

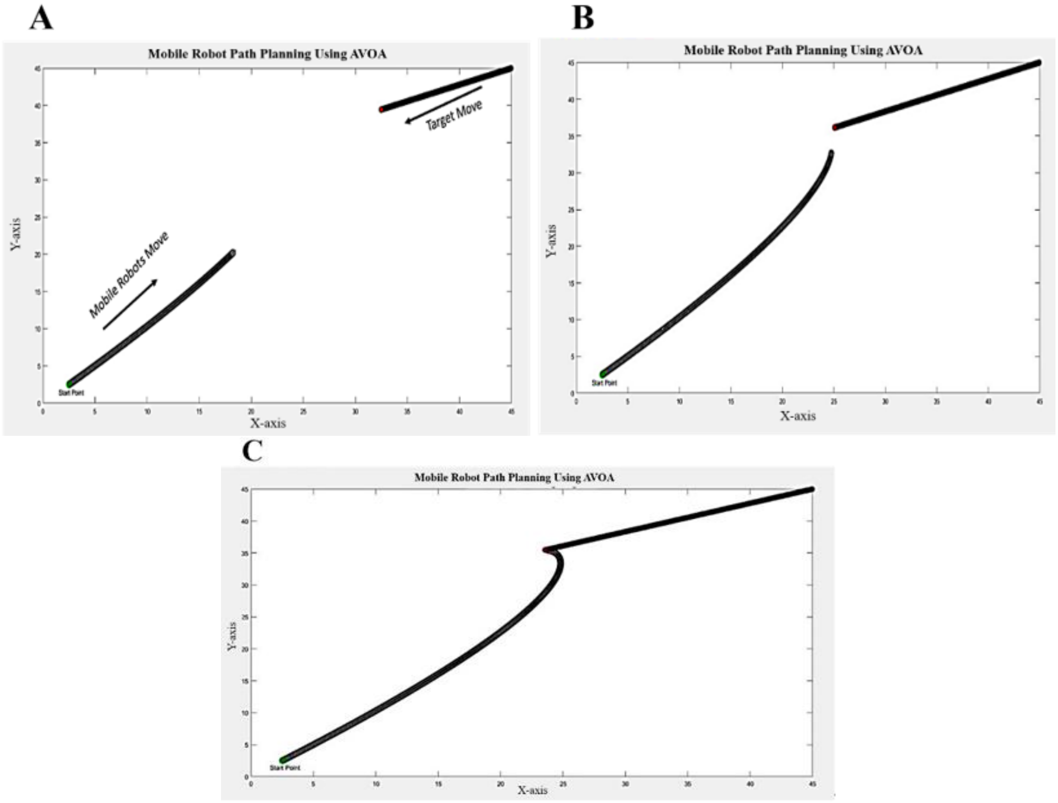

Figure 12. Simulation graph robot path planning with dynamic Goal unknown environment test of proposed AVOA

Table 3. Enhancement rates in the path lengths of the proposed algorithm

Figure Number	Map	Enhancement rate
6	MAP 1	0.364621%
7	MAP2	0.284221%
8	MAP3	0.238941%
9	MAP4	0.294113%
10	MAP 5	1.10651%

6. CONCLUSIONS

The route of efficient navigation has been developed using an online procedure and employing a recently developed approach, which is the AVOA in an unknown environment in the presence of a variety of static and dynamic obstacles. The proposed work is split into two phases: the sensors' reading phase, and the route calculation phase. This algorithm gives the shortest route with a successful obstacle avoidance mechanism for a single mobile robot system within a short time. The proposed approach also exhibits the greatest performance when the environment consists of moving obstacles and goals. The simulational results are compared with other intelligent approaches including the APSO with average enhancement rates in four maps shown in Figures (6-10). Specifically, the enhancement rate obtained by AVOA was 2.21%, and the enhancement rate compared to the Hybrid Fuzzy-WDO for map (1) was 1.02207%. In addition, the environment with a moving target has been successfully tested, in which the robot could successfully follow and reach the moving target. Moreover, the mobile robot has done well in avoiding two moving obstacles in a dynamic environment.

These outcomes can significantly enhance the performance of mobile robots in real-time applications. For our future work, we will extend our method to work experimentally in an unknown maze with dynamic obstacles.

