TELKOMNIKA Telecommunication, Computing, Electronics and Control
Vol. 19, No. 4, August 2021, pp. 1334~1341
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v19i4.18697

Online 3D path planning for Tri-copter drone using GWO-IBA algorithm

Sadiq Jawad R., Muna H. Saleh
Department of Electrical Engineering, University of Baghdad, Iraq

Article Info

Article history:

Received Nov 18, 2020
Revised Mar 2, 2021
Accepted Mar 11, 2021

Keywords:

GWO
Obstacles avoidance
Online
Path planning
Tri-copter

ABSTRACT

Robots at present are involved in many parts of life, especially mobile robots, which are two parts, ground robots and flying robots, and the best example of a flying robot is the drone. Path planning is a fundamental part of UAVs because the drone follows the path that leads it to goal with obstacle avoidance. Therefore, this paper proposes a hybrid algorithm (grey wolf optimization-intelligent bug algorithm (GWO-IBA)) to determine the best, shortest and without obstacles path. The hybrid algorithm was implemented and tested in the MATLAB program on the Tri-copter model, and it gave different paths in different environments. The paths obtained were characterized by being free of obstacles and the shortest paths available to reach the target.

Corresponding Author:

Sadiq Jawad R.
Department of Electrical Engineering, College of Engineering
University of Baghdad
Baghdad, Karrada, Al-Jadriya, Iraq
Email: eng.sadiq92@gmail.com

1. INTRODUCTION

Recently, mobile robots are starting to take ample space in daily life. They are used in the military, industry, agriculture, exploration, and research, and mobile robots need a pre-tagged target to determine the path to the desired target. So, these robots' path planning process is essential for the robot to reach the target safely and securely [1], [2]. The path planning process for ground robots is done by an algorithm that deals with a two-dimensional environment because the ground robot is moving in two directions (X, Y plane) while flying robots move in a three-dimensional environment [3], [4]. One of the vital mobile robots operating in the 3D environment is the drone [5], where at present, the Drones play an important role in daily life, especially in the military field, in addition to the civilian side of photographing and packages delivered, and in agriculture and discovery field [6], [7]. So, the path planning for the drone trip is crucial to driving the drones to the goal. The choice of the drone route should be fast, safe, and free from obstacles and select the shortest path to reach the desired target [8], [9]. Many researchers have studied methods different to determine the best path for the drones.

Most of the research has dealt with a quadcopter drone model like; (Zhang *et al.* [10] proposal to plan online route UAV using Various differential development algorithm [10]). (Ragi and Chong [11] proposed UAV Path Planning in a Dynamic Environment via Partially Observable Markov Decision Process [11]). (Chen *et al.* [12] proposed UAV layout using the artificial domain update method using optimal control theory [12]). (Primetista *et al.* [13] suggested risk and aware of the planning path of the UAVs in urban environments [13]). (Nguyen *et al.* [14] proposed online track chart joint detection, tracking multiple objects with wireless markers [14]).

In this paper, a Tri-copter model was used to test a suggested algorithm that determined the best path for a trip in addition to avoiding obstacles in a dynamic 3D environment described in section two. In sections three and four, a hybrid algorithm grey wolf optimization–intelligent bug algorithm (GWO-IBA) was proposed to determine the best and shortest path with obstacles avoidance. The algorithm was applied to a Tri-copter model using the MATLAB program to simulate the model and the algorithm in section five.

2. PROBLEM STATEMENT

In the beginning, assume that the Tri-copter plane has a radius (r) and is located at the starting point with coordinates (xs, ys, zs). What is required is that the Tri-copter moves from the starting position to the desired target with coordinates (xg, yg, zg) in a 3D environment that has dynamic and static obstacles between the starting point and the target. So, the required is to draw a flight path in a 3D environment from the beginning to the endpoint. This path should be barrier-free (avoid obstacles), be the shortest available paths, and it should choose and calculate the path in real-time and speed during the flight process. The cost function for the algorithm that maps this path is:

$$costfunction = \sqrt{(x_g - x_s)^2 + (y_g - y_s)^2 + (z_g - z_s)^2} \quad (1)$$

The cost function is a distance between the current point and target, which minimizes the distance between the starting point and the ending point. Any algorithm that works to reduce the distance between two points will meet the local minimum point's problem, as the algorithm cannot exit from this point as shown in Figure 1, therefore need to propose an algorithm that deals with this problem.

