Al-Khwarizmi Engineering Journal, Vol. 18, No. 2, June, (2022)
P. P. 13- 28

Al-Khwarizmi Engineering Journal

Path Planning and Obstacle Avoidance of a Mobile Robot based on GWO Algorithm

Tahseen Fadhil Abbas*   Alaa Hassan Shabeeb**
,Department of Production Engineering and Metallurgy/ University of Technology/ Baghdad/ Iraq*
*Email: 70047@uotechnology.edu.iq
**Email: alaa.h.shabeeb@uotechnology.edu.iq

(Received 16 January 2022; Accepted 27 March 2022)
https://doi.org/10.22153/kej.2022.03.003

Abstract

planning is among the most significant in the field of robotics research. As it is linked to finding a safe and efficient route in a cluttered environment for wheeled mobile robots and is considered a significant prerequisite for any such mobile robot project to be a success. This paper proposes the optimal path planning of the wheeled mobile robot with collision avoidance by using an algorithm called grey wolf optimization (GWO) as a method for finding the shortest and safe. The research goals in this study for identify the best path while taking into account the effect of the number of obstacles and design parameters on performance for the algorithm to find the best path. The simulations are run in the MATLAB environment to test the performance of the proposed algorithm. Simulations results showed that the proposed path planning algorithm effective performance by finding the shortest and free-collision path in different collide environments. Furthermore, the superiority of the proposed algorithm was proved through comparisons with other famous path planning algorithms with different static environments.

Keywords: *Grey Wolf Optimization (GWO), Mobile Robot, Path Planning, Obstacle Avoidance.*

1. Introduction

The fourth industrial revolution and digital innovation have led to the increased interest in the use of robots across industries; thus, robots are being used in various industrial sites, households, military, and medical [1]. In the field of robotics, two important issues are travel preparation (path planning) and collision avoidance, which are studied and discussed by numerous researchers in the past three decades. The primary reason for motion planning is to discover an optimal or almost optimal track from initial to final destination point with the potential for collision avoidance. Therefore, route planning is an important aspect of designing a quick and efficient navigation procedure [2,3]. Path planning can be considered a problem of optimization since it's aimed at finding a path with the shortest distance beneath certain restrictions such as the given environment with motion [4]. Therefore, ensuring a safe route and good path is a difficult challenge for any Wheeled Mobile Robot (WMR). So, attention to the path-planning approach to enable a mobile robot to navigate from the starting point to the final destination with avoiding obstacles is a fundamental need. The major focus of the path planning challenge is on both efficiency and safety. WMR's most fundamental need from an engineering standpoint would be to arrive at its allocated destination safely. To achieve this, any collisions with obstacles must be avoided and prevented [5].

As a result, obstacle avoidance is a prerequisite in the path-planning issue. Secondary criteria might be developed when obstacle avoidance was determined as the primary need. The length of the path should be considered. The goal of the path-planning challenge is for WMR to move from its current location to its destination location with the shortest path. This implies that the method will be more plausible if the path is shorter. Furthermore, the shorter the path, the less time the robot will need to complete the journey [6]. The algorithm's efficiency is also a secondary criterion. In this scenario, the computational cost that the algorithm requires to complete its assigned job is referred to as algorithm efficiency. So, when considering obstacle collision and path length, the algorithm's processing cost must be considered. If an algorithm is computationally costly yet produces a path that isn't considerably better than the competition, it loses its competitive edge. A credible algorithm should balance between the amount of time it takes to run and the quality of the results it generates [7].

In the present work, Grey Wolf Optimization (GWO) algorithm is presented, to solve path planning problems and obstacle avoidance in stationary environments. The goal of this study is to generate an optimal path that would lead to WMR moving in the safe paths, with the quickest time possible.

2. Literature Survey of Previous Research

Recently, researchers and scientists have developed a variety of nature-inspired algorithms. The common objective of these algorithms is to improve the quality of the solutions, stability, and performance of convergence to solve and enhance the path planning of WMR. Ronald Uriol and Antonio Moran. [8] Presented Ant Colony Optimization (ACO) algorithm to solve the path planning challenge for mobile robots, by creating a graphical simulation interface to test and examine the algorithm's performance by looking at parameter settings such as (number of iterations and population) and robot operating environments (number of obstacles). According to the simulation results, a uniform increase in the number of obstacles is primarily overcome by a moderate increase in the ant population, to reduce simulation time and traveled path without collision with obstacles. Harshal S. Dewang et al. [9] (2018) Proposed Adaptive Particle Swarm Optimization (APSO) for getting a solution to the mobile robot path planning problem. The adaptive algorithm was tested by running different simulated experiments in several static environments. To evaluate APSO, a comparison between classical PSO and the proposed algorithm has been carried out based on two parameters: the running time of the algorithm and path length. The simulation results indicated that the APSO was effective in the navigation process, with the shortest path and less computational time compared to classical PSO. Rakaa T. Kamil, et al. [10] Proposed a modified version of the artificial Bee Colony (ABC) algorithm, namely Adaptive Dimension Limit-Artificial Bee Colony (ADL-ABC) algorithm to find the best route with circular-shaped static obstacles in the static environment for a mobile robot. Two tests (or cases) are applied in this work to make a comparison between ABC and ADL-ABC algorithms. The simulation results showed that the ADL-ABC was better than ABC in finding a short path with a fewer number of iterations and less computational time. Fatin H Ajeil et al. [11] have proposed the hybridization of particle swarm optimization (PSO) and modified frequency bat algorithm (MFB) for the planning of the optimal path with an obstacle avoidance strategy. The overall simulation results evaluated indicated the efficacy of the proposed PSO-MFB algorithm in static and dynamic environments. A. Mallikarjuna Raoa et al. [12] proposed adapted Grey Wolf Optimization (GWO) algorithm to robot path planning problems. the robot path planning was implemented in an environment containing three circular shapes. The simulations of the path planning were achieved by finding three points between start, target point. During each iteration, these three points are updated by the GWO algorithm. If the solution point is in the obstacle zone, then the violation is added to the cost function. The results comparison show that the optimal path is found best results for the user test environment. Muna Mohammed Jawad, and Esraa Adnan Hadi [13] have proposed the hybrid (FFCPSO) algorithm to the optimal path for mobile robots by combining the advantages of Chaotic PSO (CPSO), and Firefly (FF) to solve the global path-planning problem for the single and multi-robot environment. The simulation results are carried out in a MATLAB environment. The overall evaluated results indicated that the FFCPSO is a better choice than individuals' algorithms (PSO, FF).

Try researchers and scientists have developed several of the strategies for improved path planning of the mobile robot. The strategy proposed in this research for improved finding the best and safe next position for a mobile robot depended on the GWO algorithm with study the effects of the design parameters on performance for the algorithm to reduce the total time for the path-finding in various environments.

3. Environment Model and Obstacle Extended

The WMR environment in this research work contains many static obstacles, after detection and processing, all obstacles are surrounded and regulated by circle shape to reduce the computation complexity and same time improve the measurement accuracy of the system. In this process, to ensure WMR safety when attempting mobility in the environment, the obstacle size will be increased by adding a safety distance value, as shown in figure 1. After creating a 2D map with a start, destination point, and obstacles, it's used to construct WMR movement. WMR is represented in this map as a point by a collection of cartesian coordinate positions (x,y).

