Fırat Üniversitesi Deneysel ve Hesaplamalı Mühendislik Dergisi

İnsansız Araç Navigasyonunun Optimize Edilmesi: Verimli Rota Planlaması için Hibrit PSO-GWO Algoritması

Gökhan ALTUN[1], İlhan AYDIN[2*]

[1]Tatvan Meslek Yüksekokulu, Bitlis Eren Üniversitesi, Bitlis, Türkiye.
[2]Bilgisayar Mühendisliği, Mühendislik Fakültesi, Fırat Üniversite, Elazığ, Türkiye.
[1]altungokhann@gmail.com, [2]iaydin@firat.edu.tr

Geliş Tarihi: 14.06.2024
Kabul Tarihi: 16.09.2024

Düzeltme Tarihi: 10.09.2024

doi: https://doi.org/10.62520/fujece.1501508
Araştırma Makalesi

Alıntı: G. Altun ve İ. Aydın, "İnsansız araç navigasyonunun optimize edilmesi: verimli rota planlaması için hibrit pso-gwo algoritması", Fırat Üni. Deny. ve Hes. Müh. Derg., vol. 4, no 1, pp. 100-114, Şubat 2025.

Öz

Bu çalışma, insansız araçların kullanımında önemli bir yere sahip olan otonom sistemler için rota planlama problemini ele almayı amaçlamaktadır. Belirtilen problemin çözümünde kullanılacak olan meta-sezgisel algoritma yaklaşımlarının performansını artırmak amacıyla hibrit bir algoritma önerilmiştir. Önerilen hibrit algoritmada, Parçacık Sürü Optimizasyonu (PSO) algoritmasının basit kullanımı ve güçlü küresel arama yetenekleri, Gri Kurt Optimizasyonu (GKO) algoritmasının güçlü keşif ve yerel minimumdan kaçınma özellikleriyle birleştirilmiştir. Önerilen hibrit yaklaşım, hem hesaplama doğruluğunu hem de işlem süresinde verimliliği sağlamayı hedeflemektedir. Hibrit yaklaşım kullanılarak, bilinmeyen bir ortamda sensörler yardımıyla rotalar hesaplanmıştır. Hibrit algoritmanın performansı, bireysel PSO ve GKO algoritmaları ile karşılaştırılmıştır. Karşılaştırma sırasında algoritmalar; optimum rotayı bulma süreleri, hesaplanan rota uzunluğu, gerekli iterasyon sayısı ve yerel minimumdan kaçınma yetenekleri açısından değerlendirilmiştir. Sonuçlar, özel olarak geliştirilmiş bir arayüz kullanılarak simüle edilmiş ve rota hesaplama süresi açısından önemli bir avantaj sağlandığı gözlemlenmiştir. Ayrıca, PSO yaklaşımında mevcut olan yerel minimum problemi başarılı bir şekilde ortadan kaldırılmış ve GKO yaklaşımına kıyasla iterasyon sayısı ile işlem süresi iyileştirilmiştir. Bu yaklaşımın, özellikle afet yönetimi senaryolarında fayda sağlaması beklenmektedir. Çünkü otonom insansız araçlar, arama, kurtarma ve kaynak dağıtımı için bilinmeyen veya engelli ortamlarda verimli rota planlaması yapılmasına yardımcı olabilir.

Anahtar kelimeler: Parçacık sürü optimizasyonu, Gri kurt optimizasyonu, Rota planlama, İnsansız hava aracı, Hibrit algoritma.

*Yazışılan yazar

Firat University Journal of Experimental and Computational Engineering

Optimizing Unmanned Vehicle Navigation: A Hybrid PSO-GWO Algorithm for Efficient Route Planning

Gökhan ALTUN [1], İlhan AYDIN [2*]

[1]Tatvan Vocational School, Bitlis Eren University, 13200, Bitlis, Türkiye.
[2]Computer Engineering Department, Faculty of Engineering, Firat University, 23119, Elazig, Türkiye.
[1]altungokhann@gmail.com, [2]iaydin@firat.edu.tr

Received: 14.06.2024
Accepted: 16.09.2024

Revision: 10.09.2024

doi: https://doi.org/10.62520/fujece.1501508
Research Article

Citation: G. Altun and İ. Aydın, "Optimizing unmanned vehicle navigation: a hybrid pso-gwo algorithm for efficient route planning", Firat Univ. Jour.of Exper. and Comp. Eng., vol. 4, no 1, pp. 100-114, February 2025.

Abstract

This study aims to address the route-planning problem for autonomous systems, which plays a significant role in the operation of unmanned vehicles. A hybrid algorithm has been proposed to enhance the performance of metaheuristic algorithm approaches used to solve the specified problem. In the hybrid algorithm, the simplicity and powerful global search capabilities of the Particle Swarm Optimization (PSO) algorithm are combined with the strong exploration and local minimum avoidance features of the Grey Wolf Optimization (GWO) algorithm. The proposed hybrid approach seeks to achieve both computational accuracy and efficiency in processing time. Using the hybrid approach, routes were calculated in an unknown environment with the help of sensors. The performance of the hybrid algorithm was compared with that of the standalone PSO and GWO algorithms. The comparison evaluated the algorithms based on their execution time for finding the optimal route, the length of the calculated route, the required number of iterations, and their ability to escape local minima. The results were simulated using a custom-built interface, demonstrating a significant advantage in terms of route calculation time. Furthermore, the local minimum problem inherent in the PSO approach was successfully mitigated, while the iteration count and processing time were improved compared to the GWO approach. This approach can be particularly beneficial in disaster management scenarios, where autonomous unmanned vehicles can assist in efficiently planning routes for search, rescue, and resource delivery in unknown or obstructed environments.

Keywords: Particle swarm optimization, Grey wolf optimization, Route planning, Unmanned aerial vehicle, Hybrid algorithm

[*]Corresponding author

Plagiarism Checks: Yes – Ithenticate
Complaints: iaydin@firat.edu.tr

1. Introduction

Advancements in hardware and software technology have created a growing need for automating tasks in various fields. As a result, we observe significant developments in industrial, agricultural, military, and daily life applications. Route planning plays a crucial role in enabling unmanned vehicles to operate autonomously. This problem has been a topic of interest for researchers for a long time, with the earliest studies tracing back to the Traveling Salesman Problem (TSP) introduced by William Hamilton [1]. However, as technology has advanced and the application areas of the TSP have expanded, new constraints and parameters have emerged, increasing the complexity of the problem. Initially, solutions relied on non-intuitive methods, but these approaches often came with significant computational costs and resource usage. The growing constraints and problem complexity have driven the search for more efficient and practical solutions. Consequently, heuristic, meta-heuristic, and deep learning methods have been developed as alternatives to traditional approaches.