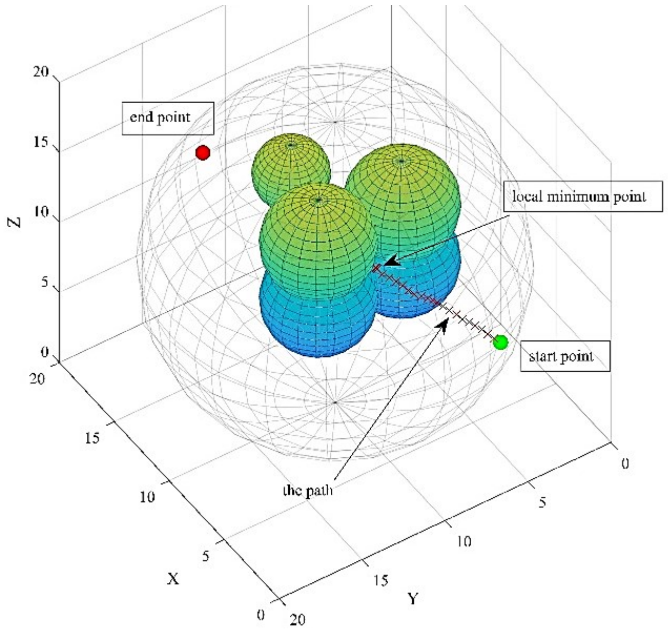


Figure 1. The local minimum point in path planning

3. OPTIMIZATION TECHNIQUE GWO-IBA

In general, the wolves have a robust capacity to catch the victim. Wolves live in social life and, any group of wolves has a rigid social hierarchy, as shown in Figure 2 [15]. The wolves' group leadership hierarchy can be divided into four kinds of wolfs: omega, delta, beta, and alpha, where alpha is the best individual. The alpha wolf is the leader wolf in the group, and all other wolves should follow it. The second-best individual is beta that can be three wolfs. The beta is vassal wolves that assist the alpha in making a decision. The delta is the third-best individual in the group; they are vassal also. Delta wolves depend on the beta and alpha, but they command on the omega, and the rest of the wolf group count as omega. In this optimization method (GWO), the optimal solutions are leading by alpha, beta, and delta[16], [17]. They lead residue wolves (omega) to the optimal point in the searching area, as shown in Figure 2.

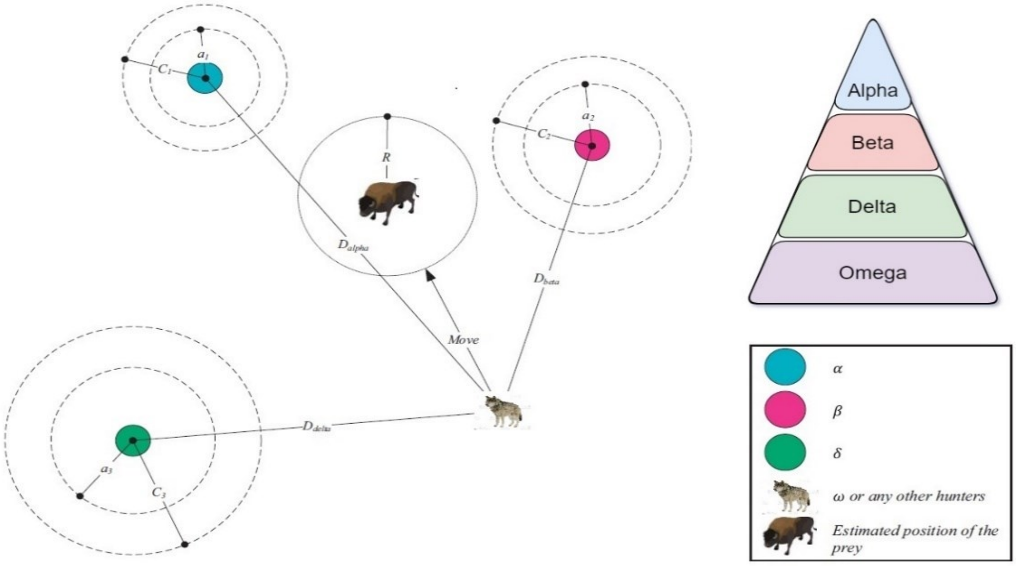


Figure 2. The GWO method and social hierarchy [18]

In the optimization process, as shown in (2) and (3) used to update the locations of wolves [19], [20].