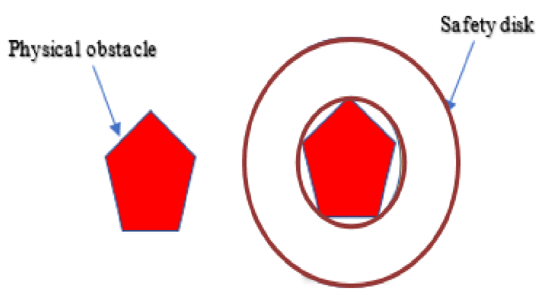


Fig. 1. Increase the Size of The Obstacles.

4. Fitness Function and Selection

The goal of the path planning issue is to discover the best route between a beginning point and a target point. The best road to take may be the one that is the shortest takes the least amount of time and uses the least amount of energy. In most path planning issues, the shortest path is considered the goal function. The x-y coordinates of the mobile robot change once it moves from one spot to the next. In this study, the objective function value for each particle/ agent used is given in Eq. (1)

$$\boldsymbol{f(i)} = \begin{cases} \sum \sqrt{(\boldsymbol{Rx_i} - \boldsymbol{Rx_i} + \mathbf{1})^2 + (\boldsymbol{Ry_i} - \boldsymbol{Ry_i} + \mathbf{1})^2} & \text{(for feasible paths)} \\ \sum \sqrt{(\boldsymbol{Rx_i} - \boldsymbol{Rx_i} + \mathbf{1})^2 + (\boldsymbol{Ry_i} - \boldsymbol{Ry_i} + \mathbf{1})^2} + \textbf{Penalty} & \text{(for infeasible paths)} \end{cases} \quad \ldots(1)$$

Where; f (i) is the fitness function of the summation distance between path points, $\boldsymbol{Rx_i}$ and $\boldsymbol{Ry_i}$ is robot's current position, $\boldsymbol{Rx_i} + \mathbf{1}$ and $\boldsymbol{Ry_i} + \mathbf{1}$ is the robot's next position.

5. Grey Wolf Optimization (GWO)

The GWO method, proposed by Mirjalili et al. [14], is a newcomer in the field of nature-inspired optimization algorithms. Grey wolves' hunting tactics and social order are used to create the Grey Wolf Optimizer (GWO). Grey wolves are divided into four groups: alpha, beta, delta, and omega wolves, according to the hierarchy. The alpha wolf is the group's leader or dominant wolf, and alpha wolves follow the other wolves in the pack. The alpha is the best wolf in terms of managing the group. Beta wolf is the second in the wolf group's social structure. Beta assists the alpha wolf in a variety of duties. The delta wolf must subordinate to the alpha and beta wolves, but the omega wolves are judged by the delta wolf [14]. Grey wolves' collective hunting method is another fascinating social characteristic. The grey wolves use the approach to first detect the location of prey (solution) and then encircle it under the guide of the alpha wolf. In a mathematical model of grey wolf hunting strategy, the alpha, beta, and delta wolves are assumed to have a greater understanding of probable prey locations. As a consequence, the GWO algorithm updates the locations of wolves using the first three best solutions (alpha, beta, and delta), figure.2, showing a flowchart of the GWO algorithm. In the GWO code, there are no omega wolves.

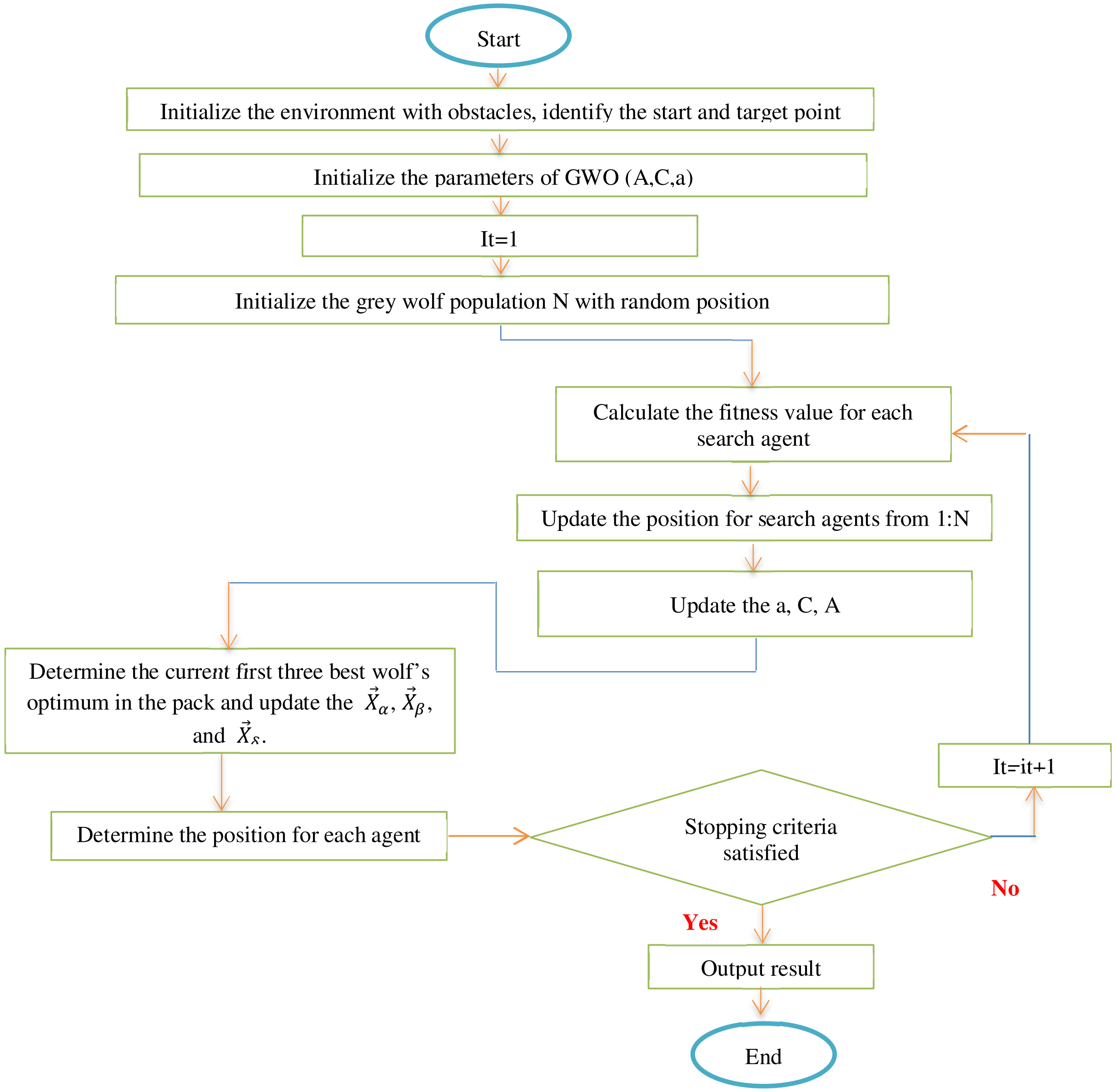


Fig. 2. Flowchart of Path Planning based on GWO Algorithm [15].

