AUSUD BANDIRMA ONYEDİ EYLÜL ÜNİVERSİTESİ JITSA

Research Article

Advanced Multi-Robot Path Planning Based on Grey Wolf and Teaching-Learning Based Optimisation

Oğuz MISIR[1*]

[1] Bursa Technical University, Engineering and Natural Science Faculty, Department of Mechatronics Engineering, Bursa 16310, Türkiye

*Correspondence: oguz.misir@btu.edu.tr

DOI: 10.51513/jitsa.1608792

Abstract: Multi-robots stand out for their flexibility, scalability, and robustness in complex tasks by collaborating. Rather than a single robot undertaking a task, many robots can perform one or more tasks, which increases the task efficiency. Mobile robots require path planning to reach the targeted locations while working in areas such as service, logistics, agriculture, and production. This situation is also valid for multi-robots. In this study, an advanced multi-robot path planning method adapted to the path planning of multi-robots is proposed by combining the advantageous aspects of the Grey Wolf Optimization algorithm and the Teaching and Learning Based Optimization algorithm for the path planning of multi-robots. The aim of the study is to develop a method that can solve the path planning required by mobile robots in their tasks in a more efficient and high performance way. The proposed method was compared with other algorithms. Simulations containing combinations of population numbers, robot numbers, and different environments were applied. The proposed method shows high performance compared to other methods in simulations applied to the multi-robot path-planning problem. According to the comparison results, the proposed method showed high performance in terms of parameter results, such as reaching a faster solution, closing to the target, and total fitness values used in the evaluation of the robot team.

Keywords: Multi-robot, Path planning, Teaching-learning, Grey wolf optimization

Gri Kurt ve Öğretme-Öğrenme Tabanlı Optimizasyona Dayalı Gelişmiş Çoklu-Robot Yol Planlaması

Özet: Çoklu robotlar, işbirliği yaparak karmaşık görevlerde esneklik, ölçeklenebilirlik ve gürbüzlük özellikleriyle ön plana çıkmaktadır. Tek bir robotun bir görevi üstlenmesinden ziyade birçok robot bir veya birden fazla görevi üstlenebilir ve bu durum görev verimliliğini artırmaktadır. Mobil robotların servis, lojistik, tarım, üretim gibi alanlarda görev alırken hedeflenen konumlara gidebilmeleri için bir yol planlamasına ihtiyaç duyarlar. Bu durum çoklu robotlar içinde geçerlidir. Bu çalışmada Çoklu robotların yol planlaması için Gri Kurt Optimizasyonu algoritması ile Öğretme ve Öğrenme Tabanlı optimizasyon algoritmasının avantajlı yönleri birleştirilerek çoklu robotların yol planlamasına uyarlanan gelişmiş çoklu robot yol planlaması yöntemi önerilmektedir. Önerilen gri kurt optimizasyon tabanlı diğer algoritmalar ile karşılaştırılmaktadır. Popülasyon sayısı, robot sayısı ve farklı ortamlar kombinasyonlarını içeren simülasyonlar uygulanmıştır. Önerilen yöntem, çoklu robot yol planlaması probleminde uygulanan simülasyonlarda diğer yöntemlere kıyasla yüksek performans göstermektedir. Karşılaştırma sonuçlarına göre önerilen yöntem, daha hızlı çözüme ulaşma, hedefe yakınsama ve robot takımının değerlendirilmesinde kullanılan toplam uygunluk değerleri gibi parametre sonuçlarında yüksek performans göstermiştir.

Anahtar Kelimeler: Çoklu robot, Yol planlama, Öğretme-öğrenme, Gri kurt optimizasyon

1. Introduction

In recent years, the widespread use of mobile robots in many areas such as defense, transportation, home robotics, health, industry and agriculture has attracted attention(Chakraa et al., 2023). It is also observed that mobile robots take the workload of humans and fulfil many tasks without the need for human assistance. Due to the tasks they are used for, mobile robots need to explore their environment and plan

ORCID: 0000-0002-3785-1795
Received 28.12.2024; Received in revised form 14.02.2025; Accepted 19.02.2025

a path (Shoeib et al., 2024). Path planning is the task of a mobile robot to safely reach a target point from a specified starting point by avoiding obstacles. The integration of robotic technologies with informatics and artificial intelligence has provided robots with the ability to cooperate (Zhu & Zhang, 2021). Their ability to collaborate allows multi robots to perform one or more complex collaborative tasks. It is both time consuming and tiring a single robot to perform a complex task. Collaboration between multiple robots is more efficient than that of a single robot (Cao et al., 2023). Simultaneously, they are both capable and robust in accomplishing tasks. Multi-robots attract attention owing to their interaction, distributed structure, and cooperation features. They can fulfil complex tasks in a scalable manner (Nedjah & Junior, 2019). The scalable structure in challenging tasks means that even if a failure occurs to one or more of the robots, the fulfilment of the task is not affected.

Path planning is one of the main tasks of multi-robot as in mobile robots. Multi-robot path planning (MRPP) is a problem in which a solution is sought for multiple robots to reach predetermined targets simultaneously (Kumar & Sikander, 2024). The aim of MRRP is to allow robots to reach a target by avoiding obstacles to themselves and their surroundings. This is more difficult than the path planning of a single mobile robot and requires interaction (Li & Yang, 2020).

Path planning has been investigated as graph-based, traditional and meta-heuristic approaches. In addition to these approaches, artificial intelligence, hybrid solutions and many solutions involving different techniques are among the path planning methods (Lin et al., 2022). Graph-based approaches determine the path on a graph consisting of nodes and edges. Dijkstra, A*, RRT and RRT* algorithms are known graph-based path planning methods(Tan et al., 2021).

Conventional approaches are based on mathematical modelling and optimization techniques. Potential Field Methods, Dynamic Programming and Gradient Based Methods are among the traditional methods (Qin et al., 2023). Meta-heuristic based optimization methods, which are used in solving difficult optimization problems, are also used in solving problems with large search space such as path planning. Meta-heuristic based methods are used to reach the solution in a fast and optimized way. Algorithms such as Ant colony optimization, artificial bee colony and differential evolution have been used in path planning (Mittal et al., 2022). Hybrid approaches that combine the advantages of more than one method are among the techniques sought for solutions in path planning. Path planning algorithms that combine the advantages of heuristic and graph-based methods, RRT (Rapidly exploring Random Tree), artificial intelligence, and hybrid-based methods that combine the advantages of optimization methods are methods that are sought as solutions. Recently, various artificial intelligence methods have emerged to investigate path planning. These methods are based on machine, deep learning, and reinforcement learning (Apuroop et al., 2021).

The methods used in path planning in MRPP are also used in conventional path planning. However, the topologies of communication and decision making between robots in the MRPP may differ. Communication between robots can be categorized as a coupling approach, in which robots are in constant interaction with each other, and a decoupling approach, in which there is no direct communication between robots (Heselden & Das, 2023).

In terms of decision-making strategies, MRPP produces centralized, decentralized and distributed strategies(Abujabal et al., 2023). Centralized decision-making, where a single central control unit coordinates all the robots. In the decentralized approach, robots make decisions based on their controller and use their individual data. Distributed decision making is based on the ability of each robot to make decisions on its own in a distributed (individual) manner to enable robots to work collaboratively to solve a problem in the decision-making process. Without a central decision-making unit, robots make individual decisions to solve a problem and contribute to solving the overall problem (Keskin et al., 2024).

In a recent study, a new online MRPP method was developed and a learning-based artificial bee colony (ABCL) algorithm was presented. The proposed ABCL algorithm aims to improve search efficiency, and significant improvements over the original ABC algorithm have been reported (Cui et al., 2024). In another study, the authors presented a two-level method for finding obstacle-avoiding paths in the MRPP problem. At a lower level, they developed the SI-RRT* algorithm for a single robot. Considering the concept of safe time intervals, path planning was developed for both dynamic and static obstacle avoidance. At the upper level, SI-CPP, and SI-CCBS. SI-MRPP methods have been developed to avoid collisions and overlaps between robots. The proposed SI-RRT* was found to be probabilistically and asymptotically optimal compared with the other methods (Sim et al., 2024).

MRPP is considered an optimization problem, and a solution is sought to reach the goal through unobstructed paths. In a study, an adaptive multi-UAV (unmanned aerial vehicle) path planning method (AP-GWO) was developed using the grey wolf algorithm (GWO). The proposed method addresses the problems of a long convergence time and path deviation in multi-UAV mission deployment. AP-GWO introduces innovations, such as a spiral position method inspired by the whale algorithm, to adjust exploration and exploitation search features and adaptively adjust leadership features (Jiaqi et al., 2022). In this study, a MRPP that combines the advantages of the grey wolf algorithm (GWO) and the Teaching Learning based algorithm (TLBO) is proposed. The Grey Wolf Algorithm is a meta-heuristic optimization algorithm (Mirjalili et al., 2014). The algorithm provides an optimization solution technique based on the hierarchy of the wolves. TLBO is an optimization algorithm developed with reference to the teaching and learning processes of teachers and students. A computational process consisting of a Teaching phase and a Learning phase is used (R. V. Rao et al., 2011). By combining the GWO and TLBO algorithms, a more effective balance between exploration and exploitation can be achieved. In the MRPP problem, a solution that can avoid both robots and obstacles is sought using GWO and TLBO.