$$\vec{D} = \vec{C} \times \overrightarrow{X_p}(n) - \vec{X}(n) \tag{2}$$

$$\vec{X}(n + 1) = \overrightarrow{X_p}(n) - \vec{A} \times \vec{D} \tag{3}$$

where (n) is an iteration, $\vec{C}\ and\ \vec{A}$ are vectors can be calculated by use as shown in (4) and (5), $\overrightarrow{X_p}(n)$ is a vector refer to the prey position, $\vec{X}(n)$ is the wolf position vector.

$$\vec{C} = 2.r_1 \tag{4}$$

$$\vec{A} = 2.a.r_2 - a \tag{5}$$

$$a = 2(1 - \frac{n}{N\ max}) \tag{6}$$

where (N max) is a maximum iteration of the searching, r_1 and r_2 are a random vector [0, 1]. In mathematical calculations of how to catch grey wolves, it was assumed that alpha, beta, and delta knew the possible location of prey. Therefore, the three best solutions are saved, and the other wolf group is obligated to update their positions according to the best research process, as shown in the following as shown in (7)-(13) [21].

$$\overrightarrow{D_\alpha} = \overrightarrow{C_1} \times \overrightarrow{X_\alpha}(n) - \vec{X}(n) \tag{7}$$

$$\overrightarrow{D_\beta} = \overrightarrow{C_2} \times \overrightarrow{X_\beta}(n) - \vec{X}(n) \tag{8}$$

$$\overrightarrow{D_\delta} = \overrightarrow{C_3} \times \overrightarrow{X_\delta}(n) - \vec{X}(n) \tag{9}$$

$$X_1 = X_\alpha - A_1 \times D_\alpha \tag{10}$$

$$X_2 = X_\beta - A_2 \times D_\beta \tag{11}$$

$$X_3 = X_\delta - A_3 \times D_\delta \tag{12}$$

$$X(n + 1) = \frac{(X1 + X2 + X3)}{3} \tag{13}$$

The GWO algorithm was chosen to determine the shortest optimal path, with some modifications to the algorithm to deal with the 3D environment. Its ability to avoid obstacles in addition to mixing it with the Intelligent Bug Algorithm (IBA) [22] in case the algorithm reaches the local minimum value of the distance. Where the Bug Algorithm is an algorithm used to avoid obstacles as the robot, after detecting the obstacle, moves Along the boundaries of the obstacle until it reaches the point where the obstacle was detected (completes a cycle around the obstacle) and then calculates the best point to move through it to the target. Then the algorithm was developed into Bug-2 where it relied on the calculation of slope as shown in (14) where the leave point it when the slope equal to the original slope. Some research has been done to make the algorithm smarter, but it still works in a two-dimensional environment [23], [24]. Where (x_1, y_1) is the start point and (x_2, y_2) is endpoint.

$$slope = tan^{-1}\frac{x_1 - x_2}{y_1 - y_2} \quad (14)$$

4. PROPOSED METHOD

The wolf moves in three directions x, y, z, as shown in Figure 3 in a three-dimensional environment. The algorithm is modified to choose the shortest path for prey and avoid obstacles in the way. However, the algorithm stops and cannot return if it reaches a local minimum point because all the solutions proposed by the algorithm will move it away from the target. It means all new proposed points will have a greater distance than the distance of the local (existing) point to the destination. So, it was combined with the Bug algorithm to make it more intelligent. The built-in modified algorithm (hybrid algorithm) can avoid obstructions and exit from the local minimum points of distance from the target.