For the route-planning problem, the environment is categorized into two types: known and unknown. In a known environment, route planning for unmanned vehicles is performed using approaches such as heuristic, meta-heuristic, and reinforcement learning algorithms. Additionally, strategies like Grid-Based Search, Random Search, Virtual Line-Based Search, and Line-Based Search have been developed to adapt these algorithms to the problem. By employing these methods, route information is preloaded into the unmanned vehicle, enabling its movement along the calculated path. In contrast, in an unknown environment, the unmanned vehicle must process real-time data using additional hardware components such as sensors and cameras. The data collected is used to calculate the direction and the longest possible movement by running the algorithm over the maximum area detectable by the sensors or cameras.

To address the routing problem in the unknown environment defined in this study, algorithms that are efficient in both processing time and resource consumption are essential. Since computational operations will be performed on UAVs, the algorithms must deliver fast responses while consuming minimal resources. Based on the literature review, PSO was chosen due to its simplicity, ease of implementation, and speed. However, PSO has notable limitations, including a tendency to get stuck in local minima and weak exploration capabilities. Şenel et al. [2] proposed a hybrid approach combining GWO and PSO algorithms for classical optimization problems. To prevent PSO from getting stuck in local minima, the GWO algorithm was run with a low population size and a limited number of iterations. Successful particles were then transferred to PSO to improve the solution. Kamboj [3] introduced a hybrid approach that combines GWO and PSO algorithms. In their study, the PSO algorithm was executed first, and the three best results were assigned as alpha, beta, and delta wolves for the GWO algorithm. This method was applied to optimize the timeline for coordinating energy production facilities. Mahapatra et al. [4] developed hybrid optimization methods to minimize total energy loss and reactive power loss in power planning. They proposed three hybrid techniques: linear weight declining PSO, constant inertia weight PSO, and a GWO-PSO hybrid approach. In the PSO-GWO hybrid approach, particles generated by GWO were transferred to PSO, which prevented PSO from getting stuck in local minima and led to an optimal solution. Singh and Singh [5] proposed the hybrid Particle Swarm Optimization and Grey Wolf Optimizer (HPSOGWO), which integrates the efficient exploitation of PSO with the robust exploration capabilities of GWO to improve convergence performance and address limitations such as PSO's susceptibility to local minima and GWO's weaker exploitation. Nguyen et al. [6] focused on speed control for a non-linear DC motor system. Their work involved optimizing parameters for a PID-type fuzzy logic controller using PSO, GWO, Cuckoo-GWO hybrid, and PSO-GWO hybrid algorithms. Negi et al. [7] proposed a PSO-GWO hybrid approach to address the reliability allocation and optimization problem for complex bridge systems and life-support systems in space capsules.Thobiani et al. [8] conducted a study to detect vertical and horizontal cracks in plates. GWO and PSO-GWO hybrid methods were used to tune the parameters of artificial neural networks. Gul et al. [9] proposed a PSO-GWO hybrid algorithm for solving the path planning problem required by autonomous guided robots. Their method aimed to improve performance in route planning by leveraging metaheuristic approaches. Liu and Wang [10] employed the Quantum Particle Swarm Optimization (QPSO) algorithm for dynamic route planning of UAVs. They used the rounding timed active area control method to avoid local minima. Chen et al. [11] utilized the advanced artificial potential field-based path planning algorithm for route planning of UAVs in dynamic environments. Their approach addressed the local minimum problem and conducted route planning in a 2D environment. Zhang et al. [12] solved the path planning problem for mobile robots using the advanced localized PSO algorithm. By modifying inertia weights, acceleration factors, and localization, they aimed to overcome the local minimum problem in the PSO algorithm. Xu et al. [13] used the Gravity Search Algorithm (GSA) for UAV route planning. Due to the low performance of the standalone approach, they explored a hybrid method. However, the hybrid solution increased the cost, and the problem was addressed by controlling the convergence rate on the gravitional threshold parameter for GSA. Tang et al. [14] performed dynamic route planning for multiple robots in unknown terrain using the GWO algorithm. Their approach enabled the robots to reach fixed and moving targets without hitting obstacles, relying on sensors to control specific areas in unknown environments. He et al. [15] addressed UAV route planning using a deep reinforcement learning method. Their study simulated obstacle-filled environments using AirSim, where sensors on the UAVs guided route planning. Garip et al. [16] proposed a hybrid approach for mobile robot route planning. The outputs of the cuckoo search, PSO, and firefly algorithms were utilized as inputs for other algorithms to enhance route planning performance. Yılmaz and Aydoğmuş [17] tackled the route planning problem for an unmanned vehicle in a 3D environment. They utilized the CoppeliaSim simulator to implement the deep deterministic policy gradient algorithm. Their study reported up to 80% success but noted shortcomings in environmental perception. Sun et al. [18] developed a hybrid algorithm to solve the route planning problem in environments with fixed obstacles. Their hybrid approach combined the ant swarm algorithm and the intelligent water drop algorithm. Routes were evaluated based on path length and the ability to avoid restricted areas. Wan et al. [19] applied the Advanced Whale Optimization Algorithm (AWOA) and the Dynamic Artificial Potential Field (DAPF) method for route planning in dynamic environments. AWOA was used for global path planning, while DAPF helped avoid moving obstacles.

This paper presents a novel hybrid PSO-GWO algorithm for route planning in the autonomous navigation of unmanned vehicles. By leveraging the strengths of hybrid approaches, the proposed algorithm combines the simplicity and speed of PSO with the robust exploration and local minimum avoidance capabilities of GWO. The study aims to solve the route-planning problem with improved computational efficiency and accuracy, providing innovative solutions for dynamic, obstacle-filled, and unknown environments. Comprehensive performance evaluations demonstrate that the proposed algorithm offers significant advantages over existing methods in terms of computation time, route length, and the number of iterations required.