The contributions of this study are summarized as follows:

- A methodology has been developed for the MRPP that utilizes the advantageous aspects of the GWO and TLBO algorithms to develop a global best solution with fast closing towards the optimal solution and increasing the diversity.
- Effectiveness of GWO-TLBO algorithm for MRPP problem in different environments and with different number of robots has been investigated.
- The developed GWO-TLBO MRPP is compared with other GWO according to parameters, such as the fitness function and distance to the target.

2. Materials and methods

In the proposed methodology, an optimal solution is sought for the MRPP by utilizing the advantages of the GWO and TLBO algorithms. In this context, the GWO, TLBO, and GWO-TLBO combinations are explained in the methodology. It was analysed how the proposed GWO-TLBO method was adapted for both GWO and TLBO.

2.1. Grey Wolf Optimization

The grey wolf algorithm (GWO) is an optimization algorithm inspired by the hunting behavior of grey wolves (Mirjalili et al., 2014). Since grey wolves in the wild live in herds, the algorithm developed is based on herd intelligence. Grey wolves establish a hierarchy in their hunting behavior. This hierarchy consists of leader and lower classes. The hierarchical structure is inspired as follows. They are categorized as α wolf, β wolf, δ wolf and ω wolf respectively. At the top of this hierarchical structure, the α wolf is the leader of the herd, managing herd dynamics and making critical decisions. This is followed by the β wolf and others. At the next level, there are δ wolves that obey the wolves in the upper classes of the hierarchy and perform tasks such as surveillance and exploration. In the lowest class, ω wolves. They form the rest of the population and cooperate with other wolves. The main components of the GWO algorithm are the hierarchy of wolves and hunting behavior. Hunting behavior consists of searching, encircling and hunting. This behavior is expressed mathematically as a hunting mechanism. Initially, wolves are modelled mathematically as in equations (1) and (2) to search and encircle their prey(Dong et al., 2022).

$$D = |C.X_T(t) - X(t)| \qquad (1)$$

$$X(t+1) = X_T(t) - A.D \qquad (2)$$

Here, D is the distance between the wolf and prey. $X_T\ (t)$ is the target prey position at time t. $X(t)$ is the grey wolf position at time t. A and C are coefficients that affect the distance between the wolf and prey and the position of the target prey, respectively. A and C are described in Equations (3) and (4), respectively. A is a parameter that emerged as the convergence factor and decreases towards zero with each iteration in 2.

$$A = 2.a.r_1 - a \qquad (3)$$

$$C = 2.r_2 \qquad (4)$$

r_1 and r_2 are randomly generated numbers ranging from 0 to 1. Here also A and C denote a search behaviour that determines the exploration and exploitation behavior of the algorithm. When the coefficient A is greater than 1 at each iteration, the wolves perform their search behavior with a global exploration behavior. When A is less than 1, the wolves approach the target and an exploitation behavior is applied.

The behavior of wolves hunting prey is expressed in equations (5-7). For the wolves to move towards the target, α wolf, β wolf and δ wolf in the hierarchy play an important role in determining the position of the targeting robot with respect to a prey. Here, α wolf, β wolf and δ wolf are the best candidate solutions respectively. Briefly, when fitness values are ranked, they represent the best solutions. The hierarchy consists of the α wolf, β wolf and δ wolf. The position details of the α wolf, β wolf and δ wolf are given in equations (5) and (6). (Liu et al., 2023).

$$\begin{aligned} D_\alpha &= |C_1.X_\alpha - X| \\ D_\beta &= |C_2.X_\beta - X| \\ D_\delta &= |C_3.X_\delta - X| \end{aligned} \qquad (5)$$

$$\begin{aligned} X_1 &= X_\alpha - A_1.D_\alpha \\ X_2 &= X_\beta - A_2.D_\beta \\ X_3 &= X_\delta - A_3.D_\delta \end{aligned} \qquad (6)$$

Here C_1, C_2, C_3 and A_1, A_2, A_3 are random coefficients derived according to equations (3) and (4). The wolf in the search sequence determines the new position by averaging the positions of these three wolves. Equation (7) describes the new position determined.

$$X(t+1) = \frac{X_1+X_2+X_3}{3} \qquad (7)$$

The stages of the GWO algorithm are used to determine the exploration and exploitation behavior in the hunting process. While wolves search for a global solution while searching for a target, they focus on the local solution when they approach the target.

2.2. Teaching Learning Based Optimization

The TLBO algorithm is a population-based optimization algorithm inspired by teacher and student behavior. This method was introduced by Rao et al. (2010). They search for a solution in the population to obtain a global solution. TLBO consists of two stages. These are Teaching Phase and Learning Phase (R. V. Rao et al., 2011). At the stage of the teaching phase, the teacher uses experience to increase the average result of the class (R. V. Rao, 2016). In short, the teacher is what the teacher conveys. The class is defined as $\{X_1, X_2, \ldots, X_N\}$. N is the number of classes. It is the teacher who represents the best solution. It is denoted by $X_{Teacher}$. The average of the class is denoted by M. Equation (8) is expressed by M.

$$M_t = \frac{1}{N}\sum_{i=1}^{N} X_i \quad (8)$$

The updating of students' positions is determined by equation (9).

$$X_{new} = x_{old} + r_T.(X_{Teacher} - TF.M) \quad (9)$$

Here TF is the training factor. TF is a factor that determines how much information the teacher conveys. r_T is the training factor. $r_T$is a randomly generated number in the interval [0 1]. TF is calculated as expressed in equation (10).

$$TF = round[1 + rand(0,1)] \quad (10)$$

In the learning phase, students advance their knowledge through information received from the teacher or by interacting with other students. The way of individual and interactive learning varies. The fitness function values of X_i and X_j of two randomly selected students are compared as expressed in equations (11) and (12) (R. V. Rao et al., 2011).

$$X_{new} = X_{old} + r_L.(X_i - X_j) \quad f(X_i) < f(X_j) \quad (11)$$

$$X_{new} = X_{old} + r_L.(X_j - X_i) \quad otherwise \quad (12)$$

Here r_L is a random number between [0 and 1]. X_{old} contains the previous result here.

2.3. Advanced Grey Wolf Optimization - Teaching Learning Based Optimization

The developed method utilizes the advantageous aspects of GWO and TLBO algorithms. With GWO, new positions are determined hierarchically and with TLBO, the teaching and learning phase enables individuals to learn through experience transfer and interaction. The TLBO algorithm produces a more optimal fitness function value than the best solution α wolf in the GWO algorithm. The GWO algorithm has the advantage of being strong in the local solution as well as the target solution. The TLBO algorithm shows high performance in global optimum solutions. TLBO performs a balanced search and distribution and explores a large solution space, while GWO focuses on the target by shrinking the search space as it approaches the target. Figure 1 shows the proposed advanced GWO-TLBO method.

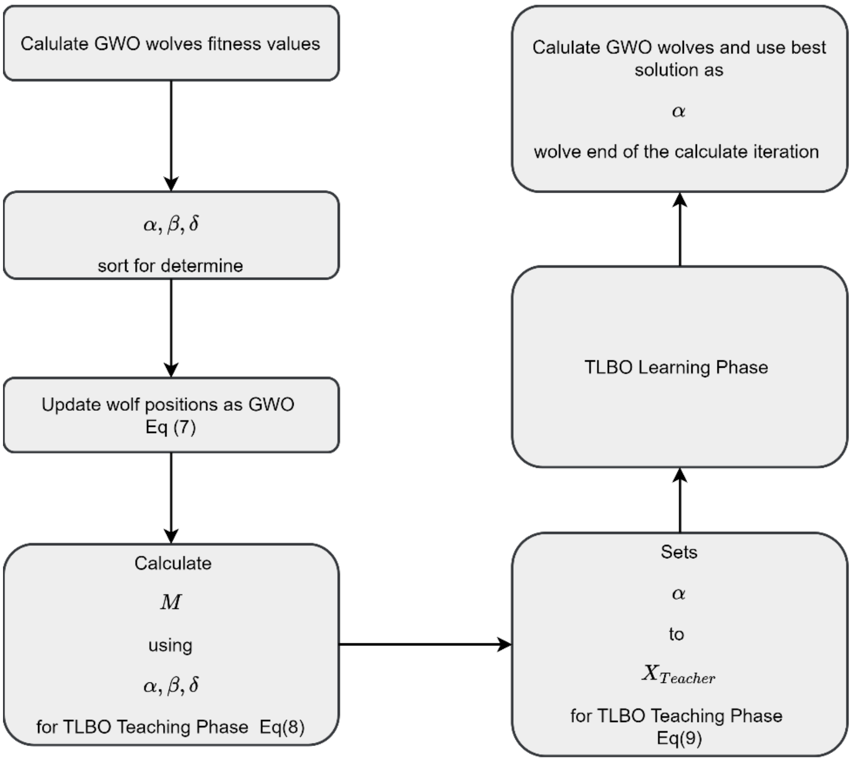


Figure 1. Developed GWO-TLBO method

The GWO-TLBO algorithm ranks the best fitness value of the population wolves as in GWO to find the positions of α wolf, β wolf and δ wolf. In Equation (7), the new position of GWO is determined by averaging the positions of these three wolves. It updates the positions according to GWO. For the learning phase of the TLBO algorithm, it uses the positions α, β and δ for the calculated student mean M. Then the TLBO learning phase is applied. In Equation (8), the average of the class represents the solution of M. At the end of the computation iteration, the α position of the GWO algorithm is used as the best solution. When the iteration cycle is completed, the position α is affected by both the GWO algorithm and the TLBO algorithm at the end of the process. The position is updated to reach the best solution. This update is described in equation (9).