The following is a mathematical model of the grey wolf hunting mechanism [15]:

$$\vec{U}_\alpha = \left|\vec{C}_\alpha.\vec{X}_\alpha - \vec{X}_t\right|;\ \vec{U}_\beta = \left|\vec{C}_\beta.\vec{X}_\beta - \vec{X}_t\right|;\ \vec{U}_\delta = \left|\vec{C}_\delta.\vec{X}_\delta - \vec{X}_t\right| \quad \ldots(2)$$

$$\vec{P}_\alpha = \left|\vec{X}_\alpha - \vec{A}_\alpha.\vec{U}_\alpha\right|;\ \vec{P}_\beta = \left|\vec{X}_\beta - \vec{A}_\beta.\vec{U}_\beta\right|;\ \vec{P}_\delta = \left|\vec{X}_\delta - \vec{A}_\delta.\vec{U}_\delta\right| \quad \ldots(3)$$

$$\vec{X}_{(t+1)} = \frac{\vec{P}_\alpha + \vec{P}_\beta + \vec{P}_\delta}{3} \quad \ldots(4)$$

$\vec{U}_\alpha$, $\vec{U}_\beta$, and $\vec{U}_\delta$ are the distance vectors between the prey and the wolf (alpha, beta, delta), $\vec{X}_\alpha$, $\vec{X}_\beta$, and $\vec{X}_\delta$ represent the location vector of the prey For alpha, beta, and delta wolves. $\vec{X}$ represents the grey wolf's location vector at t+1 iteration, the coefficient vectors of alpha, beta, and delta wolves are denoted by $\vec{C}_\alpha$, $\vec{C}_\beta$, $\vec{C}_\delta$, $\vec{A}_\alpha$, $\vec{A}_\beta$ and $\vec{A}_\delta$ respectively. The trial vector for an alpha, beta, and delta wolves is denoted by the letters $\vec{P}_\alpha$, $\vec{P}_\beta$, and $\vec{P}_\delta$. For alpha, beta, and delta wolves, the coefficient vectors are determined as follows:

$$\vec{A}_i = 2\vec{a}.\vec{r}_1 - \vec{a} \quad \ldots(5)$$

$$\vec{C}_i = 2.\vec{r}_2 \quad \ldots(6)$$

Where i is α, β, and δ. a denotes a linear reduction in the vector from 2 to 0 during optimization, the initial random vector in [0,1] is denoted as r. Members of the grey wolf pack change their locations based on alpha, beta, delta, and delta wolves, as well as prey. The grey wolves catch their victim and then attack it to conclude the hunt.

This situation is defined as a decreasing vector in the mathematical model given below:

$$\vec{a} = 2 - \left(\frac{2*iter}{iter_{max}}\right) \quad \ldots(7)$$

6. Simulation Result

6.1. Environment Setup

Three environments are selected to validate the proposed approach. In all cases, the map dimensions are (2400 X 2200) mm, number of handle points is equal to 3 (via points). These environments have static obstacles located at random positions and are unknown in the workspace environment. There is no prior information of the initial WMR location, obstacles, and target location, and assume WMR environment information is defined based on assuming using an external sensor (overhead Camera), to build the environment simulation as input to simulated and path find based on a proposed intelligent algorithm. MATLAB R2019b programming language used to create the simulation code for path planning using tested on Intel(R) core i7, 2.2 GHz CPU, 8.00 GB RAM system. In the mass point simulation, WMR is considered a dimensionless point. Table 1 shows the parameter settings of algorithm parameters that have been used in the simulation.

Table 1,
Parameters used for the GWO algorithm.

parameter	value
wolf sizes (population)	100
No of iterations	10, 20, 40, 80

6.2. Simulation Analysis and Discussion

To determine the effectiveness and efficiency of the proposed algorithm presented in this research work to find the optimal path to WMR navigation, the algorithm was tested in three cases studies (different collide environments). In each environment the optimal path after 100 iterations was selected depending on the safe path as the first step, and the path distance between the start point to the target point, which is determined depending on Eq. 1.

<u>Case 1:</u>

The environment for case study one contains five obstacles as shown in Figure (4). The **start point** of WMR (red circle) and the **target point** (blue circle) are placed at (**165.50, 321.99**) mm, and (**1797.09, 1773.72**) mm respectively as shown in Figure 3.

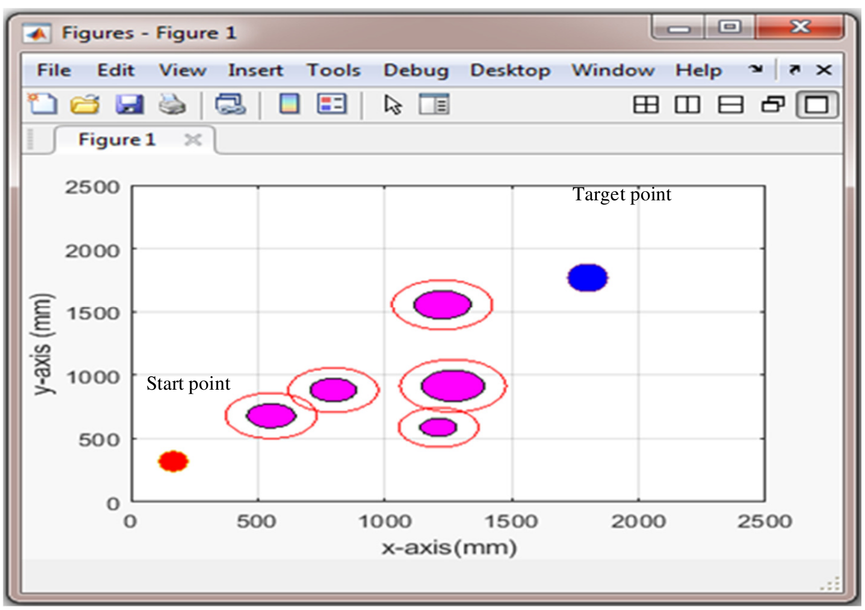


Fig. 3. The Environment of Case 1.

The simulation result of the optimum route for the first case based on the proposed algorithm is shown in figure 4. While figure 5. plotted the relation of the length of the optimal path (better fitness) and the number of iterations for estimation of the optimal value of the objective function for the presented algorithm. In the simulation results of case one, the optimal path of the GWO is equal to **2252.24** mm with a computation time of 9.03 sec. which achieved the optimal solution (**better fitness**).