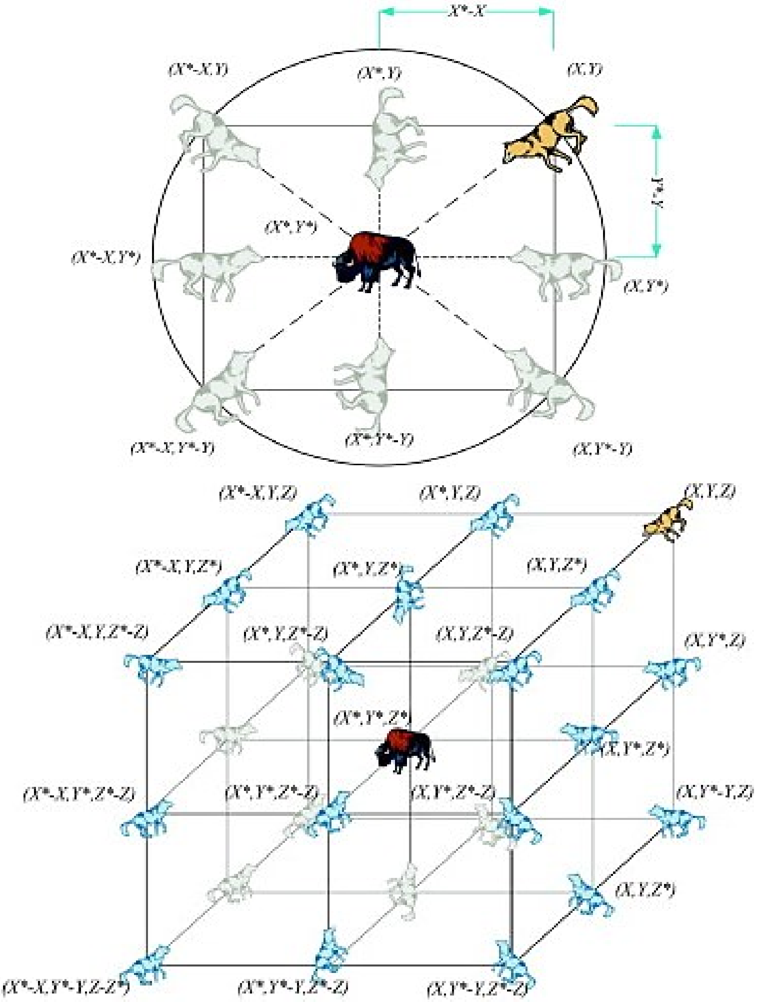


Figure 3. Movement of a wolf in GWO [25]

In other words, the grey wolf algorithm selects the shortest obstacle-free path by finding new points for the transport of the Tri-Copter. Suppose this algorithm fails to find points closer to the target and is free of obstacles. In that case, the other algorithm works by drawing a path along with the barrier until it reaches the opposite side of the obstacle where the wolf algorithm can operate. Moreover, the flowchart is illustrated in Figure 4 how the algorithm works and defines it for the shortest path free of obstacles.