2. The Proposed Hybrid PSO-GWO Algorithm for Route Planning

One of the key challenges in enabling autonomous functionality for UAVs and other unmanned vehicles is route planning. Route planning can be classified into two categories: static and dynamic. Static route planning involves calculating a path to the target within a known field before movement begins, ensuring that the vehicle avoids obstacles and prohibited zones. In this case, the computational cost increases logarithmically with the number of obstacles in the field. Dynamic route planning, on the other hand, involves creating a route in a previously unknown field using sensors or cameras to scan a limited range around the vehicle. Since the field structure and obstacle locations are not known in advance, the algorithm must calculate the shortest route while minimizing computation time, often relying on random movements to navigate the environment. For unmanned vehicles, the route must be as short as possible. However, when the environment is unknown and the route is generated dynamically, the importance of computation time increases significantly. Any delay in calculation directly impacts the total time required to reach the target, which in turn increases fuel consumption. In dynamic route planning, both total travel time and route length are critical factors for reducing fuel consumption and avoiding issues related to fuel shortages. Metaheuristic algorithm approaches are widely used in route planning due to their efficiency and adaptability. Various metaheuristic algorithms have been developed, many of which have successfully solved a wide range of problems. However, each algorithm has its own strengths and weaknesses. Among the most widely used metaheuristic methods in recent years are the PSO and GWO algorithms. PSO is known for its simplicity, ease of implementation, and swarm-based approach [20]. One significant drawback, however, is its tendency to become trapped in local minima, which can impede its search for the optimal solution. In contrast, the GWO algorithm excels in exploration and has demonstrated strong performance in avoiding local minima [21]. However, GWO's balance between exploration and exploitation is somewhat limited and dependent on specific conditions [22]. In this study, the exploration capability of the PSO algorithm is combined with the exploitation strength of the GWO algorithm. In the problem-solving process, PSO begins by assigning a random position to each particle in the swarm. The solution search continues until either a predefined iteration limit or a termination criterion is reached. During this process, the global best solution (gbest*) represents the optimal result obtained by the entire swarm, while the personal best solution ($p$best*) denotes the best result discovered by each individual particle. Each particle's velocity is updated based on these best solutions, as illustrated in (1), and its position is then updated using the newly computed velocity vector, as given in (2).

$$v_{i+1} = w * v_i + c_1 * rand_1 * (pbest - x_i) + c_2 * rand_2 * (gbest - x_i) \tag{1}$$

$$X_{i+1} = x_i + v_{i+1} \tag{2}$$

In (1), the values of c_1 and c_2 are fixed constants randomly selected between 0 and 2. Similarly, $rand_1$ and $rand_2$ are randomly generated values between 0 and 1, while w represents the inertia constant, selected within the range of 0 to 2.

The GWO algorithm is a nature-inspired metaheuristic method that replicates the leadership hierarchy and hunting strategies of grey wolves in their natural environment. It models the hunting process through three key phases: encircling the prey, attacking the prey (exploitation), and searching for the prey (exploration). These phases enable the algorithm to maintain a balance between global exploration and local exploitation, thereby achieving efficient optimization. During the hunt, grey wolves exhibit encircling behavior around their prey. This behavior is mathematically represented using (3)-(4).

$$D = \left| C.X_{p(t)} - X_{(t)} \right| \tag{3}$$

$$X_{t+1} = \left| X_{p(t)} - A.D \right| \tag{4}$$

In (3)-(4), t represents the current iteration. The parameters A and C denote the coefficient vectors, while X and X_p represent the position vectors of the grey wolf and the prey, respectively. A and C values are calculated by using (5) and (6).

$$A = |2ar_1 - a| \tag{5}$$

$$C = |2r_2| \tag{6}$$

The r_1 and r_2 values used in (5)-(6) are randomly generated values between 0 and 1. The value of a is shown with a linearly decreasing value from 2 to 0 by using (7).

$$a = 2.\left(1 - {}^{i}/_{iteration}\right) \tag{7}$$

In the algorithm, after the prey is encircled, the hunting phase begins. The positions of the wolves in the herd are updated using Equations (8) to (14), which build upon the containment formulas (3) and (4). These updates are guided by the positions of the alpha, beta, and delta wolves, which are the closest to the current position of the prey.

$$D_\alpha = \left| C.X_\alpha - X_{(t)} \right| \tag{8}$$

$$D_\beta = \left| C.X_\beta - X_{(t)} \right| \tag{9}$$

$$D_\delta = \left|C.X_\delta - X_{(t)}\right| \quad (10)$$

$$X_1 = |X_\alpha - A.D_\alpha| \quad (11)$$

$$X_2 = \left|X_\alpha - A.D_\beta\right| \quad (12)$$

$$X_3 = |X_\alpha - A.D_\delta| \quad (13)$$

$$X_{t+1} = \frac{(X_1 + X_2 + X_3)}{3} \quad (14)$$

As the value of a approaches zero during the optimization, the hunt transitions into the attack phase, whre coefficient vector A takes random values within the range of $[-1,1]$. This progression allows the steps of encircling the prey, hunting, and attacking to be executed sequentially. In this study, a hybrid approach is proposed by integrating the strong exploration capabilities of the GWO with the efficient exploitation strengths of the PSO. The flowchart of the hybrid algorithm is shown in Figure 1.

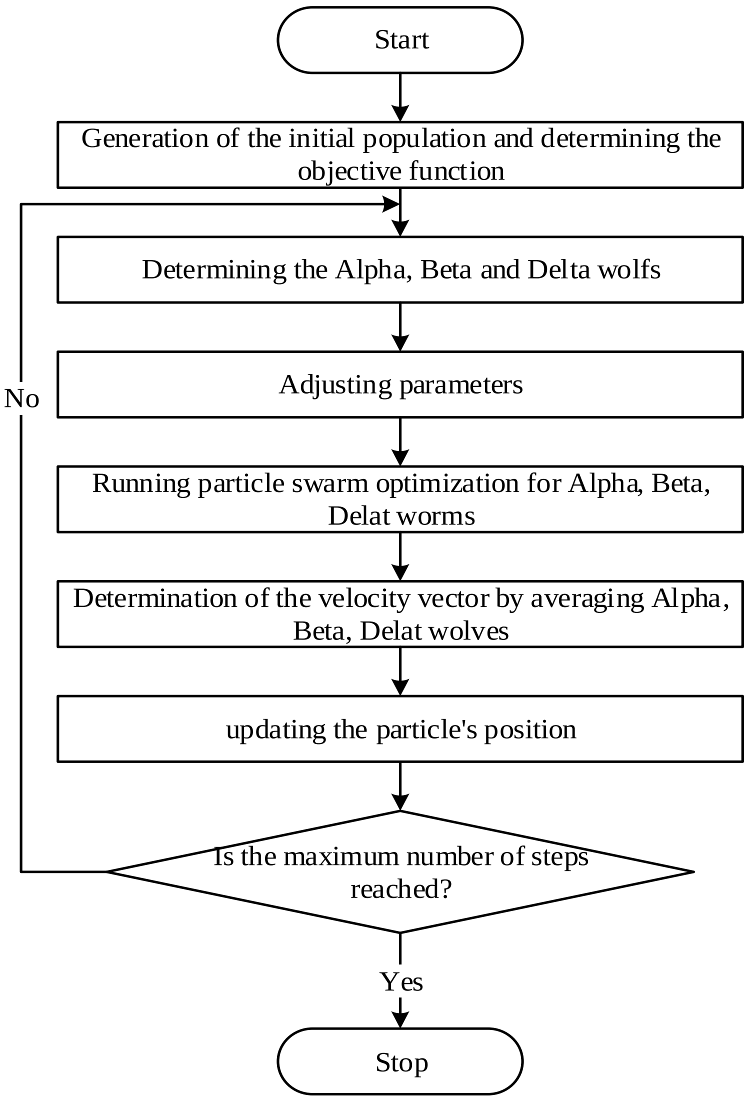


Figure 1. Flowchart Diagram of Hybrid Algoritm

The formula for calculating the parameter a in the GWO algorithm has been adapted to the constant w to regulate the oscillation of the velocity vector during calculations as shown in (15). The coefficient c_1 used in the velocity vector calculation is determined using (16).