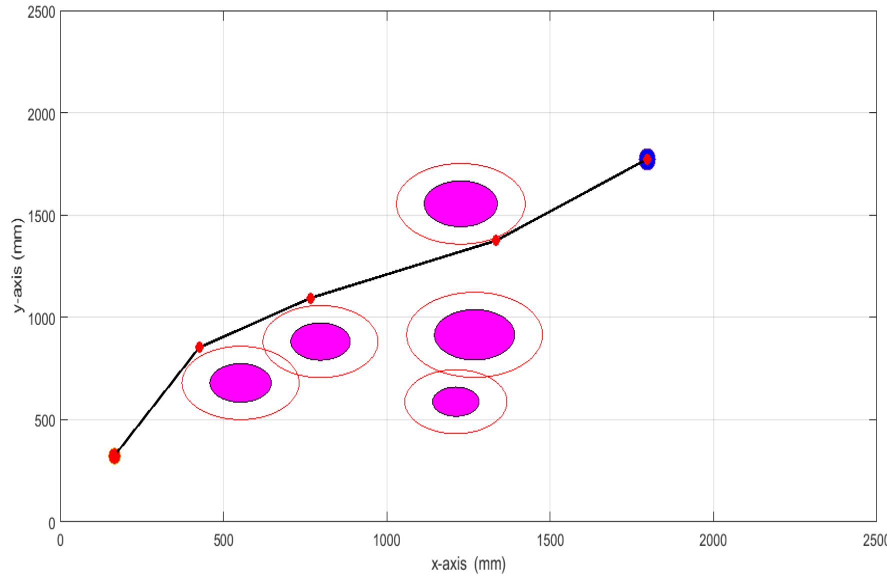


Fig. 4. Path Found by GWO Algorithm for Case 1

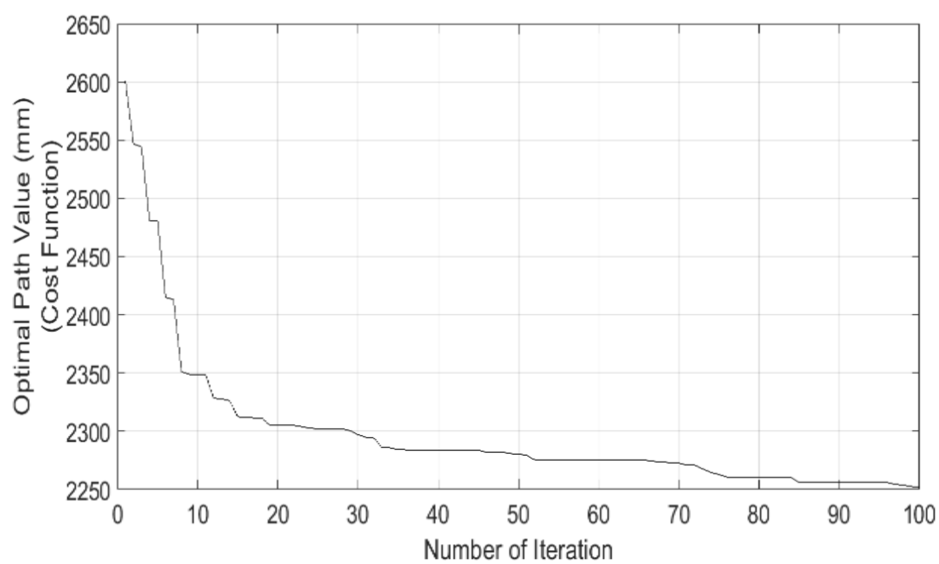


Fig. 5. Best Distance function Founded by GWO for Case 1.

Case 2:

In this case study, the environment contains eight obstacles as shown in figure (4). The **start point** of WMR (red circle) and the **target point** (blue circle) are placed at (**168.46, 322.19**) mm, and (**2039.59, 1766.56**) mm respectively as shown in Figure 6.

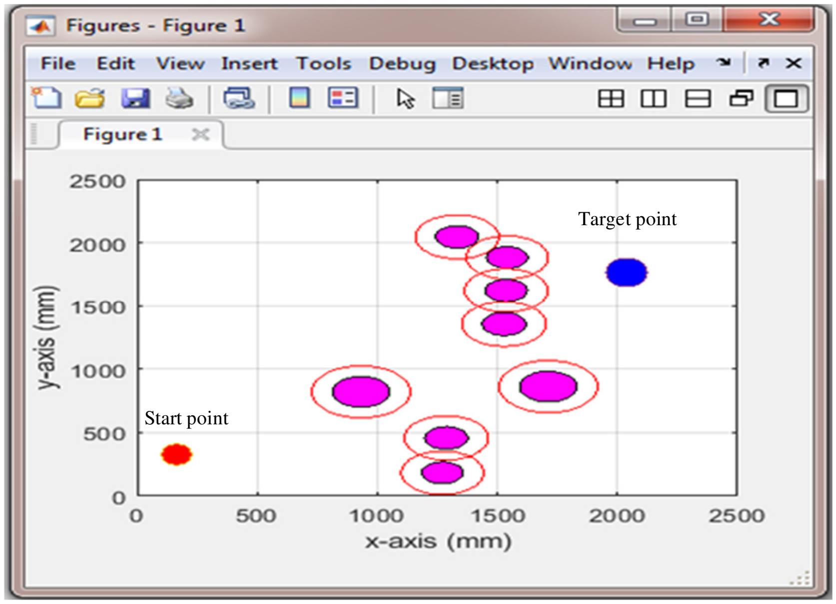


Fig. 6. The Environment of Case 2.

The simulation result of the optimum route for the second case based on the proposed algorithm is shown in Figure 7. While figure 8. plotted the relation of the length of the optimal path (better fitness) and the number of iterations for estimation of the optimal value of the objective function for the presented algorithm. In the simulation results of case two, the optimal path of the GWO is **2493.03** mm with a computation time of 10.59 sec. which achieved the optimal solution (**better fitness**).

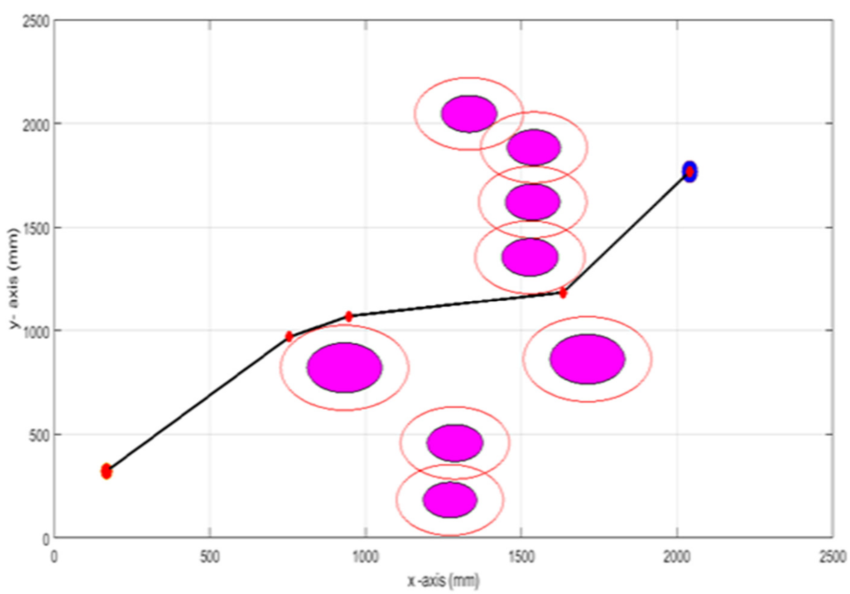


Fig. 7. Path Found by GWO Algorithm for Case 2.

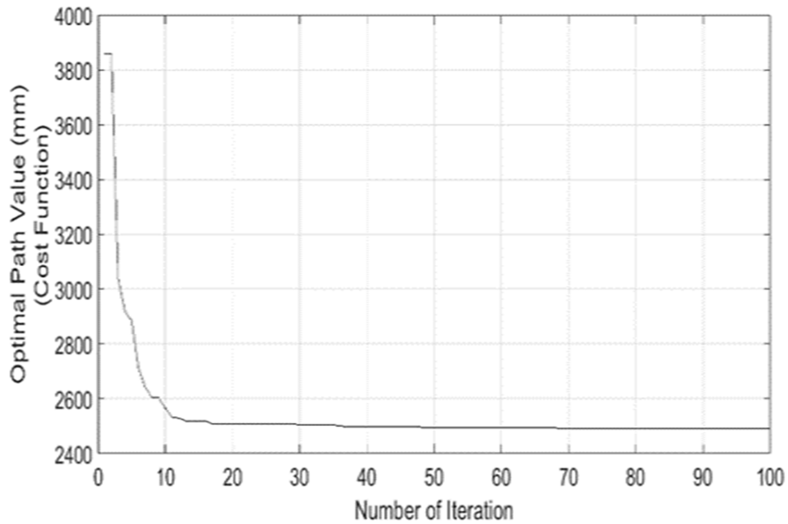


Fig. 8. Best Distance function Founded by GWO Algorithm for Case 2.