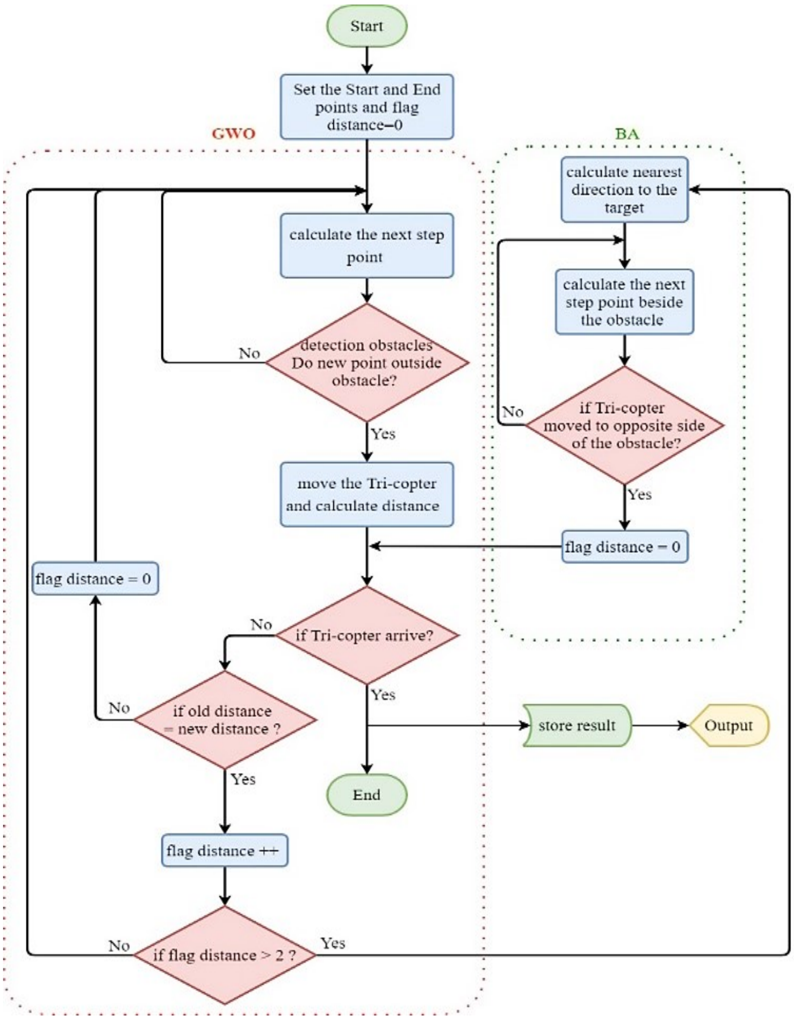


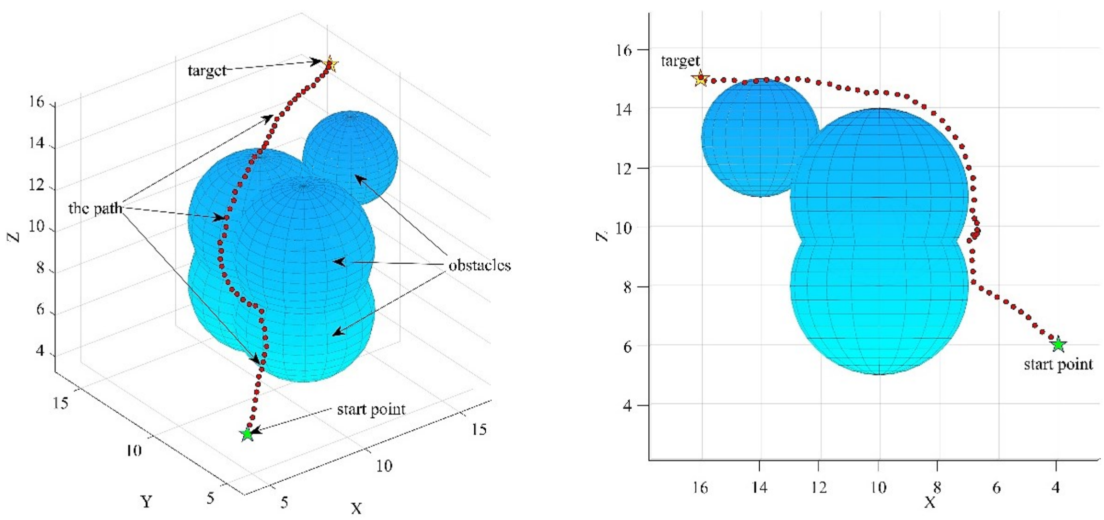
Figure 4. Flowchart for the hybrid GWO-IBA method

5. SIMULATION RESULTS

The Tri-copter movement was simulated in the MATLAB 2020a program. A three-dimensional environment was painted with different obstacles. The starting point and endpoint specified, and the hybrid algorithm tested according to the parameters list used to run the algorithm are listed in Table 1. The results are shown in Figures 5-8. In Figures 7 and 8, the small red points (red path) refer to the path generated by the GWO algorithm, and the small blue points (blue path) refer to a path generated by the IBA.

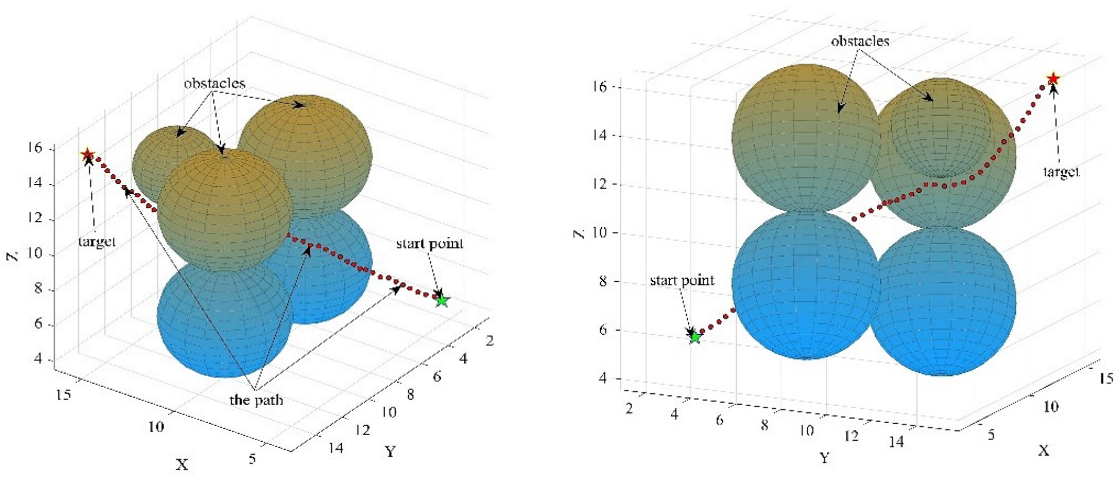
Table 1. List of the constant that uses in simulation