REFERENCES

[1] Wahhab, O.A.R.A., Al-Araji, A.S. (2021). An optimal path planning algorithms for a mobile robot. IRAQI Journal of Computers, Communication, Control & Systems Engineering, 21(2): 44-58. https://doi.org/10.33103/uot.ijccce.21.2.4

[2] Hasan, A.M., Rafaat, S.M. (2020). Optimized formation control of multi-agent system using PSO algorithm. Indones J Electr Eng Comp Sci, 20(3): 1591-1600. https://doi.org/10.11591/ijeecs.v20.i3.pp1591-1600

[3] Raheem, F.A., Hameed, U.I. (2018). Interactive heuristic d* path planning solution based on PSO for two-link robotic arm in dynamic environment. World Journal of Engineering and Technology, 7(1): 80-99. https://doi.org/10.4236/wjet.2019.71005

[4] Abed, M.S., Lutfy, O.F., Al-Doori, Q.F. (2021). A review on path planning algorithms for mobile robots. Engineering and Technology Journal, 39(5): 804-820. https://doi.org/10.30684/etj.v39i5A.1941

[5] Abdulakareem, M.I., Raheem, F.A. (2020). Development of path planning algorithm using probabilistic roadmap based on ant colony optimization. Engineering and Technology Journal, 38(3): 343-351. https://doi.org/10.30684/etj.v38i3A.389.

[6] Raheem, F.A., Raafat, S.M., Mahdi, S.M. (2022). Robot path-planning research applications in static and dynamic environments. In Earth Systems Protection and Sustainability, pp. 291-325. https://doi.org/10.1007/978-3-030-85829-2_12

[7] Kamil, R.T., Mohamed, M.J., Oleiwi, B.K. (2020). Path planning of mobile robot using improved artificial bee colony algorithm. Engineering and Technology Journal, 38(9): 1384-1395. https://doi.org/10.30684/etj.v38i9A.1100

[8] Khaleq Zghair, N.A., Al-Araji, A.S. (2021). A one decade survey of autonomous mobile robot systems. International Journal of Electrical & Computer Engineering, 11(6): 4861-4906. https://doi.org/10.11591/ijece.v11i6.pp4891-4906

[9] Wang, Y., Zhou, H., Wang, Y. (2017). Mobile robot dynamic path planning based on improved genetic algorithm. AIP Conference Proceedings, 1864(1): 020046. https://doi.org/10.1063/1.4992863

[10] Akka, K., Khaber, F. (2018). Mobile robot path planning using an improved ant colony optimization. International Journal of Advanced Robotic Systems, 15(3): 1729881418774673. https://doi.org/10.1177%2F1729881418774673

[11] Rao, D.C., Kabat, M.R., Das, P.K., Jena, P.K. (2019). Hybrid IWD-DE: A novel approach to model cooperative navigation planning for multi-robot in unknown dynamic environment. Journal of Bionic Engineering, 16(2): 235-252. https://doi.org/10.1007/s42235-019-0020-9

[12] Chhillar, A., Choudhary, A. (2020). Mobile robot path planning based upon updated whale optimization algorithm. In 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, pp. 684-691. https://doi.org/10.1109/Confluence47617.2020.9058323

[13] Oleiwi, B.K., Mahfuz, A., Roth, H. (2021). Application of fuzzy logic for collision avoidance of mobile robots in dynamic-indoor environments. In 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), Dhaka, Bangladesh, pp. 131-136. https://doi.org/10.1109/ICREST51555.2021.9331072

[14] Mohanty, P. K., Dewang, H.S. (2021). A smart path planner for wheeled mobile robots using adaptive particle swarm optimization. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 43(2): 743. https://doi.org/10.1007/s40430-021-02827-7

[15] Rahman, I., Vasant, P.M., Singh, B.S.M., Abdullah-Al-Wadud, M. (2016). On the performance of accelerated particle swarm optimization for charging plug-in hybrid electric vehicles. Alexandria Engineering Journal, 55(1): 419-426. https://doi.org/10.1016/j.aej.2015.11.002

[16] Abdollahzadeh, B., Gharehchopogh, F.S., Mirjalili, S. (2021). African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. Computers & Industrial Engineering, 158: 107408. https://doi.org/10.1016/j.cie.2021.107408

[17] Malu, S.K., Majumdar, J. (2014). Kinematics, localization and control of differential drive mobile robot. Global Journal of Research in Engineering, 14(1).

[18] Madhloom, A., Raheem, F.A., Kareem, A.R. (2022). A modified Kalman filter-based mobile robot position measurement using an accelerometer and wheels encoder. Engineering and Technology Journal, 40(1): 267-274. https://doi.org/10.30684/etj.v40i1.2082

[19] Liu, J., Yang, J., Liu, H., Tian, X., Gao, M. (2017). An improved ant colony algorithm for robot path planning. Soft Computing, 21(19): 5829-5839. https://doi.org/10.1007/s00500-016-2161-7

[20] Raheem, F.A., Abdulkareem, M.I. (2020). Development of A* algorithm for robot path planning based on modified probabilistic roadmap and artificial potential field. Journal of Engineering Science and Technology, 15(5): 3034-3054.

[21] Pandey, A., Parhi, D.R. (2017). Optimum path planning of mobile robot in unknown static and dynamic environments using Fuzzy-Wind Driven Optimization algorithm. Defence Technology, 13(1): 47-58. https://doi.org/10.1016/j.dt.2017.01.001