$$w = 1 - {}^{i}/_{iteration} \tag{15}$$

$$c_1 = 4 - (2 * w * random(0,1)) \tag{16}$$

With the proposed approach, the influence of the velocity vector decreases as the number of iterations increases, keeping oscillation under control. In the hybrid approach, the exploration capability of GWO and the exploitation features of PSO are combined. In the GWO algorithm, the velocity updates for the alpha, beta, and delta wolves are performed using (17) to (19). Subsequently, the overall velocity is calculated as mean values of three wolves using (20), and the position of each particle is updated based on (21).

$$v_{i\alpha}^{t+1} = w.v_i^t + r_1.c_1.(x_\alpha - x_i^t) \tag{17}$$

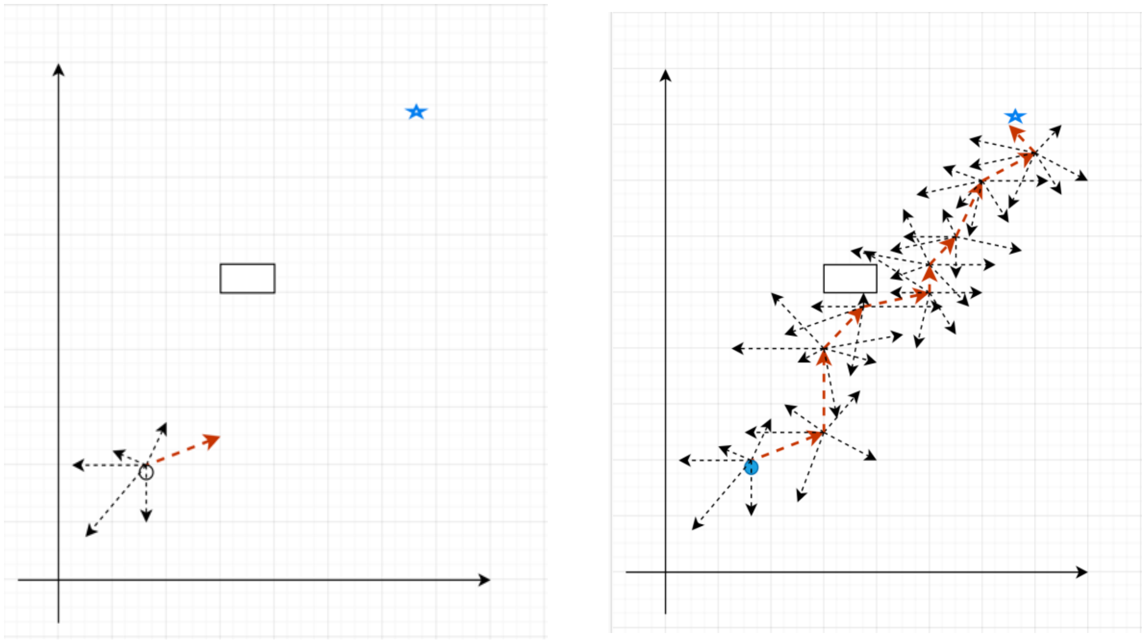
$$v_{i\beta}^{t+1} = w.v_i^t + r_1.c_1.\left(x_\beta - x_i^t\right) \tag{18}$$

$$v_{i\delta}^{t+1} = w.v_i^t + r_1.c_1.(x_\delta - x_i^t) \tag{19}$$

$$v_i^{t+1} = \frac{\left(v_{i\alpha}^{t+1} + v_{i\beta}^{t+1} + v_{i\delta}^{t+1}\right)}{3} \tag{20}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{21}$$

A balanced exploration capability is achieved by using the best three particles in the population and averaging the three newly obtained velocity vectors. To evaluate the efficiency of the proposed hybrid algorithm, the route planning problem for UAVs is addressed. The route planning is performed in an unknown environment, where UAV relies on sensors to explore the area and execute the routing process. The maximum area detectable by the sensor is assumed to have a diameter of 20 units. This means the UAV can identify obstacles within a 20-unit detection range. The direction and movement distance are calculated based on the available information. If the calculated movement exceeds the sensor’s maximum detectable range, the movement is adjusted by taking the modulus of the calculated value. The maximum value is avoided because the proximity of obstacles outside the detection range is unknown. The goal is to avoid restricting the mobility of unmanned vehicles. In the route planning problem for UAVs, the search strategy used alongside the selected algorithm is crucial. The chosen search strategy must be compatible with the algorithm and applicable to the field structure being studied. The selected search algorithm and strategy play a significant role in route planning, influencing computation time, result performance, and the accuracy of algorithm. In this study, a random search strategy is combined with a metaheuristic algorithm for operation in an unknown environment. In the random search strategy, the step length is determined first. From the UAV’s current position, random points are generated within the maximum step length using Euclidean calculations. For these points to be valid, the UAV must not collide with obstacles or pass through restricted areas between its current position and the new position. Figure 2 illustrates the Random Search Strategy for an UAV navigating an environment with obstacles.



(a) Initial State **(b)** Latest State

Figure 2. Random Search Strategy

In Figure 2(a), the UAV starts at its initial position and generates random points (indicated by arrows), selecting the point closest to the target (blue star) based on Euclidean distance. In Figure 2(b), the UAV progresses toward the target by iteratively updating its position while avoiding collisions with obstacles (rectangles). The selected path is shown in red, demonstrating how the UAV dynamically moves closer to the target without crossing restricted areas.