Parameters	Means	Values	Units
Iteration	Max. number of iterations	10	unitless
W	Number of wolves	30	unitless
D	Dimension	3	unitless
R	The radius of the Tri-copter	20	cm
s	Max. step of the Tri-copter	60	cm



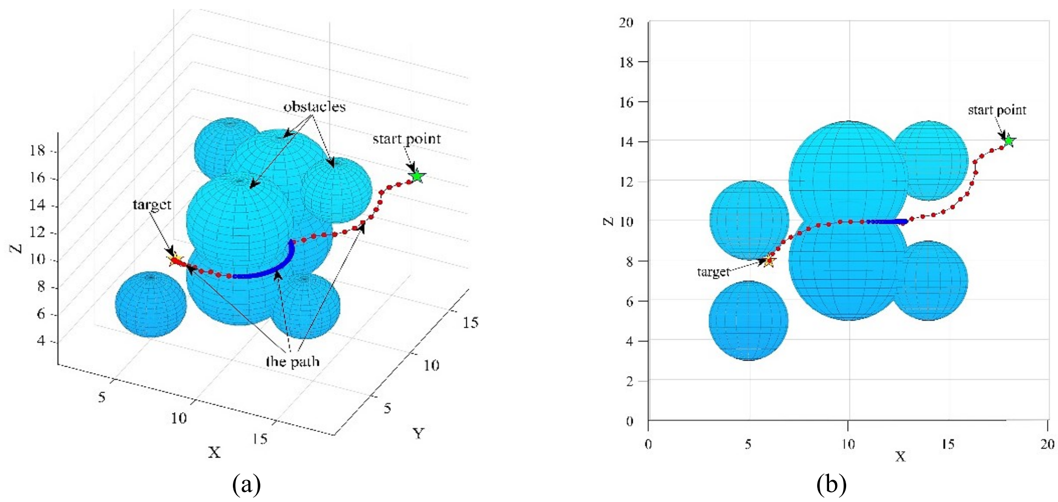
(a) (b)

Figure 5. The path planning results without a local minimum point: (a) 3D view, (b) side view



(a) (b)

Figure 6. The path planning results with multi obstacles: (a) 3D view, (b) opposite side view



(a) (b)

Figure 7. The path planning results with a local minimum point: (a) 3D view, (b) side view

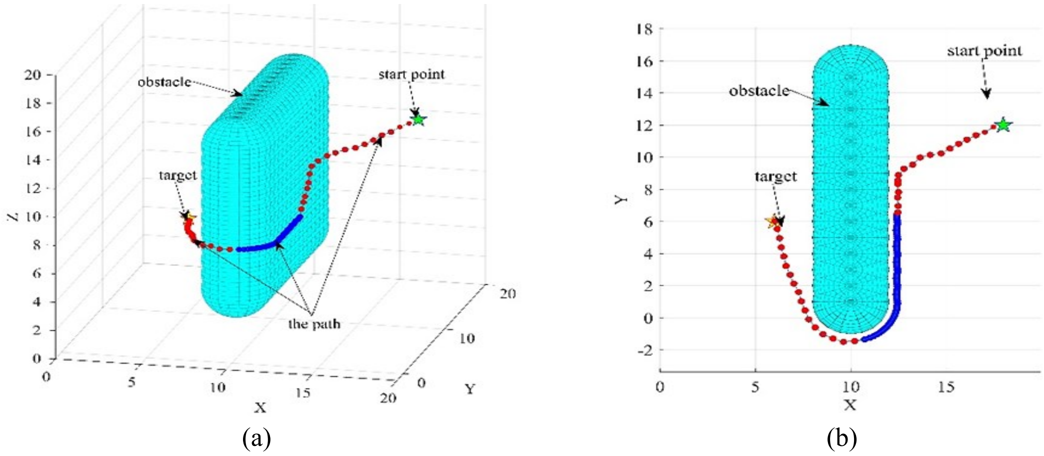


Figure 8. The path planning results for enormous obstacles: (a) 3D, (b) top view

6. DISCUSSION

The previous results showed that the hybrid algorithm was able to solve the problem of the local minimum point that it faced in the GWO or any traditional algorithm that deals with immediate path planning. Thus, its superiority over the traditional algorithms by choosing the shortest path and avoid obstacles. Moreover, the proposed algorithm has achieved suitable results in drawing the path without problems, as shown in Figures 5 and 8, adding to the algorithm has the ability to dealing with various obstacles in the three-dimensional environment.