3. MRPP Using Grey Wolf Optimization - Teaching Learning Based Optimization

The environment in which mobile robots move is full of static obstacles. Mobile robots make path planning in different positions to reach different obstacles. The path planning of each robot plans its path to avoid obstacles and each other in order to reach its target position. Robots are expressed as $\{R_1, R_2, R_3 \ldots R_n\}$. R_n n. refers to the robot's position pair. The environment in which the robots move is expressed as $E = \{E_x, E_y\}$. $\{E_x, E_y\}$ respectively evaluate the boundary of the motion environment. The obstacles in the environment are expressed as $O = \{O_1, O_2, O_3 \ldots O_n\}$. O_n is the position of the nth obstacle. In this study, the obstacles are defined as circles. Each obstacle has a radius. Since each robot has a starting and ending position, the starting position of the robot is expressed as S_{R_n} and the target position is expressed as F_{R_n} $[x, y]$ position. Here R_n denotes the nth robot. The path of each robot to the target is expressed as P_{R_n}.

Algorithm 1 describes the pseudo-code of the MRPP according to the proposed GWO-TLBO algorithm. The algorithm performs path planning of the robots based on a certain number of iterations. When any of the robots reaches the goal, the iteration cycle is terminated.

Algorithm 1: Pseudo code of GWO-TLBO MRPP

```
Determine Enviroment E = {E_x, E_y}   % Define Environment
Set Obtacles O = {O_1, O_2, O_3 ... O_n}   % Define Obstacles
Determine Robots { R_1, R_2, R_3 ... R_n}   % Define Robots
Set Robots Starts and Final Positions {(S_R1, F_R1), (S_R2, F_R2), ... (S_Rn, F_Rn)}   % Define Positions
Initilize  P_Rn   % Initialize a path structure for each robot
Determine Obj(.) Function  % Define fitness function
Set Initialize GWO(wolves) parameter
Generate population as GWO wolves
Set Initialize TLBO (Teaching(X_Teacher,M) Learning) parameter

%Execute the Path Planning for Each Robot
For 1 to  n do ← { R_1, R_2, R_3 ... R_n}
  For 1 to  iteration do
        Calculate Fitnes values as Obj(.) for GWO wolves population
        α , β , δ ← Sort wolves as fitness values and determine best first three wolves
        Uptade wolves position as GWO
        Determine the class mean M using the positions of the worms α, β, δ
        X_Teacher= α wolf  % In the TLBO algorithm, assign the best wolf (α) as the teacher.
        Run TLBO Teaching Phase
        Run TLBO Learning Phase
        Select α best solution
        Add  α to P_Rn %Add the best solution to the current robot's path plan.
        If the robot reaches the target,
          End the loop
```

3.1. Object function of Multi Robot Path Planning

The fitness function of the MRPP according to the GWO-TLBO algorithm proposed in Algorithm 1 is explained in this section. The fitness function is determined by calculating each robot's distance to the target (T_n) distance to the nearest obstacle (D_n) and collision penalty score C_n according to the distance to neighboring robots.

The fitness function is described in equation (13). The purpose of this equation is to create a single fitness value $Obj(n)$ that measures how successful and safe each robot is in path planning. The fitness function is determined by T_n, the distance to the nearest obstacle, the inverse proportion of D_n and the sum of C_n. The fitness function reaches a minimum as the distance to the target gets closer, the distance to the obstacle gets further away and the total penalty score decreases with respect to the distance to neighboring robots.

$$Obj(n) = T_n + \frac{1}{D_n+1} + C_n \qquad (13)$$

The distance of each robot to the target T_n is described in equation (14) and the distance to the nearest obstacle D_n is described in equation (15).

$$T_n = \left\| F_{R_n} - S_{R_n} \right\| \qquad (14)$$

$$D_n = \min(\|R_n - O\|) \qquad (15)$$

Equation (16) describes the total collision penalty score C_n according to the distance to neighboring robots.

$$C_n = \sum_{i=1}^{n} \sum_{j=1, j \neq 1}^{n} P_{ij} \qquad (16)$$

Equation (17) describes the calculation of the penalty score P_{ij}.

$$P_{ij} = P_S \qquad (17)$$

The P_S used here is called squared penalty. It is used to penalize the minimum approach distance. P_S is described in equation (18).

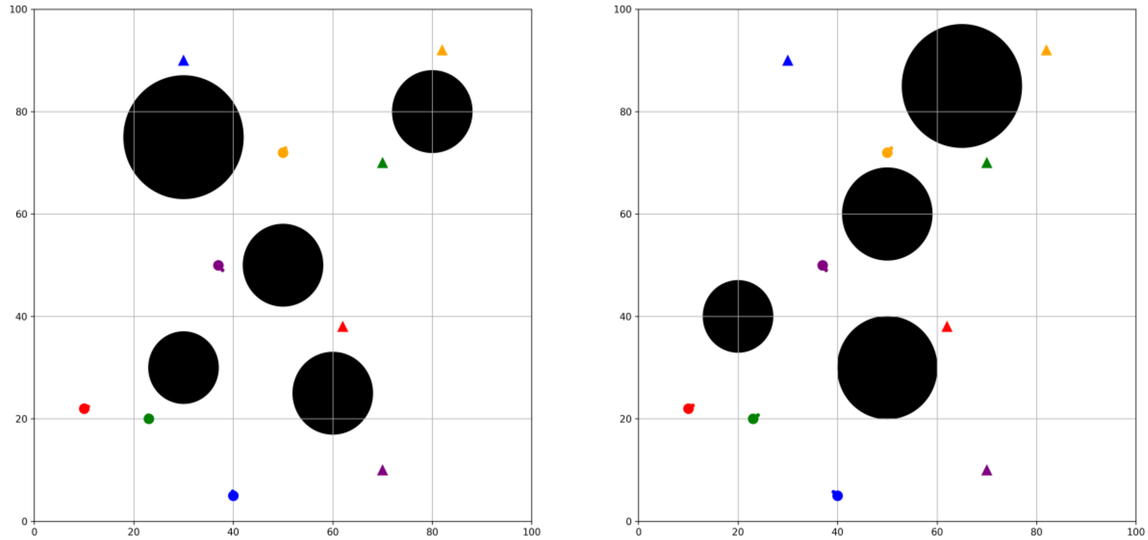
$$P_S = \begin{cases} (d_{min} - d_{ij})^2 & if\ d_{ij} < d_{min} \\ 0 & otherwise \end{cases} \qquad (18)$$

Selected i. Robot i. and j. The distance between robot i and robot j is expressed as d_{ij}. Equation (19) describes d_{ij}.

$$d_{ij} = \left\| R_i - R_j \right\| \qquad (19)$$

4. Results

In this study, the proposed GWO-TLBO method for the MRPP problem is validated through simulations. The effectiveness of the proposed algorithm is analysed for different numbers of robots in obstacle-filled environments. The number of robots, environment and obstacles are determined in a way that will challenge the robots to reach the target. It is also compared with other GWO based methods. Simulations are carried out in 2 different environments shown in Figure 2. Environment-1 in (a) and Environment-2 in (b) are circular obstacle environments placed at different locations. The size of each environment is 100x100 unit squared area.



(a) Enviroment-1 (b) Enviroment-2

Figure 2. Environments for MRPP

The developed GWO-TLBO method was simulated with 5 and 10 different robots in Environment-1 and Environment-2. In addition, for the number of wolves, the population parameter used in the GWO algorithm, 10 and 20 robot combinations were also included in the simulations. The proposed method is compared with both GWO (Mirjalili et al., 2014) and Improved GWO (GWO_IM) (Ou et al., 2023) algorithms. As comparison parameters, the sum of the fitness functions of all robots according to the methods, the sum of the distances to the target and the appropriate value of each robot according to the combination of the number of populations in each method are used. In the simulation experiments, 1500 iterations were carried out according to the number of robots, environment and population number combinations. The safety distance from the obstacles was determined as 3 units. In the simulation experiments, the iteration process of the robots reaching the target is terminated.

4.1. Fitness function results for each robot

In the simulations applied according to the number of robots, number of populations, and different environments, the results were discussed according to the fitness function value of each robot. In the applied simulations, the results were evaluated depending on the number of 10 and 20 GWO populations for 5 and 10 robots, respectively. Figure 3 shows the results for 5 robots and 10 populations. According to the obtained results, the robots performing path planning using the proposed GWO-TLBO method reached the goal safely. The fact that the fitness values of some of the other compared methods remain in a constant value range throughout the iteration means that the robot cannot find a solution to the challenging fitness function or the robot cannot reach the goal, which affects the integrated success of a robot team.