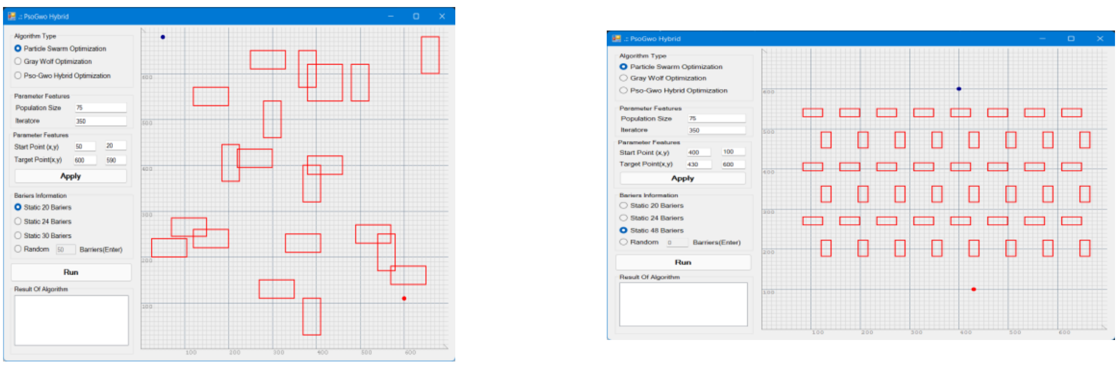
3. Application Results

The fundamental working principle of the proposed hybrid algorithm is based on the velocity update mechanism of the PSO algorithm, combined with the selection of the top three wolves in the GWO algorithm, and an iteration-based weight reduction mechanism approaching zero. To assess the effectiveness and superiority of this method, the task of discovering a randomly generated point within a 1000x1000-scale space was evaluated based on iteration count, computation time, and distance. The evaluation was conducted by comparing the proposed hybrid algorithm with standalone PSO and GWO algorithms. In the existing literature, iteration count and distance are commonly used as performance metrics. However, since computation time is critical for real-world applications such as the dynamic routing problem in UAVs, it was included as an additional evaluation criterion. To ensure a fair comparison, all experiments were performed with the same population size. Each test function was executed independently 15 times for each algorithm. The population size was fixed at 20, and the maximum number of iterations was limited to 200. The evaluation criteria included the minimum values of computation time and distance, with success defined as the step at which the iteration value fell below one unit. In the results, red markers denote local minima, while bold values indicate successful outcomes for the corresponding tests. The results are summarized in Table 1.

Table 1. Result of Testing

Processing time			Distance			Number of iterations		
PSO	GWO	Hybrid	PSO	GWO	Hybrid	PSO	GWO	Hybrid
325	513	391	0.528660654	0.05543581	**0.00036105**	138	194	**7**
276	505	386	0.000478167	0.04297425	**3.8913E-05**	40	192	**11**
275	462	416	1.719389262	0.31649431	**2.3391E-06**	31	192	**13**
697	493	**400**	10.73632129	0.16824665	**1.4392E-06**	93	182	**12**
282	497	414	7.3996E-06	0.19867104	**4.384E-07**	23	196	**12**
284	478	385	0.000419992	0.3027555	**1.2511E-05**	45	194	**15**
287	461	400	11.58451303	0.16900177	**0.00012228**	14	165	**10**
289	481	383	8.357593682	0.01555127	**0.00137818**	34	112	**12**
294	478	409	5.152039051	0.40561707	**0.00173228**	61	191	**10**
249	465	372	**5.68434E-14**	0.05788538	0.00029738	10	91	10
277	489	371	0.199539361	0.21778345	**0.00135569**	106	78	**8**
283	494	402	0.397384449	0.17952916	**0.00048415**	46	184	**4**
277	493	365	**0.001096472**	0.4388497	0.02923901	68	190	**11**
290	473	412	**2.44817E-08**	0.2689691	0.09687925	25	180	**8**
280	487	414	0.102694885	0.16521049	**0.00785485**	60	188	**12**

As shown in Table 1, PSO exhibits a local minimization problem, while GWO is computationally expensive in terms of computation time. The proposed hybrid algorithm, however, demonstrates successful outcomes in terms of both exploitation and iteration. In this study, two different field structures (regular and irregular) and varying numbers of obstacles were considered as part of a real-world problem. The fields for simulation were created using the C# programming language and the drawing library. The simulated field was scaled to a 600x600 unit structure, resulting in a total field area of 360,000 units^2. The first field structure is an irregular layout containing 20 nested obstacles of varying sizes. These nested obstacles were designed to test the algorithms'ability to handle local minimum problems during the route computation process. The second field structure is a regular layout, consisting of 48 obstacles of uniform size arranged in six rows, with eight obstacles per row. The obstacles in each row were offset to fill the gaps between obstacles in the preceding row. The visual representations of the field structures created for both scenarios are provided in Figure 3.

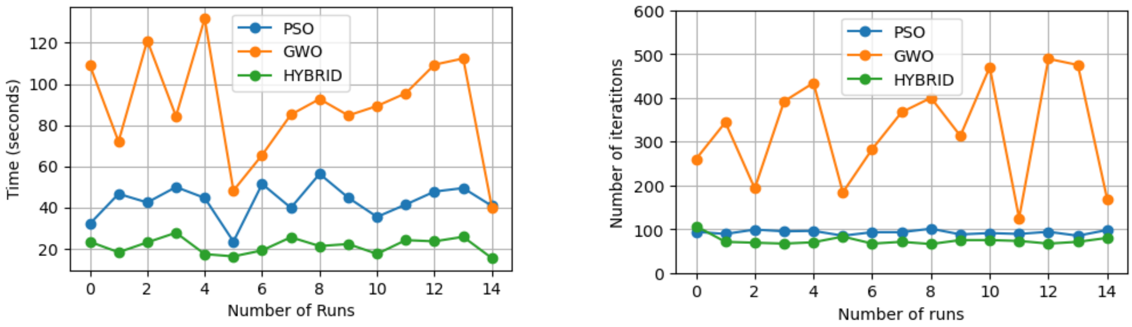


(a) Field structure with 20 obstacles

(b) Field structure with 48 obstacles

Figure 3. Field structure

The algorithms will be evaluated based on criteria such as avoiding local minima, finding the global optimal route, and determining the route in the shortest possible time in environments with fixed obstacles. In real-world scenarios, unmanned vehicles often lack prior knowledge of the area they are navigating. Instead, they rely on integrated cameras and sensors to generate routes and make movement decisions. These systems analyze the area within a specified radius before taking action. To avoid collisions with obstacles during movement, the route creation process must be instantaneous and highly efficient. In this study, a UAV is tasked with generating a route in a previously unknown field structure by processing data received from its sensors. The sensors are assumed to detect obstacles within a maximum range of 20 units. The hybrid algorithm will be used to determine both the direction of movement and the step size until the next sensor query. The initial field structure consists of 20 obstacles, some of which are nested, with a Euclidean distance of 790 units between the starting and ending points. The population size for the algorithms is set at 75, and the iteration count is limited to 500. To ensure a fair comparison, all algorithms were executed 15 times with identical population parameters under the specified criteria. The results, including the processing time for each test step, are visualized in Figure 4.