Case 3:

In case three the environment contains twelve obstacles as shown in figure (4). The **start point** of WMR (red circle) and the **target point** (blue circle) are placed at (**207.15, 318.23**) mm, and (**2048.32, 914.44**) mm respectively as shown in figure 9.

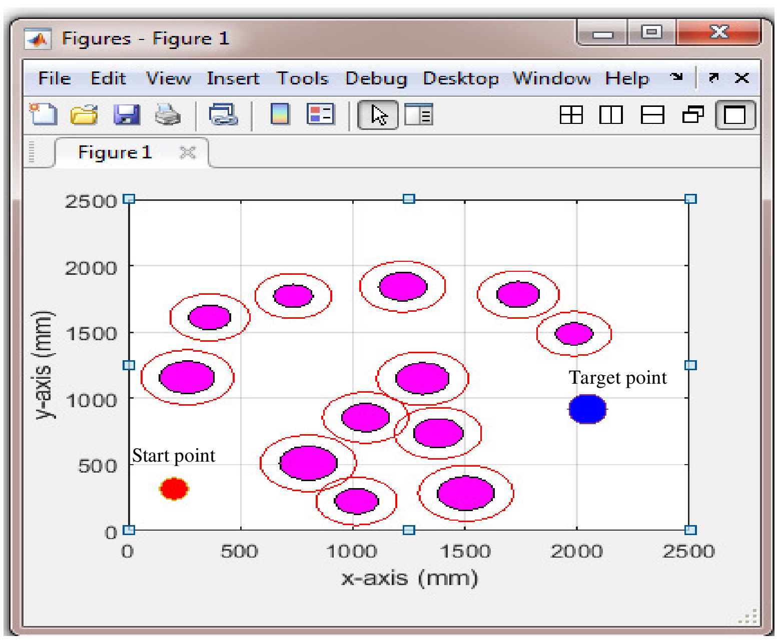


Fig. 9. The Environment of Case 3.

The simulation result of the optimum route for the third case based on the proposed algorithm is shown in figure 10. While figuring 11, plotted the relation of the length of the optimal path (better fitness) and the number of iterations for estimation of the optimal value of the objective function for the presented algorithm. In the simulation results of case three, the optimal path of the GWO is **2493.03** mm with a computation time of 11.77 sec. which achieved the optimal solution (**better fitness**).

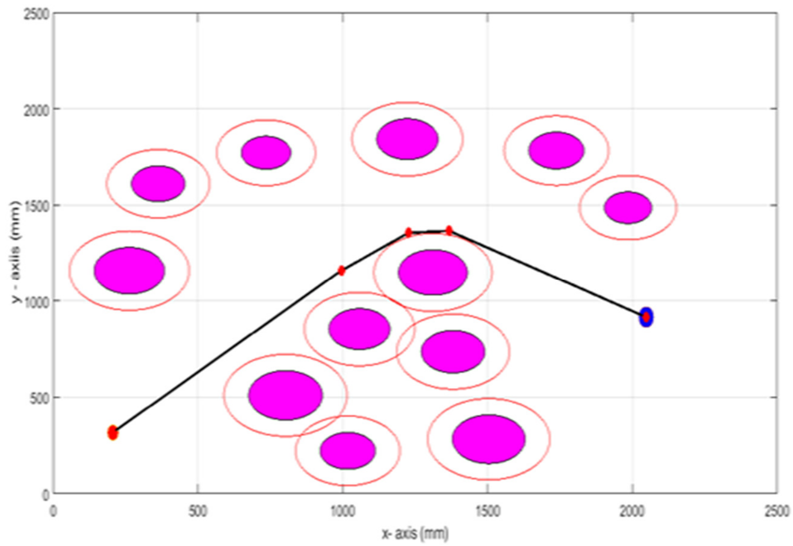


Fig. 10. Path Found by GWO Algorithm for Case 3.

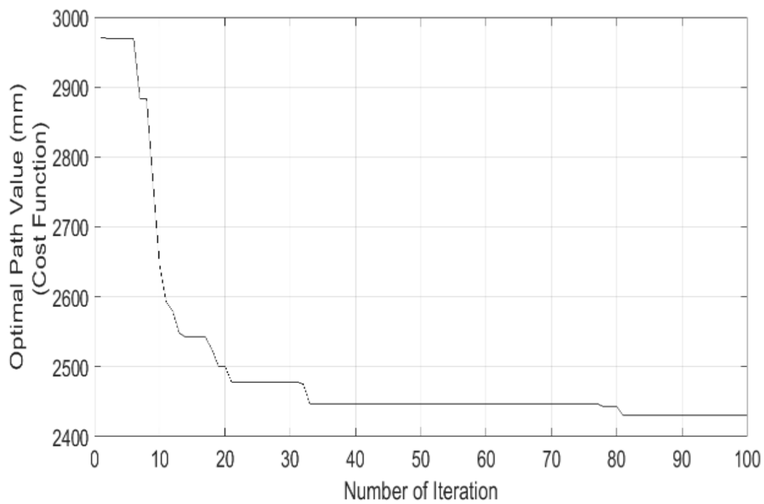


Fig. 11. Best Distance function Founded by GWO Algorithm for Case 3.

6.2.1 Evaluation of Design Parameters Effects on GWO Algorithm Performance

The evaluation is made by summarizing the results for GWO of the three cases environments after executing the algorithm 10 times with four ranges of the number of iterations (10, 20, 40, 80) to evaluate the performance results of the proposed algorithm, on the length of the path and reducing simulation time. Each environment has four paths of safe paths as the first step and distances of these four paths between start point to target point are determined depending on Eq. 1. The best path length is the optimal path between the four best paths achieved by the algorithm in this work. The black path is the best path between the four optimal paths achieved by the algorithm. The best path was selected depending on concerning the safety of the path, path length. The performance comparison of the presented algorithm was measured based on the path length, the simulation time.

Case study one: The simulation results of case one, for the GWO algorithm, with four ranges of the number of iterations as shown in figure 12. In the first path with 10 iterations, the optimal path is equal to 2264.36 mm with a computation time of 8.44 sec. The optimal path of the second path with 20 iterations is equal to 2259.79 mm with a computation time of 8.55 sec. The optimal path of the third path with 40 iterations is equal to 2256.82 mm with a computation time of 8.96 sec. For the fourth path with 80 iterations, the optimal path of the GWO is equal to **2252.24** mm with a computation time of 9.03 sec, which achieved the optimal solution (**better fitness**).

Case study two: In case two as shown in Figure 14, the simulation results of the GWO algorithm with a first range of the number of iterations, the optimal path is equal to 2685.25 mm with a computation time of 8.48 sec. The shortest path of the second range of iterations is equal to 2513.48 mm with a computation time of 9.10 sec. The optimal path of the third of iterations is equal to 2495.37 mm with a computation time of 9.89 sec. For the fourth path with the fourth range of iterations, the best path of the GWO is equal to 2493.03 mm with a computation time of 10.59 sec, which achieved the optimal solution (better fitness).