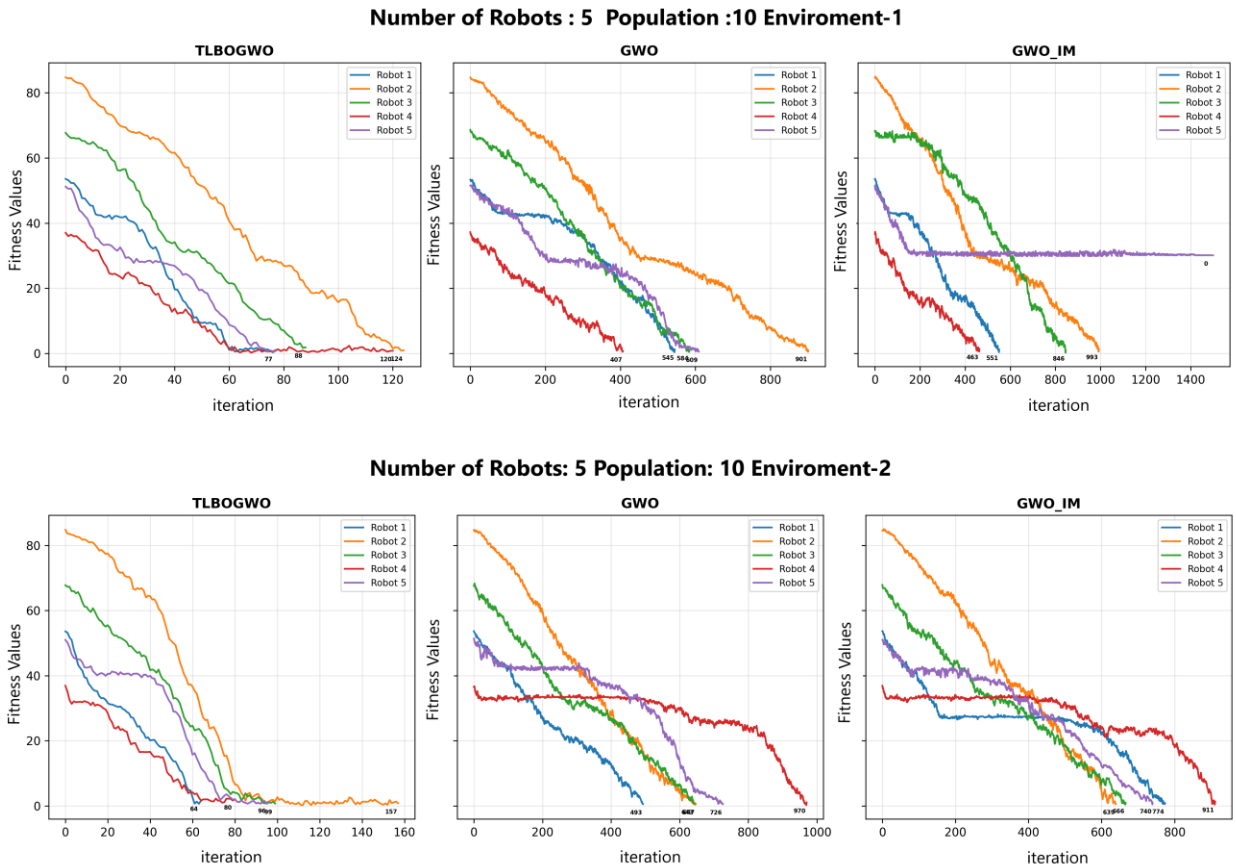


Figure 3. 5 robots, 10 populations Fitness value results for Environment-1 and Environment-2

Figure 4 shows the results for Environment-1 and Environment-2 with 5 robots and 20 populations. The increase in the number of populations accelerated the minimum value of the fitness function of the proposed GWO-TLBO method. It also closes to the minimum faster than other methods.

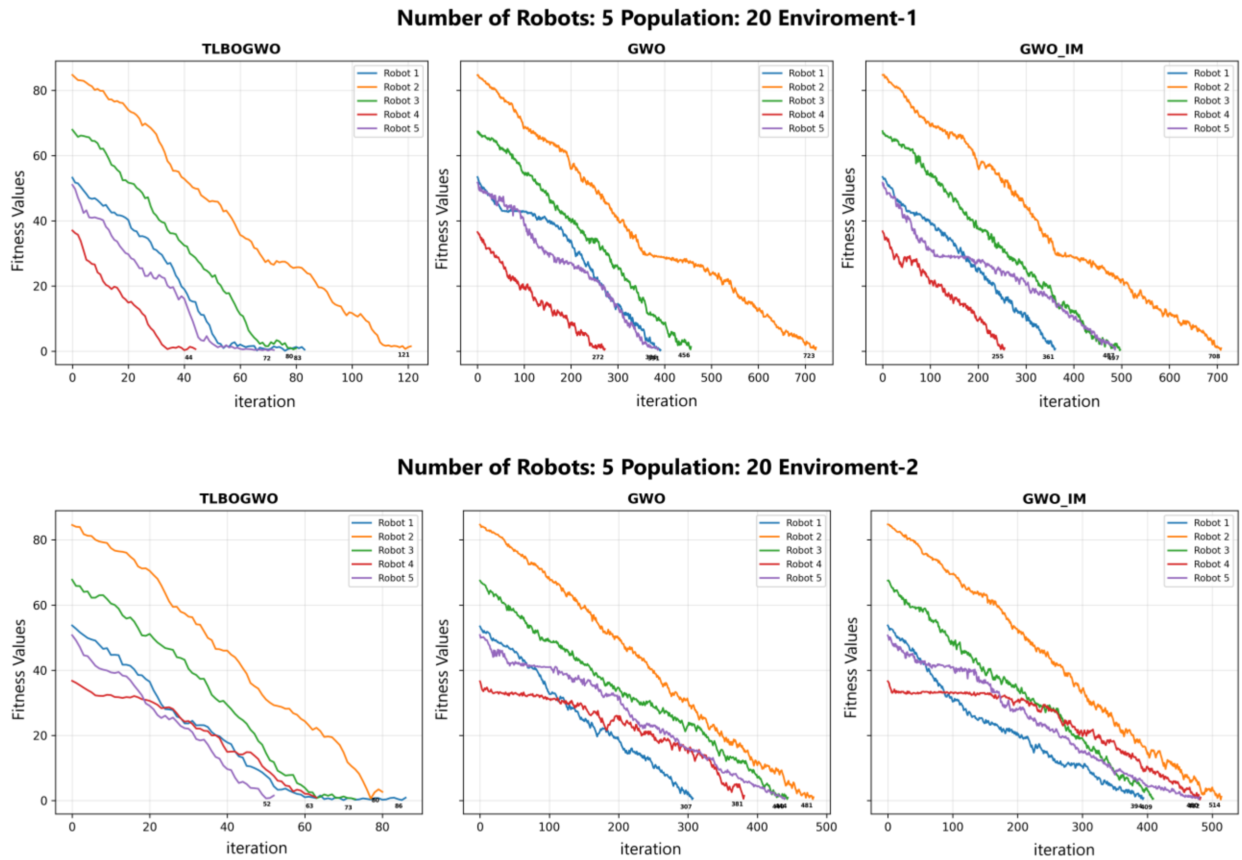


Figure 4. 5 robots, 20 populations Fitness value results for Environment-1 and Environment-2

Figure 5 shows the results for Environment-1 and Environment-2 with 10 robots and 10 populations. In order to evaluate the flexibility and scalability of the proposed method, a comparison is considered in terms of fitness values for increasing the number of robots in Environment-1 and Environment-2. The performance of the proposed method in the difficult fitness function is clearly successful in terms of closing speed.