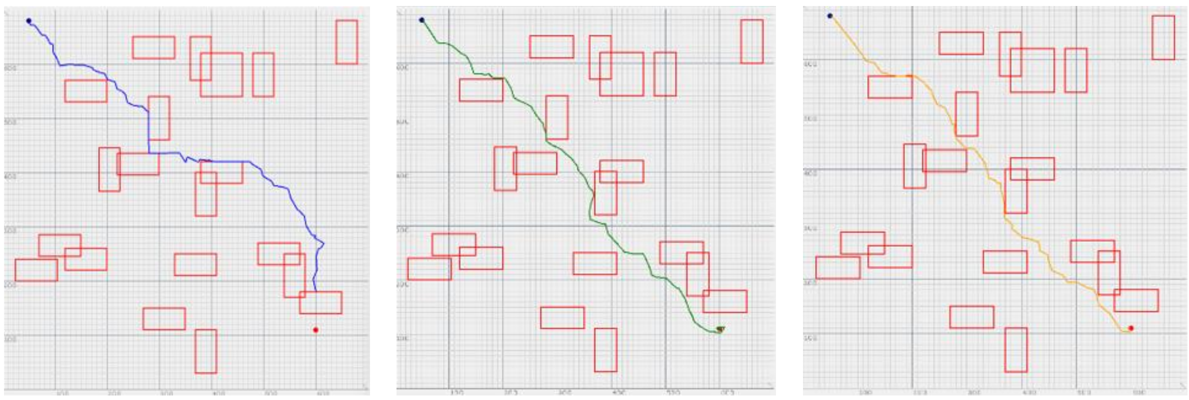


(a) Processing time

(b) Number of Iterations

Figure 4. Performance comparisons of three algorithms for twenty obstacles

As illustrated in Figure 4, the proposed hybrid algorithm outperforms the other algorithms in terms of runtime and the number of iterations. The hybrid algorithm demonstrates a shorter runtime compared to PSO and converges more quickly to the optimal solution. Figure 5 provides a visual representation of the routes identified by each algorithm.



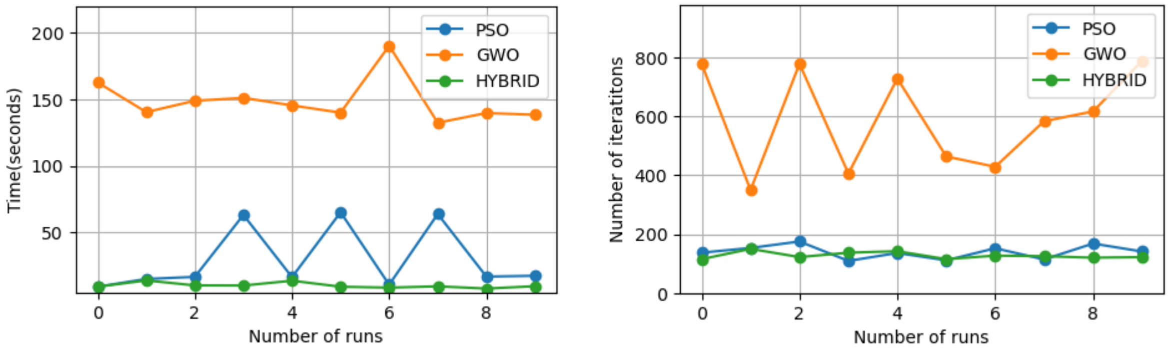
(a) PSO

(b) Hybrid

(c) GWO

Figure 5. Route of each algorithm for twenty obstacles

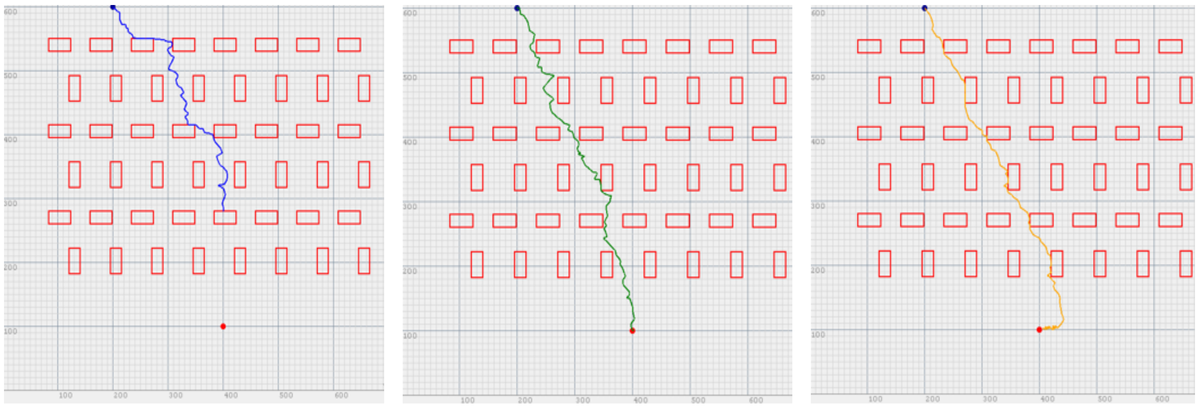
In Figure 5, it can be observed that the PSO algorithm becomes trapped in local minima due to the nested obstacles in the field structure. In contrast, the proposed hybrid approach not only provides an advantage in terms of processing time but also successfully avoids local minima. Although the GWO algorithm excels in exploration and exploitation, it is more costly in terms of computation time and iterations compared to the hybrid approach. The second field structure comprises 48 obstacles, with a Euclidean distance of 538 units between the starting and ending points. This structure consists of 6 rows of 8 obstacles arranged side by side, designed such that the gaps between obstacles in one row align with the obstacles in the adjacent row. The population size for this scenario was set to 75, and the iteration count was defined as 800. To ensure a fair comparison, all algorithms were executed 10 times using the same population parameters under the specified criteria. The processing time and number of iterations are given in Figure 6 for 10 execution.