Case study three: The simulation results of case three as shown in figure 16, for the GWO algorithm, For the first path with 10 iterations, the optimal path is equal to 2452.54 mm with a computation time of 8.44 sec. The optimal path of the second range of iterations is equal to 2438.49 mm with a computation time of 9.12 sec. The optimal path of the third path with 40 iterations is equal to 2433.38 mm with a computation time of 10.57 sec. For the fourth path with the fourth range of iterations, the optimal path of the GWO is equal to 2430.19 mm with a computation time of 11.77 sec, which achieved the optimal solution (better fitness).

The performance results of the proposed GWO algorithm are summarized in Table 2. The performance comparison of the GWO was showed a relationship between the number of obstacles and the number of iterations to find a safe and shortest path with reduce simulation time. Results indicated that a uniform increase of obstacles is mostly solved by a moderate increase in the number of iterations to ensure reduced simulation time without collision with obstacles as shown in Figures 13, 15, and 17.

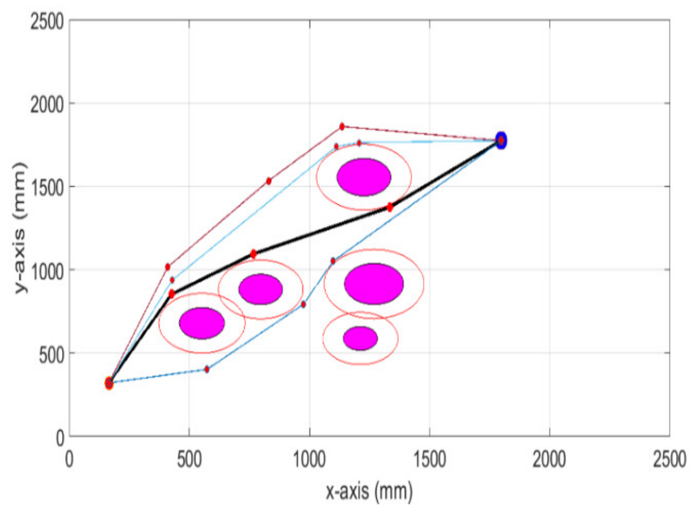


Fig. 12. Path Found by GWO Algorithm for Case 1.

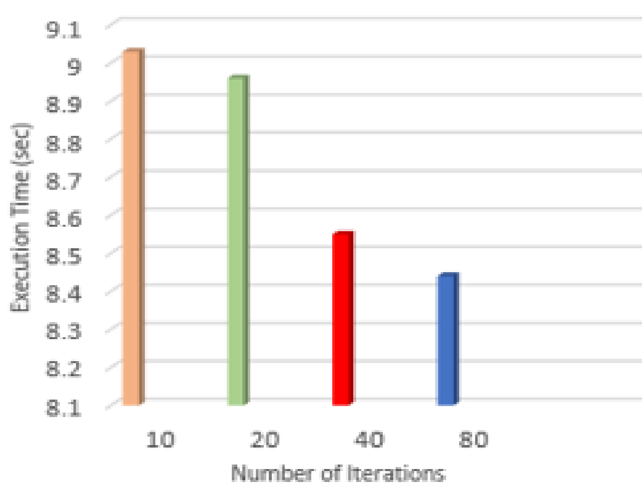


Fig. 13.The execution time of the GWO Algorithm for Case 1.

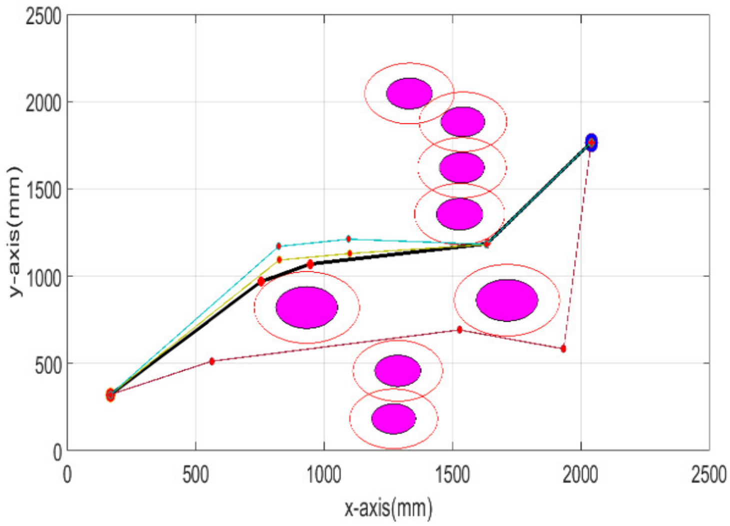


Fig. 14. Path Found by GWO Algorithm for Case 2.

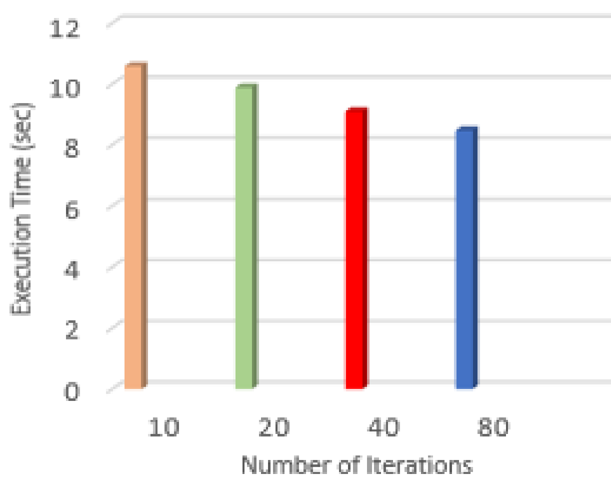


Fig. 15. The execution time of the GWO Algorithm for Case 2.

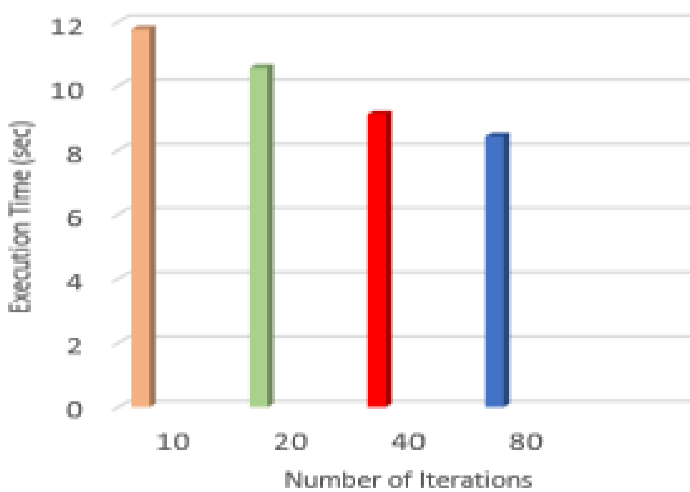


Fig. 17. The execution time of the GWO Algorithm for Case 3

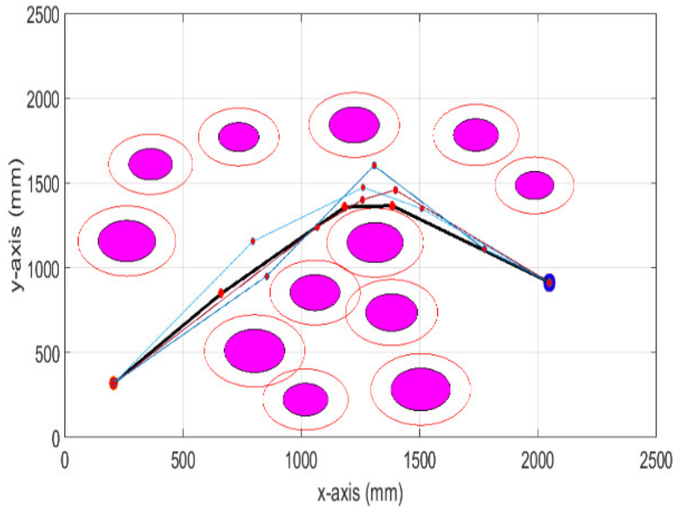


Fig. 16. Path Found by GWO Algorithm for Case 3.