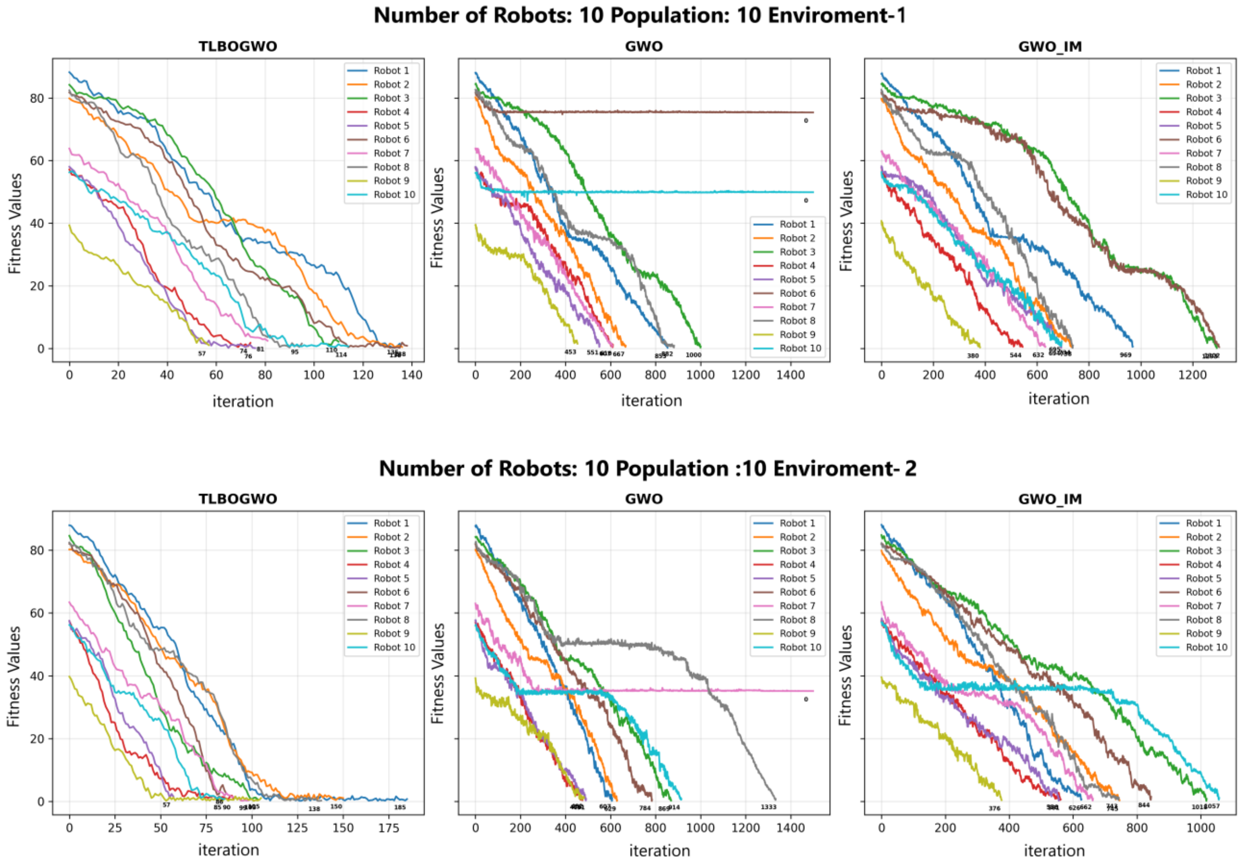


Figure 5. 10 robots, 10 populations Fitness value results for Environment-1 and Environment-2

Figure 6 shows the results for Environment-1 and Environment-2 with 10 robots and 20 populations. For 10 robots, the effectiveness of increasing the number of populations in terms of the appropriate function value in the MRPP problem is compared with other methods. The proposed method is radically successful in terms of each robot reaching the target compared to other methods. Some of the robots in the other compared methods did not reach the goal at the end of the number of iterations.

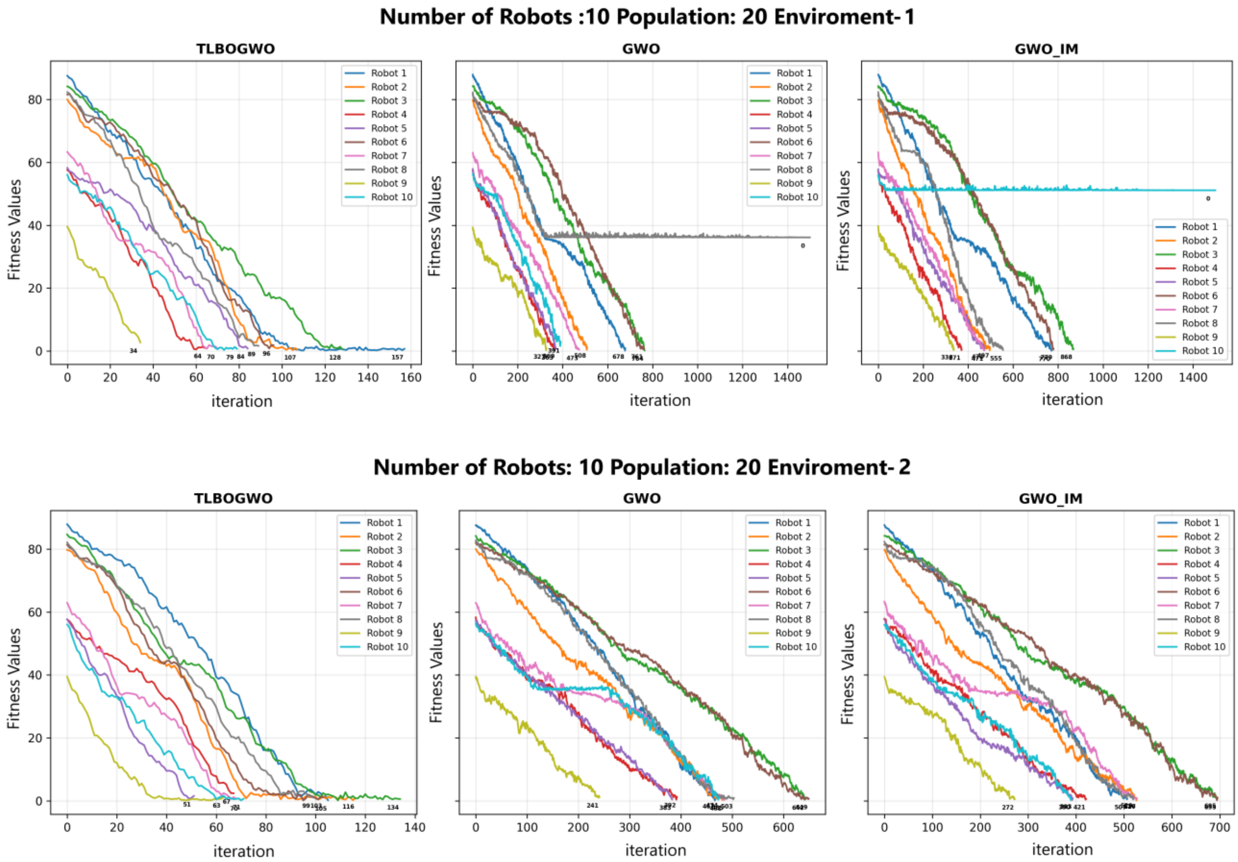


Figure 6. 10 robots, 20 populations Fitness value results for Environment-1 and Environment-2

4.2. Results According to Total Fitness Function Value

The proposed GWO-TLBO is evaluated in terms of the effectiveness of the whole robot team of the sum of the fitness value of each robot. The fitness function value calculated for each robot to reach the target is different in terms of initial and target positions. Therefore, if all robots reach the target, which is the expected behavior, the total fitness function value also changes. The total fitness function value shows the effectiveness of the path planning method with all robots reaching the target safely. Figure 7. shows the total fitness function value according to different robot numbers for Environment-1. The proposed GWO-TLBO method reaches the minimum faster in terms of total fitness value. This also shows the ability of the robot team to achieve its goals.

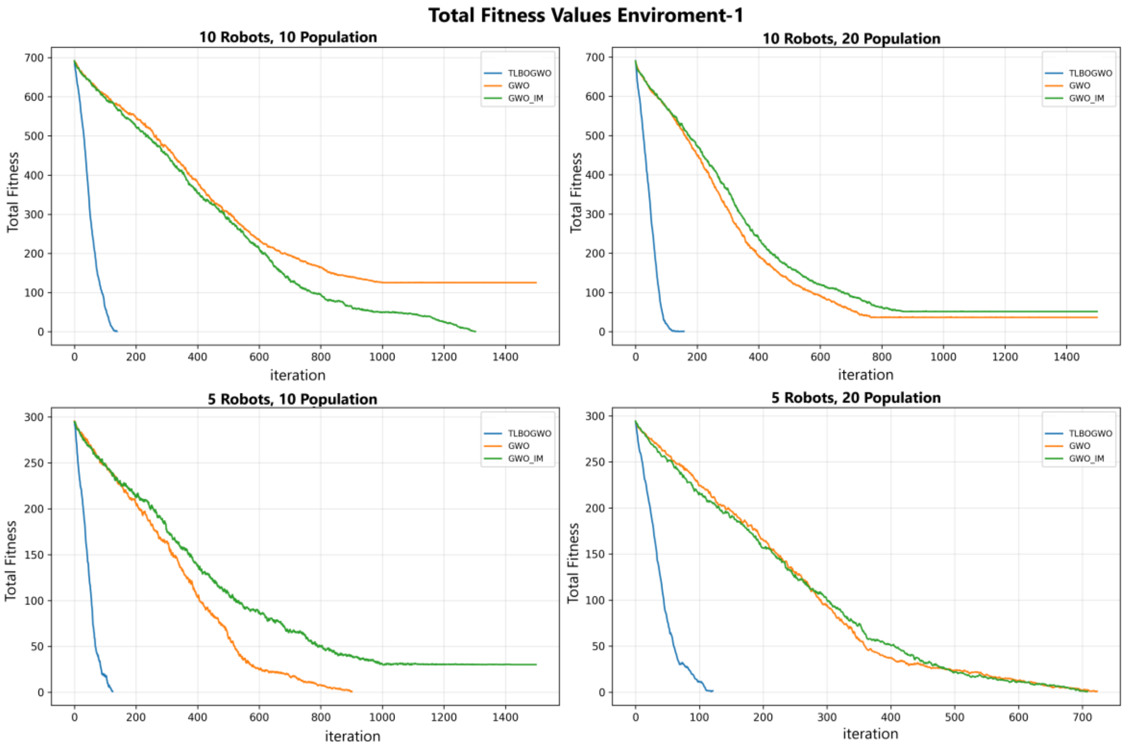


Figure 7. Total fitness values for Environment-1

Figure 8. The total fitness value according to different robot numbers for environment-2 is shown. The methods are evaluated in terms of the ability to reach the target in different environment conditions. According to the comparison results, the total fitness value of the proposed method is faster and more optimal than the other methods as in Environment-1. This is also an indication of the adaptation of the proposed algorithm to the environment.