(a) Processing Time **(b)** Number of Iterations

Figure 6. Performance comparisons of three algorithms for forty-eight obstacles

Figure 6 (a) demonstrates that the hybrid algorithm outperforms the other algorithms in terms of runtime, consistently achieving faster execution. For the PSO algorithm, the runtime was significantly higher in three of the runs, primarily due to its tendency to become trapped in local minima. In Figure 6 (b), the hybrid algorithm and PSO show comparable performance in terms of the number of iterations. However, PSO converged to a local minimum in three out of ten runs, highlighting its limitations in complex scenarios. Figure 7 illustrates the routes generated by the three algorithms when navigating through the 48-obstacle field.



(a) PSO **(b)** Hybrid **(c)** GWO

Figure 7. Route of each algorithm for twenty obstacles

In Figure 7, the PSO algorithm reached a local minimum in 3 out of 10 runs. When the time spent on local minimization is excluded from the average runtime table, a corrected value of 14.7 seconds is obtained. Even with this adjustment, the hybrid approach remains advantageous. Compared to the GWO algorithm, the hybrid approach demonstrates superiority not only in time performance but also in path distance optimization. Figure 8 presents the path distance results.

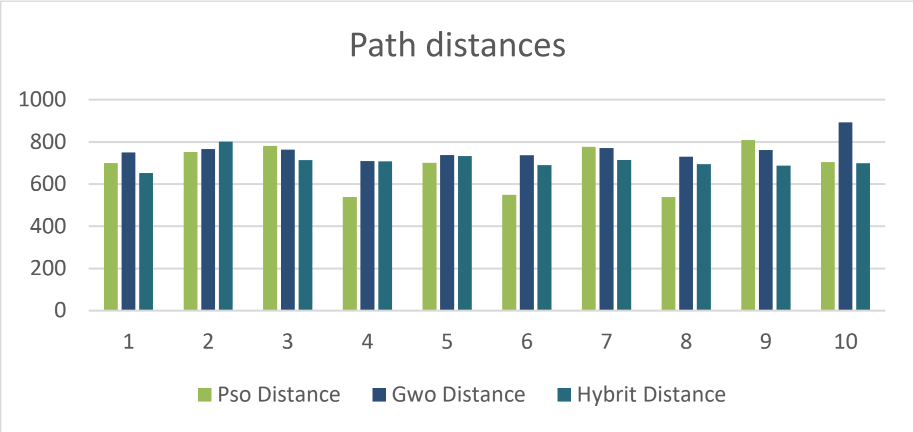
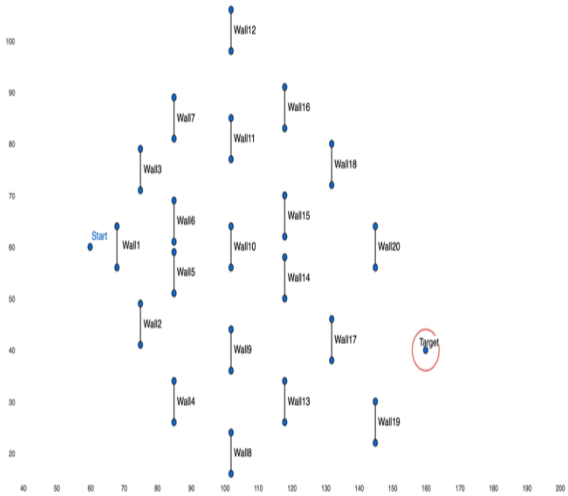
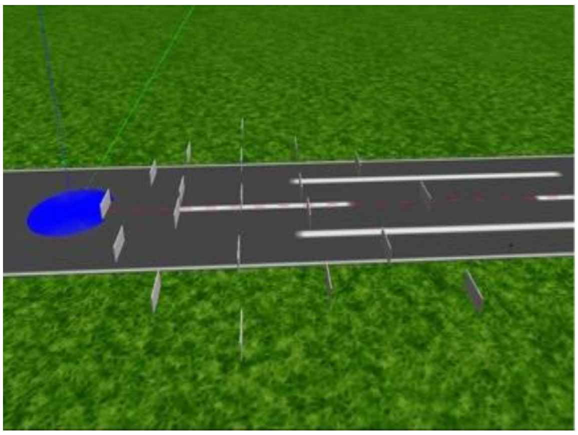


Figure 8. 48 road distances measured during tests in an obstacle field

In Figure 8, the hybrid approach demonstrated a clear advantage in path distance, achieving superior results in 7 out of 10 tests. The test results obtained in the 2D environment were further validated in a 3D simulation environment to ensure their accuracy in real-world applications. The simulations were conducted using the Gazebo application as the simulation platform. Data from the virtual environment created in Gazebo was collected using a LiDAR sensor, which was managed through ROS (Robot Operating System) software.In the simulation environment designed for UAV testing, 20 walls (dimensions: 7x5x0.2 meters) were added. The Euclidean distance between the starting point and the target point was set at 102 meters. Both 2D and 3D visualizations of the simulation environment are shown in Figure 9.



(a) 2D Environment



(b) 3D Environment

Figure 9. Gazebo Simulation Environment

After the drone takes off, the hybrid algorithm determines the direction and magnitude of movement based on data received from the LiDAR sensor, which has a measurement range of 10 meters. The 2D route plan for the autonomous flight, managed by the hybrid algorithm, is presented in Figure 10.

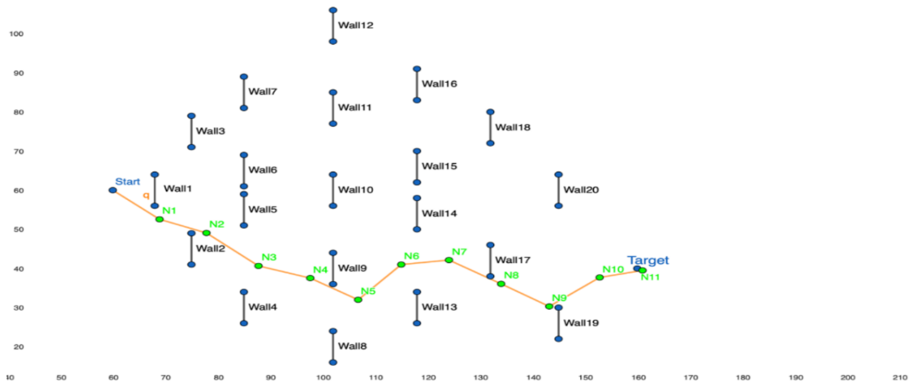


Figure 10. Autonomous flying

The route of the UAV, as depicted in the Figure 10, demonstrates the efficiency of the hybrid algorithm in navigating a complex environment with multiple obstacles. Starting from the initial position, the UAV successfully maneuvers around 20 walls while maintaining an optimized path toward the target. The numbered nodes illustrate key decision points where the algorithm determined the UAV's next direction and step size based on LiDAR sensor data. The trajectory is smooth and avoids unnecessary detours, highlighting the algorithm's ability to balance exploration and exploitation effectively. The UAV's ability to reach the target within a proximity of 0.23 meters and a total route length of 130 meters further validates the precision and adaptability of the hybrid approach in a constrained environment.