Table 2,
Comparison of the Shortest path length and the time of execution of the GWO algorithm.

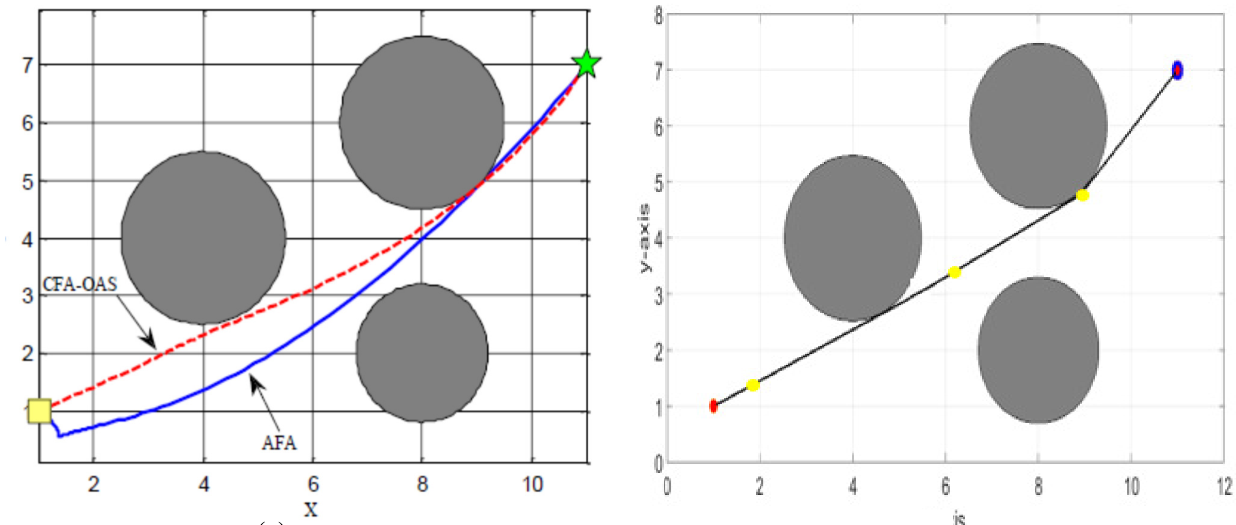
Environment	Number obstacles	Iterations number	Optimal path length (in cm)	GWO execution time (in seconds)
Case 1		10	2264.36	8.44
	5	20	2259.79	8.55
		40	2256. 82	8.96
		80	2252.24	9.03
Case 2		10	2685.26	8.48
	8	20	2513. 48	9.10
		40	2495.37	9.89
		80	2493.03	10.59
Case 3		10	2452.54	8.44
	12	20	2438.49	9.12
		40	2433.38	10.57
		80	2430.19	11.77

6.3 Evaluation of the GWO

Executing the simulation process of any algorithm technique in different environments is not sufficient to assert that it is best. It should provide some proof compared with previously applied strategies to sure that the proposed technique is better.

From this point, the proposed algorithm was compared with other algorithms for the same tested cases to determine the response in the selected environment. The environment has been generated as stated by other authors, and the GWO algorithm has been applied to similar environments, to demonstrate the efficiency of the proposed algorithm.

1- In the first comparative study, which has been carried out by reference [16]. The simulation results were obtained using CFA-OAS and adaptive firefly algorithm (AFA), the navigation system is demonstrated in Figure (18, (a)). While the result obtained using the proposed algorithm is presented in Figure (18, (b)). Table 3. summarizes the best path length, which can be achieved between the proposed GWO algorithm and the algorithms of reference [16].



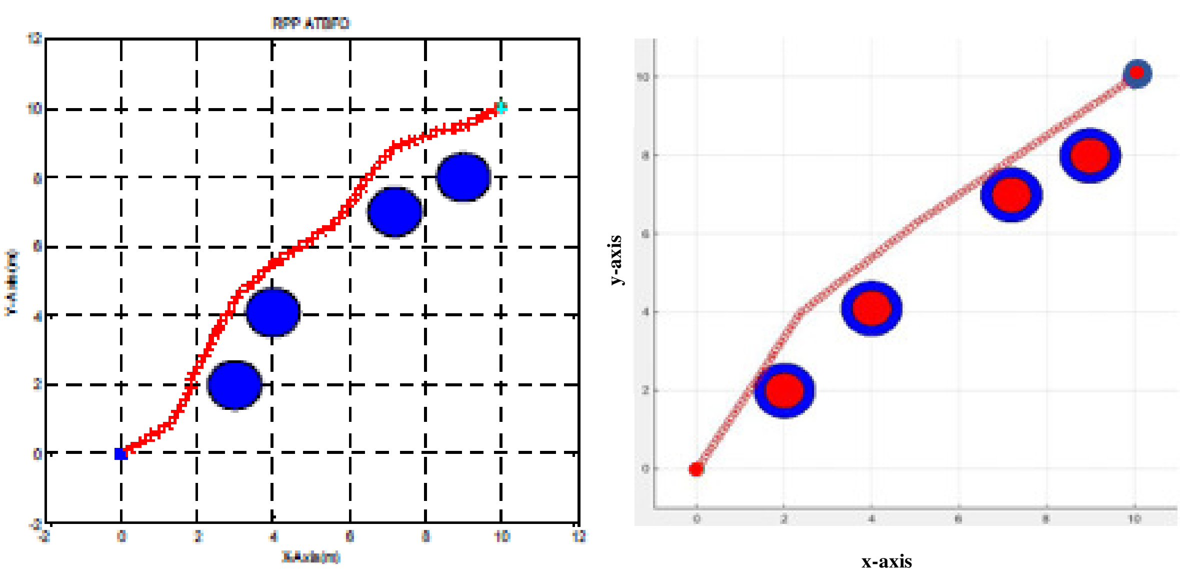
(a) (b)

Fig. (18, (a)-(b)). The Best Results Achieved by (a) Reference [16], and (b) GWO.

Table 3,
Results Comparison with reference [16].

Figure no.	Method	Navigation path length (cm)
Fig (10, (a)).	AFA [48]	12.41
Fig (10, (a)).	CFA-OAS [48]	11.85
Fig (10, (b)).	GWO	11.83

2- The second comparison investigation, based on reference [17], was carried out. They have developed their path planning algorithm based on the approach namely, the Adaptive Tumble Bacterial Foraging Optimization (AT-BFO). Figure (19, (a)) shows the simulation results obtained from the stated AT-BFO algorithm. While, the result obtained from using the proposed algorithm is illustrated in Figure (19, (b)). Table 4. summarizes the best path length, which can be achieved between the proposed GWO algorithm and the algorithms of reference [17].