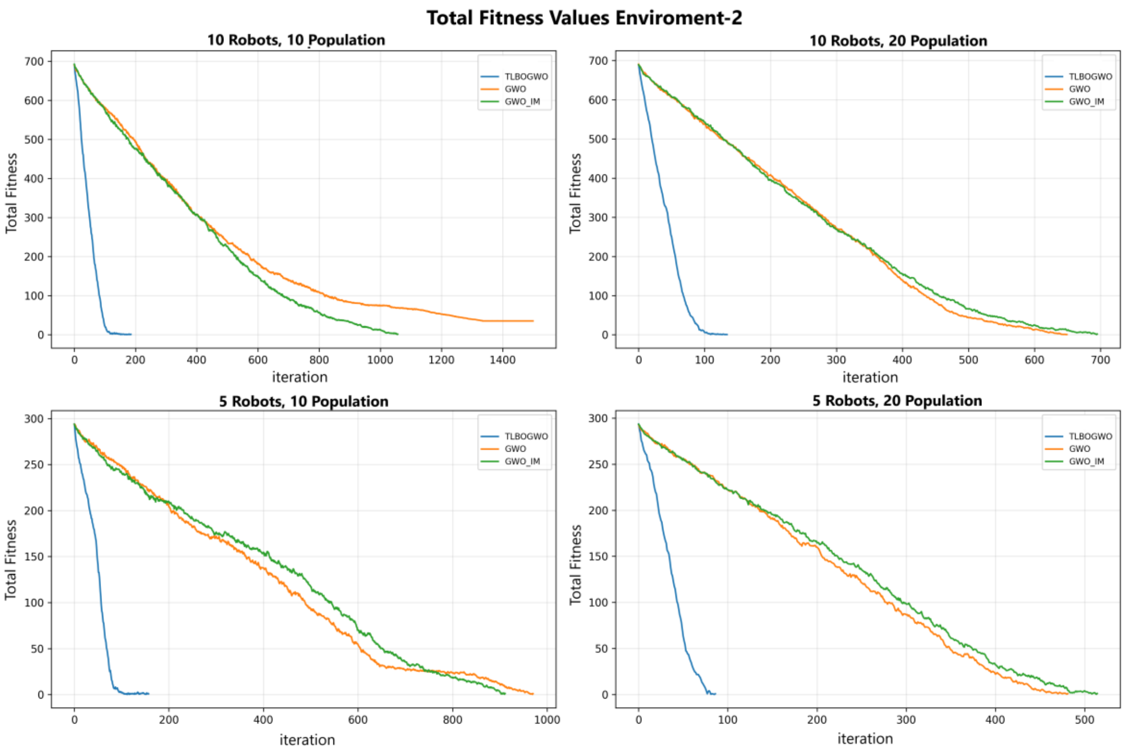


Figure 8. Total fitness values for Environment-2

GWO-TLBO provides a faster solution in the initial iterations, whereas the traditional GWO_IM provides more stable results in some cases.

4.3. Results Based on Total Distance to The Target

In MRPP, the sum of the distance of each robot to the target is one of the criterion parameters of the MRPP method. This parameter also shows the efficiency of all robots to reach the target without colliding with each other and obstacles. The proposed method is considered to evaluate and improve the overall performance of the MRPP by optimizing the sum of the distances of all robots to the target. Figure 9 shows the sum of the optimal values of all robots for different robot and population numbers for Environment-1. The proposed GWO-TLBO generally reduces the total distance to the target in a shorter time. It approaches the target faster than other methods, especially in the initial times. This is an indication that GWO-TLBO provides a more effective solution in the initial iterations.

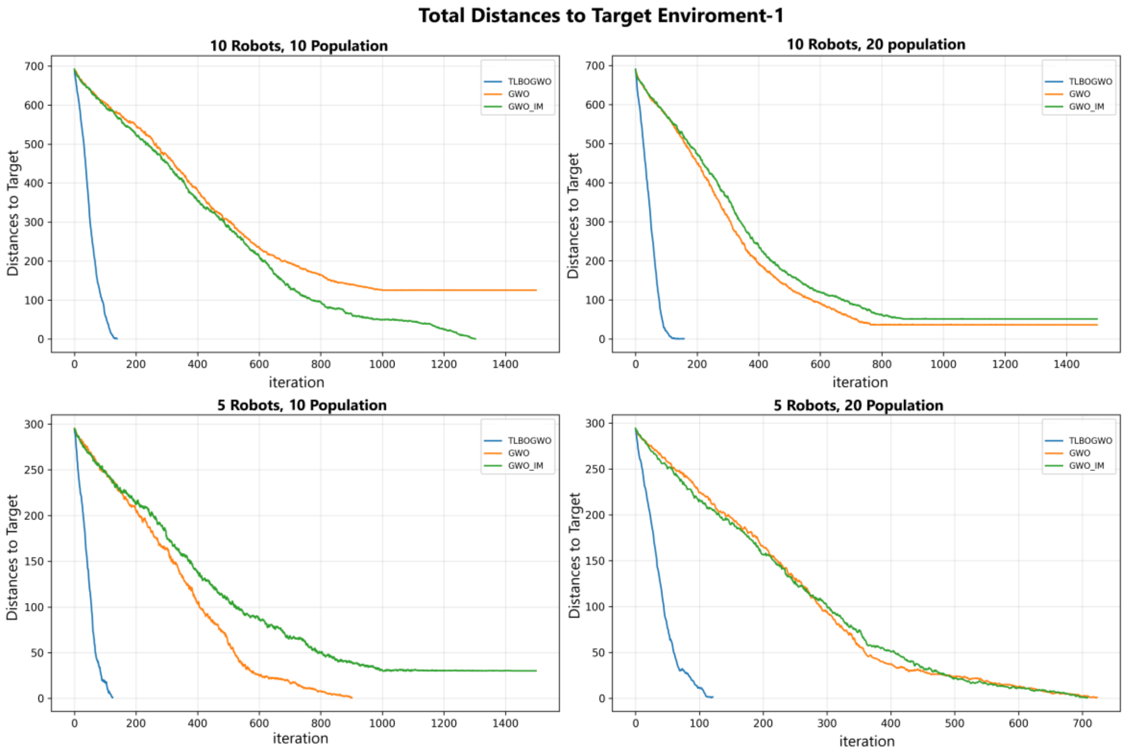


Figure 9. Total distances of all robots to the target for different numbers of robots and populations for Environment-1

Figure 10 shows the sum of the fitness values of all robots for different robot and population numbers for Environment-2. GWO_IM shows better closing to the target than GWO in some cases, but TLBOGWO performs better than the other methods.

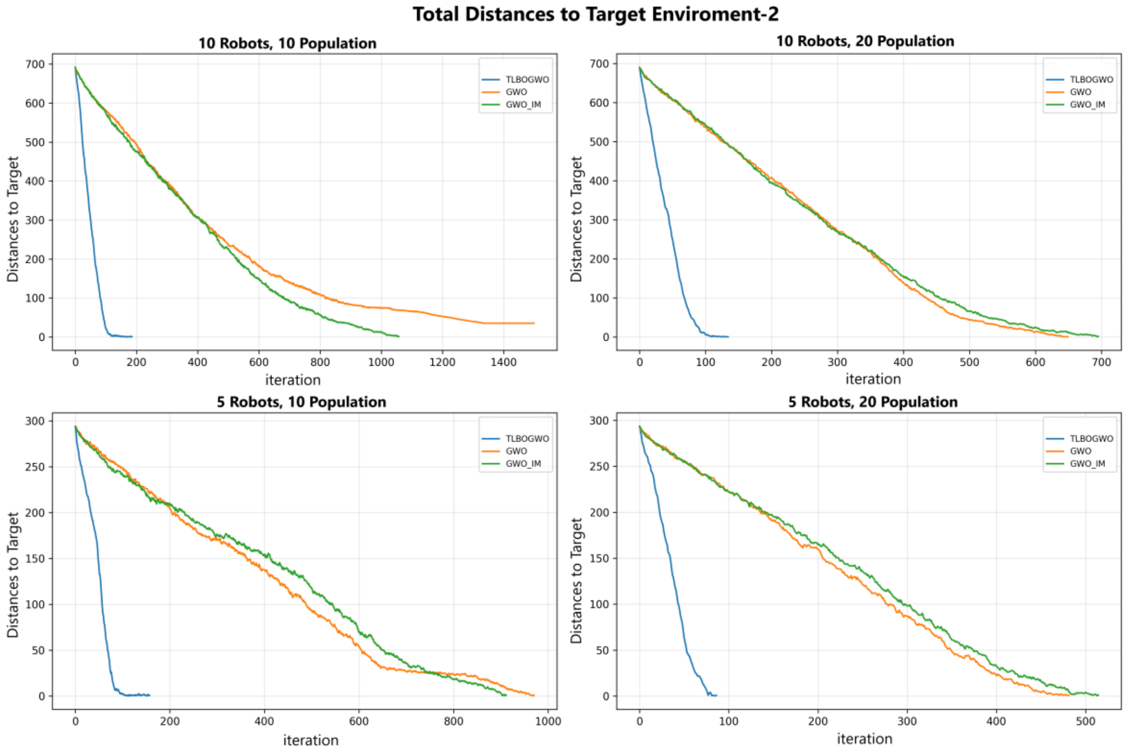


Figure 10. Total distances of all robots to the target for different numbers of robots and populations for Environment-2

When the total distances of all robots to the target are evaluated, TLBO-GWO showed a very fast convergence compared to the other algorithms compared in environment-1 and reached the target in 200 iterations. GWO_IM and GWO showed a slower convergence in total distance to the target. In the simulations performed in Environment-2, TLBO-GWO shows the fastest close, similar to Environment-1. In Environment-2, the closing performance of GWO_IM to the target is improved more significantly than that of GWO.