4. Conclusions

The test results from this study highlight the significant advantages of the proposed hybrid algorithm. Its structure leverages the simplicity, proven robustness, and exploitation capabilities of PSO, combined with the exploration strength and local minimum avoidance properties of GWO. This combination makes the hybrid algorithm particularly effective in computations where processing time is critical. For route planning—an essential aspect of unmanned vehicle operations—this approach proves valuable due to its efficiency in navigating complex field structures, minimizing processing time, and avoiding local minima. To further validate its real-world applicability, the algorithm was tested in a three-dimensional Gazebo simulation environment for autonomous UAV flight, yielding successful results. The proposed approach demonstrates readiness for real-world implementation. Future studies aim to extend its application to complex and irregular environments, such as disaster zones and conflict areas, to further enhance the efficiency and utility of unmanned aerial vehicles in challenging scenarios.

5. Acknowledgements

This study was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Project No. 123E669.

6. Author Contribution Statement

G. Altun was responsible for the conceptualization, methodology, software development, and writing of the original draft. I. Aydın contributed to the conceptualization, methodology, writing of the original draft, and the review and editing of the manuscript.

7. Ethics Committee Approval and Conflict of Interest

No ethics committee approval was required for this study and there is no conflict of interest between the authors.

8. References

[1] N. L. Biggs, E. K. Lloyd, and R. J. Wilson, Graph Theory, 1736-1936, 1986.
[2] F. A. Şenel, F. Gökçe, A. S. Yüksel, and T. Yiğit, "A Novel Hybrid PSO–GWO Algorithm for Optimization Problems," Engineering with Computers, vol. 35, no. 4, pp. 1359–1373, Dec. 2018.
[3] V. K. Kamboj, "A Novel Hybrid PSO–GWO Approach for Unit Commitment Problem," Neural Computing and Applications, vol. 27, no. 6, pp. 1643–1655, Jun. 2015.
[4] S. Mahapatra, M. Badi, and S. Raj, "Implementation of PSO, it's variants and Hybrid GWOPSO for improving Reactive Power Planning," in 2019 Global Conference for Advancement in Technology (GCAT), pp. 1–6.
[5] N. Singh and S. B. Singh, "Hybrid Algorithm of Particle Swarm Optimization and Grey Wolf Optimizer for Improving Convergence Performance," Journal of Applied Mathematics, vol. 2017, p. e2030489, Nov. 2017.
[6] D.-T. Nguyen, J.-R. Ho, P.-C. Tung, and C.-K. Lin, "A Hybrid PSO–GWO Fuzzy Logic Controller with a New Fuzzy Tuner," International Journal of Fuzzy Systems, vol. 24, no. 3, pp. 1586–1604, Nov. 2021.
[7] G. Negi, A. Kumar, S. Pant, and M. Ram, "Optimization of Complex System Reliability Using Hybrid Grey Wolf Optimizer," Decision Making: Applications in Management and Engineering, vol. 4, no. 2, pp. 241–256, Oct. 2021.
[8] A. Thobiani, S. Khatir, B. Benaissa, E. Ghandourah, S. Mirjalili, and A. Wahab, "A hybrid PSO and Grey Wolf Optimization algorithm for static and dynamic crack identification," Theoretical and Applied Fracture Mechanics, vol. 118, p. 103213, 2022.
[9] F. Gul, W. Rahiman, S. S. N. Alhady, A. Ali, I. Mir, and A. Jalil, "Meta-heuristic Approach for Solving multi-objective Path Planning for Autonomous Guided Robot Using PSO–GWO Optimization Algorithm with Evolutionary Programming," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 7, pp. 7873–7890, Sep. 2020.
[10] B. Liu and X. Wang, "UAV Route Planning Based on QPSO Algorithm under Rolling Time Domain Control," in 2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), pp. 612–617.
[11] S. Chen, Z. Yang, Z. Liu, and H. Jin, "An improved artificial potential field based path planning algorithm for unmanned aerial vehicle in dynamic environments," in 2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), pp. 591–596.
[12] L. Zhang, Y. Zhang, and Y. Li, "Mobile Robot Path Planning Based on Improved Localized Particle Swarm Optimization," IEEE Sensors Journal, vol. 21, no. 5, pp. 6962–6972, 2021.
[13] H. Xu, S. Jiang, and A. Zhang, "Path Planning for Unmanned Aerial Vehicle Using a MixStrategyBased Gravitational Search Algorithm," IEEE Access, vol. 9, pp. 57033–57045, 2021.
[14] H. Tang, W. Sun, A. Lin, M. Xue, and X. Zhang, "A GWO based multirobot cooperation method for target searching in unknown environments," Expert Systems with Applications, vol. 186, p. 115795, 2021.
[15] L. He, N. Aouf, and B. Song, "Explainable Deep Reinforcement Learning for UAV autonomous path planning," Aerospace Science and Technology, vol. 118, p. 107052, 2021.
[16] Z. Garip, D. Karayel, and M. Erhan Çimen, "A Study on Path Planning Optimization of Mobile Robots Based on Hybrid Algorithm," Concurrency and Computation: Practice and Experience, vol. 34, no. 5, Nov. 2021.
[17] T. Yılmaz and Ö. Aydoğmus, "Deep Deterministic Policy Gradient Reinforcement Learning for collision-free Navigation of Mobile Robots in Unknown Environments," Firat University Journal of Experimental and Computational Engineering, vol. 2, no. 2, pp. 87–96, Jan. 2023.
[18] X. Sun, S. Pan, N. Bao, and N. Liu, "Hybrid Ant Colony and Intelligent Water Drop Algorithm for Route Planning of Unmanned Aerial Vehicles," Computers & Electrical Engineering, vol. 111, pp. 108957–108957, Nov. 2023.
[19] R. Wan, X. Wang, and Z. Ma, "UAV route planning based on improved whale optimization algorithm and dynamic artificial potential field method," in 2023 6th International Symposium on Autonomous Systems (ISAS), pp. 1–8.

[20] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in MHS’95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, pp. 39–43.
[21] S. Mirjalili, Seyed Mohammad Mirjalili, and A. Lewis, "Grey Wolf Optimizer," Advances in Engineering Software, vol. 69, pp. 46–61, 2014.
[22] B. N. Gohil and D. R. Patel, "A hybrid GWOPSO Algorithm for Load Balancing in Cloud Computing Environment," in 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), pp. 185–191.