7. CONCLUSION

In this paper, a hybrid algorithm (GWO-IBA) was proposed where this algorithm based on grey wolf behaviour and Bug's behaviour. The proposed algorithm draws a path for the movement of the plane in a three-dimensional environment. This path is the shortest paths available at present (optimal local path) and is free from obstacles. Moreover, choosing the path was done with a minimum number of search iterations that mean a fast process. The results shown above can observe the (GWO-IBA) algorithm's performance to lead the drone from the starting point to the endpoint (the desired target) without collisions or stopping points. Furthermore, the drone size was taken into account to map a path that avoids collisions with obstacles.

REFERENCES

[1] B. K. Patle, Ganesh Babu, A. Pandey, D. R. K. Parhi, and A. Jagadeesh, "A review: On path planning strategies for navigation of mobile robot," *Def. Technol.*, vol. 15, no. 4, pp. 582-606, Aug. 2019, doi: 10.1016/j.dt.2019.04.011.

[2] J. H. Zhang, Y. Zhang, and Y. Zhaou, "Path planning of mobile robot based on hybrid multi-objective bare bones particle swarm optimization with differential evolution," *IEEE Access*, vol. 6, pp. 44542-44555, Aug. 2018, doi: 10.1109/ACCESS.2018.2864188.

[3] H. Cheon and B. K. Kim, "Online bidirectional trajectory planning for mobile robots in state-time space," *IEEE Trans. Ind. Electron.*, vol. 66, no. 6, pp. 4555-4565, Jun 2019, doi: 10.1109/TIE.2018.2866039.

[4] S. Hosseininejad and C. Dadkhah, "Mobile robot path planning in dynamic environment based on cuckoo optimization algorithm," *Int. J. Adv. Robot. Syst.*, vol. 16, no. 2, pp. 1-13, Apr. 2019, doi: 10.1177/1729881419839575.

[5] Iswanto, O. Wahyunggoro, and A. I. Cahyadi. "Quadrotor path planning based on modified fuzzy cell decomposition algorithm," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 14, no. 2, pp. 655-664, Jun. 2016, doi: 10.12928/telkomnika.v14i2.2989.

[6] N. M. A. Ahmed and M. H. Saleh, "Tri-copter Drone Modeling with PID Control Tuned by PSO Algorithm," *Int. J. Comput. Appl.*, vol. 181, no. 25, pp. 46-52, Nov. 2018, doi: 10.5120/ijca2018918060.

[7] C. Yu and Z. Wang, "UAV path planning using GSO-DE algorithm," *IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON*, Oct. 2013, doi: 10.1109/TENCON.2013.6718927.

[8] R. K. Dewangan, A. Shukla, and W. Godfrey, "Three dimensional path planning using Grey wolf optimizer for UAVs," *Appl. Intell.*, vol. 49, no. 6, pp. 2201-2217, Jun. 2019, doi: 10.1007/s10489-018-1384-y.

[9] A. K. Abdul Hassan and D. J. Fadhil, "Mobile Robot Path Planning Method Using Firefly Algorithm for 3D Sphere Dynamic & Partially Known Environment," *J. Univ. Babylon Pure Appl. Sci.*, vol. 26, no. 7, pp. 309-320, Jul. 2018, doi: 10.29196/jubpas.v26i7.1506.

[10] X. Zhang, J. Chen, Bim Xin, and H. Fang, "Online path planning for UAV using an improved differential evolution algorithm," *IFAC Proc. Vol.*, vol. 44, no. 1, pp. 6349-6354, Jan. 2011, doi: 10.3182/20110828-6-IT-1002.01807.