4.4. Path Planning Results According to Different Environments

The path planning of GWO-TLBO together with the other compared methods is analyzed in this section. In the compared methods, the flattening of the path for the MRPP is ignored. Figure 11 shows the path planning obtained from simulations with different population values, 5 and 10 robots for Environment -1. Since the proposed GWO-TLBO method updates the positions of each population member of the GWO algorithm to focus them on the target, the points close to the target are considered as the best result. The TLBO algorithm trains the population members for more optimal paths by referring to the alpha(α) wolf with a teacher-student structure. Thus, as seen in the path plans, depending on the fitness function of each robot in the proposed method, the robots determine the path plan that leads to the target in cooperation without colliding with both obstacles and each other in their current positions.

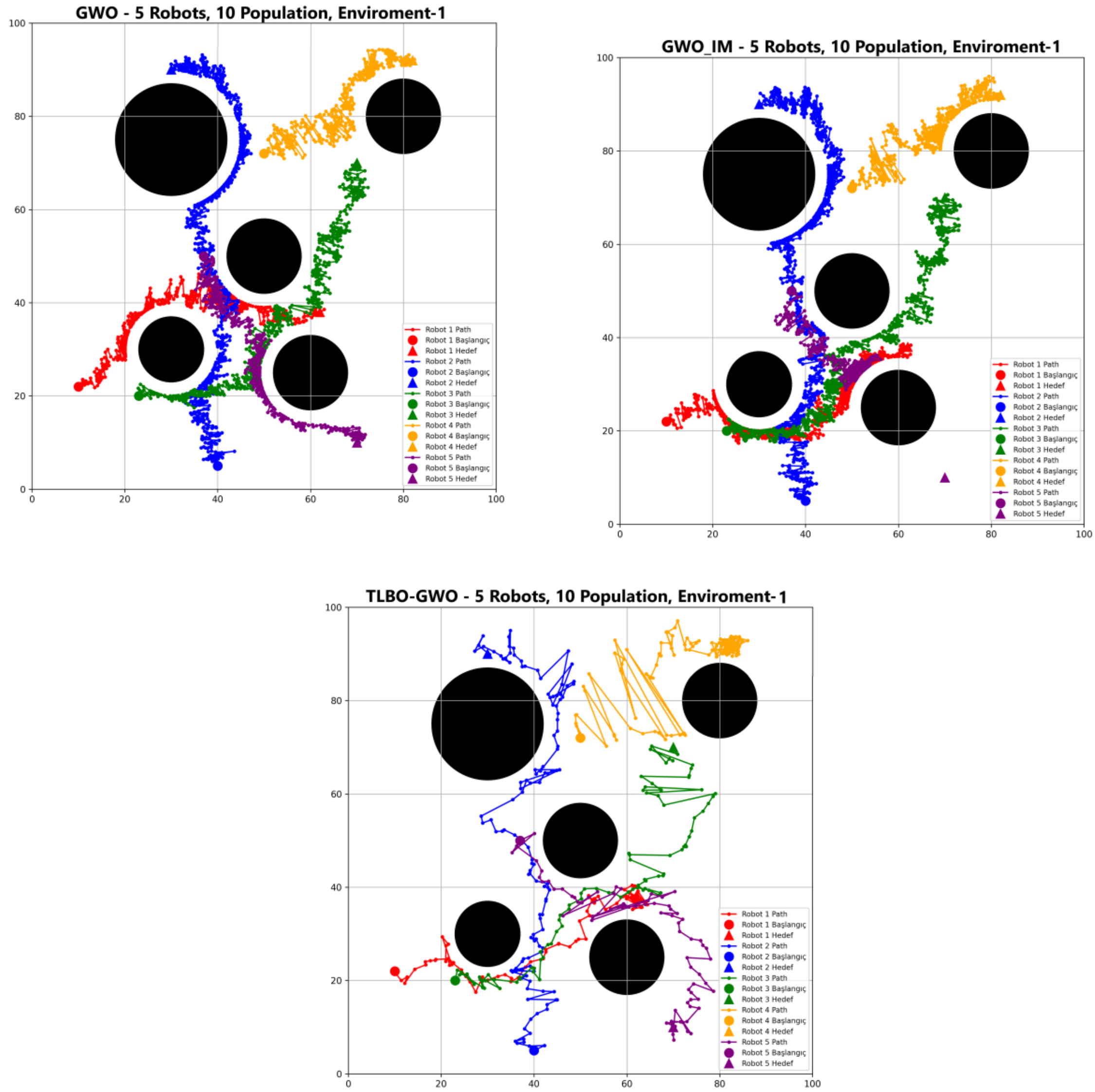


Figure 11. Simulations for 5 robots and 10 populations in Environment -1

Figure 12 shows the path planning according to the simulation results in Environment-2 with 10 robots and 20 population values. In the proposed method, each of the robots managed to avoid obstacles and reach their targets without colliding with each other. Depending on the increasing number of populations, the proposed method shows success in its ability to cope with scalable and complex environments. It is seen that the proposed method generates optimized paths for the robot team according to the fitness function, avoiding obstacles in a cooperative manner.