(a) (b)

Fig. (19, (a)-(b)). The Best Results Achieved by (a) Reference [17], and (b) GWO.

Table 4,
Results Comparison with Reference [17].

Figure no.	Method	Navigation path length (m)
Fig (19, (a)).	AT-BFO [17]	14.5346
Fig (19, (b)).	**GWO**	**14.3574**

From the results obtained, it can be noticed that the proposed algorithm (GWO) provides the shortest path from start to goal position as compared to results obtained by references, [16], and [17].

7. Conclusion

In this paper, we suggested path planning and obstacle avoidance based on the Grey Wolf Optimization path algorithm for WMR in environments with static obstacles. Through analysis of the performance of the algorithm in WMR working environments with studying the effects of the design parameters of the algorithm such as (the number of iterations and number of population) in reducing simulation time and length of the path in different collide environments

The main conclusions of this research can be drawn as follows:

1. The proposed algorithm is proved efficient in finding an optimal path without collision with obstacles.
2. The effect of the values selected of design parameters of the proposed algorithm in reducing simulation time and length of the path without collision with obstacles.
3. The directional relationship between the number of obstacles and the number of iterations. The uniform increase of obstacles is mostly solved by a moderate increase in the number of iterations
4. The proposed algorithm proved the superiority compared with algorithms applied by other researchers, in simulation mode, the results proved that the proposed algorithm is effective in generating the optimal and shortest path, which is positively reflected in the promising applications of this proposed algorithm.

The future work will test the proposed algorithm in real-time WMR navigation.

8. References

[1] T. F. Abaas, and A. H. Shabeeb. "Autonomous Mobile Robot Navigation Based on PSO Algorithm with Inertia Weight Variants for Optimal Path Planning." IOP Conference Series: Materials Science and Engineering. Vol. 928. No. 2. IOP Publishing, 2020.

[2] A. S. Al-Araji, A. K. Ahmed, and M. K. Hamzah. "Development of a Path Planning Algorithms and Controller Design for Mobile Robot." 2018 Third Scientific Conference of Electrical Engineering (SCEE). IEEE, 2018.

[3] A. S. Al-Araji, and N. Q. Yousif. "A Cognitive Hybrid Tuning Control Algorithm Design for Nonlinear Path-Tracking Controller for Wheeled Mobile Robot." Al-Khwarizmi Engineering Journal 13, no. 3 :64-73, 2017.

[4] B. B. Kwame, R. Chellali, A. Y. Appiah, and F. Kyeremeh. "An overview of nature-inspired, conventional, and hybrid methods of autonomous vehicle path planning." Journal of Advanced Transportation, 2018.

[5] I. K. Ibraheem, and F. H. Ajeil. "Path planning of an autonomous mobile robot using swarm based optimization techniques." Al-Khwarizmi Engineering Journal 12, no. 4 (2016): 12-25.

[6] H. Kong, J. Sun, and J. Hu. "Real-time Motion Planning Based on Layered Cost Map for AGV Navigation." In 2020 Chinese Automation Congress (CAC), pp. 7624-7628. IEEE, 2020.

[7] C. B. Lamini, S. and, A. Elbekri. " Genetic algorithm based approach for autonomous mobile robot path planning". Procedia Computer Science, 127, pp.180-184, 2018

[8] 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), pp. 15-21. IEEE, 2017.

[9] H. S. Dewang, P. K. Mohanty, and S. Kundu. "A robust path planning for mobile robot using smart particle swarm optimization." Procedia computer science 133 (2018): 290-297.

[10] R. T. Kamil,, M. J. Mohamed, and Bashra K. Oleiwi. "Path Planning of Mobile Robot Using Improved Artificial Bee Colony Algorithm." Engineering and Technology Journal 38, no. 9A: 1384-1395, 2020.

[11] F. H. Ajeil, I. K. Ibraheem, M. A. Sahib, and A. J. Humaidi. "Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm." Applied Soft Computing Vol 89: 106076 ,2020.

[12] A. M. Rao, K. Ramji, and T. N. Kumar. "Intelligent navigation of mobile robot using grey wolf colony optimization." Materials Today: Proceedings 5, no. 9:19116-19125, 2018.

[13] M. M. AL-Nayar, K. E. Dagher, and E. A. Hadi. "A comparative study for wheeled mobile robot path planning based on modified intelligent algorithms." The Iraqi Journal for Mechanical and Materials Engineering 19, no. 1 : 60-74, 2019.

[14] S. Nadimi, M. H., S.Taghian, and S. Mirjalili. "An improved grey wolf optimizer for solving engineering problems." Expert Systems with Applications Vol 166: 113917, 2021.

[15] T. F. Abaas, and A. H. Shabeeb. "Safe and Optimum Navigation of Wheeled Mobile Robot using Grey Wolf Optimization Algorithm." IOP Conference Series: Materials Science and Engineering. Vol. 928. No. 2. IOP Publishing, 2020.

[16] D. Pang, G. Guan, and J. Li. "Chaotic Firefly Algorithm with the Optimization Adjustment Strategy for Mobile Robot Path Planning." International Journal of Science 4, no. 3 (2017).

[17] N.H. Abbas and Ali, F.M., Path planning of an autonomous mobile robot using enhanced bacterial foraging optimization algorithm. Al-Khwarizmi Engineering Journal, 12(4), pp.26-35, 2016.

تخطيط المسار وتجنب العوائق لروبوت متنقل على أساس خوارزمية GWO

تحسين فاضل عباس*   علاء حسن شبيب**

*،**قسم هندسة الإنتاج والمعادن / الجامعة التكنولوجية / بغداد / العراق

* بريد إلكتروني: 70047@uotechnology.edu.iq

** بريد إلكتروني:alaa.h.shabeeb@uotechnology.edu.iq

الخلاصة

تعد مشكلة تخطيط المسار من بين المشكلات الأكثر أهمية في مجال أبحاث الروبوتات، نظرًا لأنه مرتبط بإيجاد طريق آمن وفعال في بيئة مزدحمة للروبوتات المتنقلة ذات العجلات، ويعتبر شرطًا أساسيًا مهمًا لنجاح أي مشروع روبوت متنقل. تقترح هذه الورقة التخطيط الأسرع والأمثل لمسار الروبوت المتحرك ذي العجلات مع تجنب الاصطدام باستخدام خوارزمية تسمى تحسين الذئب الرمادي (GWO) كطريقة للعثور على أقصر الطرق وأكثرها أمانًا. تهدف أهداف البحث في هذه الدراسة إلى تحديد المسار الأفضل مع مراعاة تأثير عدد المعوقات ومعلمات التصميم على أداء الخوارزمية لإيجاد المسار الأفضل. يتم تنفيذ عمليات المحاكاة في بيئة MATLAB لتقييم كفاءة الخوارزمية المقترحة .أظهرت نتائج عمليات المحاكاة أن خوارزمية تخطيط المسار المقترحة تؤدي أداء فعال من خلال إيجاد أقصر مسار وخالي من الاصطدامات في بيئات تحتوي على عوائق مختلفة. علاوة على ذلك، تم إثبات تفوق الخوارزمية المقترحة من خلال مقارنات مع خوارزميات تخطيط المسار الشهيرة الأخرى مع بيئات ثابتة مختلفة.