[11] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 4, pp. 2397-2412, Oct. 2013, doi: 10.1109/TAES.2013.6621824.
[12] Y. B. Chen, G. Chen Luo, J. Qiao, Y. Song Mei, "UAV path planning using artificial potential field method updated by optimal control theory," *Int. J. Syst. Sci.*, vol. 47, no. 6, pp. 1407-1420, Dec. 2016, doi: 10.1080/00207721.2014.929191.
[13] S. Primatesta, G. Guglieri, and Alessandro Rizzo, "A Risk-Aware Path Planning Strategy for UAVs in Urban Environments," *J. Intell. Robot. Syst. Theory Appl.*, vol. 95, no. 2, pp. 629-643, Aug. 2019, doi: 10.1007/s10846-018-0924-3.
[14] H. Van Nguyen, H. Rezatofighi, Ba-Ngu Vo, and D. Ranasinghe, "Online UAV Path Planning for Joint Detection and Tracking of Multiple Radio-Tagged Objects," *IEEE Trans. Signal Process.*, vol. 67, no. 20, pp. 5365-5379, Sep. 2019, doi: 10.1109/TSP.2019.2939076.
[15] N. Jain, G. Parmar, R. Gupta, and I. Khanam, "Performance evaluation of GWO/PID approach in control of ball hoop system with different objective functions and perturbation," *Cogent Eng.*, vol. 5, no. 1, pp. 1-18, Apr. 2018, doi: 10.1080/23311916.2018.1465328.
[16] O. Zebua, I. Made Ginarsa, and I. Made Ari Nrartha, "GWO-based estimation of input-output parameters of thermal power plants," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 18, no. 4, pp. 2235-2244, Aug. 2020, doi: 10.12928/telkomnika.v18i4.12957.
[17] M. A. Şen and M. Kalyoncu, "Optimal Tuning of PID Controller Using Grey Wolf Optimizer Algorithm for Quadruped Robot," *Balk. J. Electr. Comput. Eng.*, vol. 6, no. 1, pp. 29-35, Feb. 2018.
[18] A. A. Ewees and M. A. Elaziz, "Improved Adaptive Neuro-Fuzzy Inference System Using Gray Wolf Optimization: A Case Study in Predicting Biochar Yield," *J. Intell. Syst.*, vol. 29, no. 1, pp. 924-940, Sep. 2018, doi: 10.1515/jisys-2017-0641.
[19] A. Ahmed, Rajeev Gupta, and G. Parmar, "GWO/PID Approach for Optimal Control of DC Motor," *2018 5th Int. Conf. Signal Process. Integr. Networks, SPIN,* Feb. 2018, pp. 181–186, doi: 10.1109/SPIN.2018.8474105.
[20] S. S. Sabry and N. M. Kaittan, "Grey wolf optimizer based fuzzy-PI active queue management design for network congestion avoidance," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 1, pp. 199-208, Apr. 2020, doi: 10.11591/ijeecs.v18.i1.pp199-208.
[21] S. Mirjalili, "Nature-inspired Algorithms," *Stud. Comput. Intell.*, vol. 811, pp. 91-110, 2020.
[22] Y. Ichikawa and N. Ozaki, "Autonomous Mobile Robot," *Journal of Robotic Systems*, vol. 2, no. 1. pp. 135-144, 1985.
[23] N. Sharma, S. Thukral, Sandip Aine, P. B. Sujit, "A virtual bug planning technique for 2D robot path planning," *2018 Annual American Control Conference (ACC)*, Jun. 2018, doi: 10.23919/ACC.2018.8430678.
[24] M. Zohaib, M. Pasha, N. Javaid, and J. Iqbal, "IBA: Intelligent bug algorithm-a novel strategy to navigate mobile robots autonomously," *Commun. Comput. Inf. Sci.*, vol. 414, pp. 291-299, Sep. 2013, doi: 10.1007/978-3-319-10987-9_27.
[25] P. Niu, S. Niu, Nan Liu, and L. Chang, "The defect of the Grey Wolf optimization algorithm and its verification method," *Knowledge-Based Syst.*, vol. 171, pp. 37-43, May 2019, doi: 10.1016/j.knosys.2019.01.018.

BIOGRAPHIES OF AUTHORS

Sadiq Jawad R. graduated from the University of Baghdad / College of engineering/department of electrical engineering in 2014 and has several department activities related to programming and designing smart traffic lights, control of mobile vehicles, and smart homes and IoT. In 2014 the author started working at the Buzurgan oil field in Maissan southern Iraq, and cooperated with (Cnooc) China Company. He maintains and installs control panels of power stations, and his new field specialises in control and monitoring drones to detect faults in pipes and power overhead lines.

Muna H. Saleh, Ph.D. (Computer & Information Technology Engineering) Associate Professor/ Electrical Department/ Engineering Collage/ University of Baghdad/ Iraq. E-mail: dr.muna.h@coeng.uobaghdad.edu.iq B. Sc in Control and System Engineering/University of Technology/ Date of graduation is 1989. M.Sc in Control and Computer Engineering/University of Technology/ Date of graduation is 1996. Ph. D in Computer and Technology Information Engineering/ University of Technology/Date of graduation is 2006.