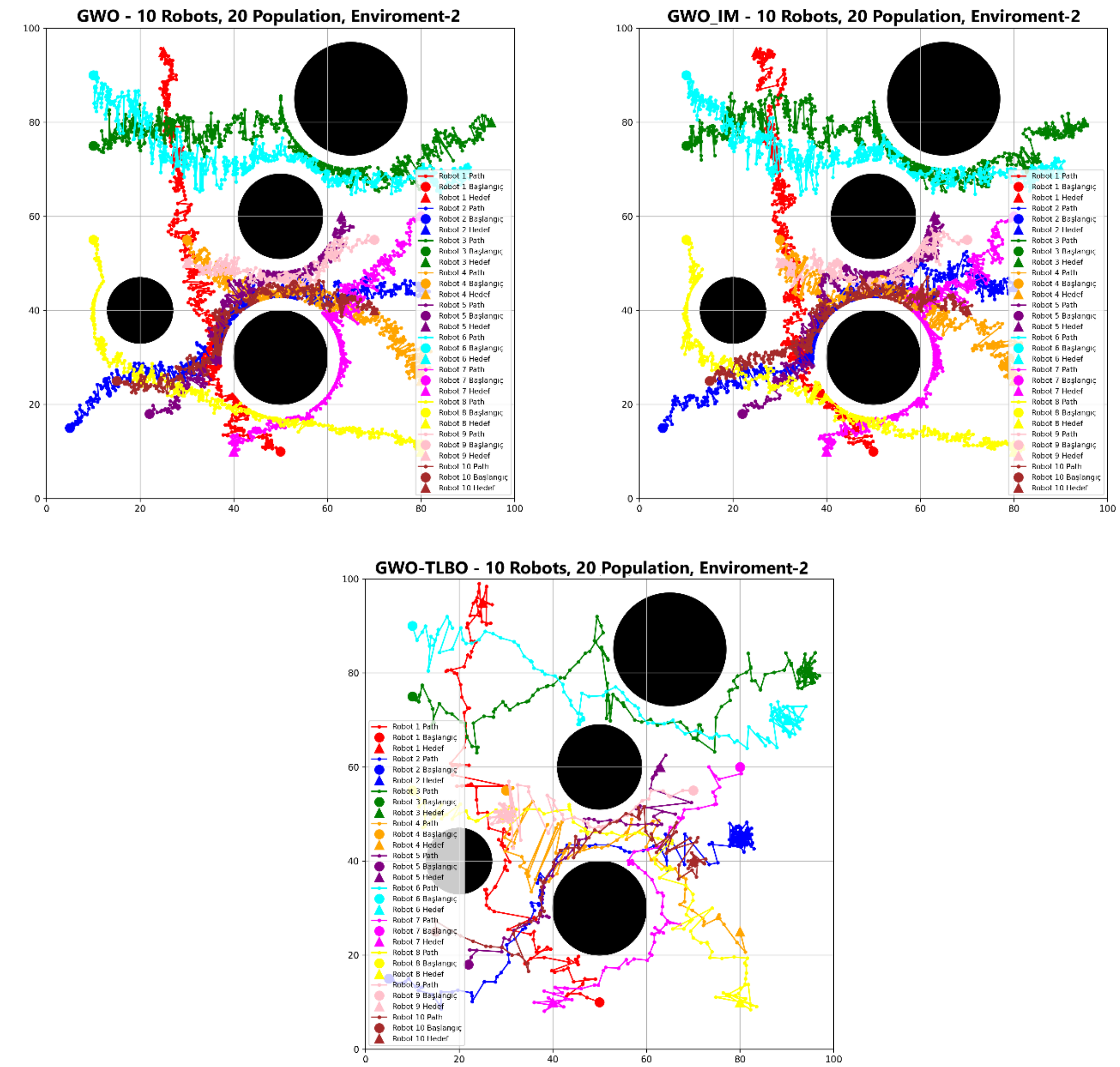


Figure 12. Simulations for 10 robots and 20 populations in Environment -2

5. Conclusion

In this study, a method for MRPP was developed using the advantages of the GWO and TLBO algorithms, and simulations were applied to demonstrate the effectiveness of the method. In the developed GWO-TLBO method, GWO stands out with its ability to avoid local minima, search in a wider area in the solution space, and focus on the solution with dynamic narrowing. Simultaneously, it diversifies the solution in the search space by preserving diversity in the search space. TLBO is effective for finding global optimum solutions. Simultaneously, it balances exploration and exploitation. By utilizing these advantages, the proposed approach increases search diversity and establishes a balanced solution search. In the GWO algorithm, the population member that obtains the best solution is assigned as a teacher in the TLBO. At the same time, the TLBO average position update mechanism transfers experience according to the average of the wolf with the three best GWO solutions. Thus, the advantages of focusing on both the solution space and the target are utilized. Through simulations, the effectiveness of the proposed algorithm is evaluated for different numbers of robots, different obstacle environments and different population diversity. In addition, the effectiveness of GWO-TLBO is compared with GWO-based methods. In order to evaluate the performance of the proposed algorithm, the fitness function value, the total fitness function value of all robots, the sum of the distances of all robots to the target are evaluated. According to the results obtained, the proposed algorithm has shown higher performance in terms of the compared parameters compared to other methods. In future studies, it is planned to focus on the adaptation of distributed algorithms for the MRPP task of GWO.

Acknowledgment and/or disclaimers, if any

The study did not receive any support. There is no institution or person to thank.

Conflict of Interest Statement, if any

There is no conflict of interest with any institution or person within the scope of the study.

References

Abujabal, N., Fareh, R., Sinan, S., Baziyad, M., & Bettayeb, M. (2023). A comprehensive review of the latest path planning developments for multi-robot formation systems. *Robotica*, *41*(7), 2079–2104. https://doi.org/10.1017/S0263574723000322

Apuroop, K. G. S., Le, A. V., Elara, M. R., & Sheu, B. J. (2021). Reinforcement Learning-Based Complete Area Coverage Path Planning for a Modified hTrihex Robot. *Sensors 2021, Vol. 21, Page 1067*, *21*(4), 1067. https://doi.org/10.3390/S21041067

Cao, Y., Long, T., Sun, J., Wang, Z., & Xu, G. (2023). Comparison of Distributed Task Allocation Algorithms Considering Non-ideal Communication Factors for Multi-UAV Collaborative Visit Missions. *IEEE Robotics and Automation Letters*. https://doi.org/10.1109/LRA.2023.3295999

Chakraa, H., Guérin, F., Leclercq, E., & Lefebvre, D. (2023). Optimization techniques for Multi-Robot Task Allocation problems: Review on the state-of-the-art. *Robotics and Autonomous Systems*, *168*, 104492. https://doi.org/10.1016/J.ROBOT.2023.104492

Cui, Y., Hu, W., & Rahmani, A. (2024). Multi-robot path planning using learning-based Artificial Bee Colony algorithm. *Engineering Applications of Artificial Intelligence*, *129*, 107579. https://doi.org/10.1016/J.ENGAPPAI.2023.107579

Dong, L., Yuan, X., Yan, B., Song, Y., Xu, Q., & Yang, X. (2022). An Improved Grey Wolf Optimization with Multi-Strategy Ensemble for Robot Path Planning. *Sensors 2022, Vol. 22, Page 6843*, *22*(18), 6843. https://doi.org/10.3390/S22186843

Heselden, J. R., & Das, G. P. (2023). Heuristics and Rescheduling in Prioritised Multi-Robot Path Planning: A Literature Review. *Machines 2023, Vol. 11, Page 1033*, *11*(11), 1033. https://doi.org/10.3390/MACHINES11111033

Jiaqi, S., Li, T., Hongtao, Z., Xiaofeng, L., & Tianying, X. (2022). Adaptive multi-UAV path planning method based on improved gray wolf algorithm. *Computers and Electrical Engineering*, *104*, 108377. https://doi.org/10.1016/J.COMPELECENG.2022.108377

Keskin, M. O., Cantürk, F., Eran, C., & Aydoğan, R. (2024). Decentralized multi-agent path finding framework and strategies based on automated negotiation. *Autonomous Agents and Multi-Agent Systems*, *38*(1), 1–30. https://doi.org/10.1007/S10458-024-09639-8/TABLES/4

Kumar, S., & Sikander, A. (2024). A novel hybrid framework for single and multi-robot path planning in a complex industrial environment. *Journal of Intelligent Manufacturing*, *35*(2), 587–612. https://doi.org/10.1007/S10845-022-02056-2/FIGURES/23

Li, J., & Yang, F. (2020). Task assignment strategy for multi-robot based on improved Grey Wolf Optimizer. *Journal of Ambient Intelligence and Humanized Computing*, *11*(12), 6319–6335. https://doi.org/10.1007/S12652-020-02224-3/TABLES/2

Lin, S. ;, Liu, A. ;, Wang, J. ;, Kong, X., Lin, S., Liu, A., Wang, J., & Kong, X. (2022). A Review of Path-Planning Approaches for Multiple Mobile Robots. *Machines 2022, Vol. 10, Page 773*, *10*(9), 773. https://doi.org/10.3390/MACHINES10090773

Liu, L., Li, L., Nian, H., Lu, Y., Zhao, H., & Chen, Y. (2023). Enhanced Grey Wolf Optimization

Algorithm for Mobile Robot Path Planning. *Electronics 2023, Vol. 12, Page 4026*, *12*(19), 4026. https://doi.org/10.3390/ELECTRONICS12194026

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software*, *69*, 46–61. https://doi.org/10.1016/J.ADVENGSOFT.2013.12.007

Mittal, H., Pandey, A. C., Saraswat, M., Kumar, S., Pal, R., & Modwel, G. (2022). A comprehensive survey of image segmentation: clustering methods, performance parameters, and benchmark datasets. *Multimedia Tools and Applications*, *81*(24), 35001–35026. https://doi.org/10.1007/S11042-021-10594-9/TABLES/6

Nedjah, N., & Junior, L. S. (2019). Review of methodologies and tasks in swarm robotics towards standardization. *Swarm and Evolutionary Computation*, *50*, 100565. https://doi.org/10.1016/j.swevo.2019.100565

Ou, Y., Yin, P., & Mo, L. (2023). An Improved Grey Wolf Optimizer and Its Application in Robot Path Planning. *Biomimetics 2023, Vol. 8, Page 84*, *8*(1), 84. https://doi.org/10.3390/BIOMIMETICS8010084

Qin, H., Shao, S., Wang, T., Yu, X., Jiang, Y., & Cao, Z. (2023). Review of Autonomous Path Planning Algorithms for Mobile Robots. *Drones 2023, Vol. 7, Page 211*, *7*(3), 211. https://doi.org/10.3390/DRONES7030211

Rao, R. V. (2016). Teaching-Learning-Based Optimization Algorithm. *Teaching Learning Based Optimization Algorithm*, 9–39. https://doi.org/10.1007/978-3-319-22732-0_2

Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, *43*(3), 303–315. https://doi.org/10.1016/J.CAD.2010.12.015

Shoeib, M. A., Lewandowski, J., & Omara, A. M. (2024). A novel methodology for vision-based path planning and obstacle avoidance in mobile robot applications. *Advanced Robotics*, *38*(12), 802–817. https://doi.org/10.1080/01691864.2024.2315591

Sim, J., Kim, J., & Nam, C. (2024). Safe Interval RRT* for Scalable Multi-Robot Path Planning in Continuous Space. *CoRR*. https://doi.org/10.48550/ARXIV.2404.01752

Tan, C. S., Mohd-Mokhtar, R., & Arshad, M. R. (2021). A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms. *IEEE Access*, *9*, 119310–119342. https://doi.org/10.1109/ACCESS.2021.3108177

Zhu, K., & Zhang, T. (2021). Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Science and Technology*, *26*(5), 674–691. https://doi.org/10.26599/TST.2021.9010012