Path planning for robots using Grey wolf Optimization

Meet Joshi - 202201065, Aryan Solanki - 202201239, Dev Davda - 202201242, Harshil Parmar - 202201371 and Dev Soni - 202201418

Abstract

Robot Path planning is a crucial aspect of mobile robot operations. Robotic systems employ intelligent algorithms to plan the robot's route from one point to another. The primary objective of path planning is to determine the feasible movements of a robot within an environment filled with obstacles, ensuring a collision-free path from the start to the target position. In this study, the Gray Wolf Optimization (GWO) algorithm was applied to solve the robot path planning problem. The GWO algorithm mimics the hunting behavior and social hierarchy of gray wolves in nature. Considering other researches on the same we have used GWO and modified GWO with an emphasis on the positioning of the alpha wolf(s). If a solution point lies within an obstacle zone, a violation is added to the cost function. The performance of the GWO algorithm was compared with other meta-heuristic algorithms for solving the robot path planning problem. The results demonstrate that the GWO algorithm successfully finds the optimal path for the test map used.

Key words: Robot Path Planning, Grey wolf optimization(GWO), Meta-heuristic algorithm

Introduction

Over the past three decades, meta-heuristic algorithms have gained significant popularity for addressing optimization problems. These algorithms are inspired by principles from evolution, physics, or social behaviours observed in natural groups like swarms or animal flocks. Meta-heuristic algorithms are typically categorized into types such as physics-based algorithms, evolutionary algorithms, swarm intelligence algorithms, bio-inspired algorithms, and other nature-inspired approaches. In physics-based algorithms, the optimization process starts with a single solution that is iteratively refined using physical equations. Evolutionary algorithms, such as the Genetic Algorithm (GA) [3][4] and Differential Evolution (DE) algorithm [9]–[11], are well-known in the meta-heuristic domain. Swarm intelligence algorithms encompass techniques like Particle Swarm Optimization (PSO). The Gray Wolf Optimization (GWO) algorithm, introduced by Mirjalili in 2014, emulates the hunting tactics and social hierarchy of gray wolves [39]. Robot path planning, particularly for mobile robots, has garnered significant interest in recent decades, with many methods proposed for navigating areas with fixed obstacles. The main objective in this challenge is to derive collision-free paths, enabling the robot to reach its target as quickly and with the shortest possible distance from the starting point [46]–[48]. The path planning problem involves determining a route from the robot's initial position to the target, considering the layout, obstacle locations, and boundary constraints. In this study, the GWO algorithm is applied to determine the optimal path from start to target without colliding with obstacles. To assess the performance of GWO, we used a test area featuring three circular obstacles of varying radii. The GWO algorithm's performance was then compared with that of established meta-heuristic algorithms, including Differential Evolution, Particle Swarm Optimization (PSO).

Gray Wolf optimizer

The Gray Wolf Optimizer (GWO) is inspired by the hunting strategies and social hierarchy of grey wolves. In their hierarchy, grey wolves are classified into four groups: alpha, beta, delta, and omega. The alpha wolf is the leader or dominant wolf and guides the pack, making it the most influential in managing the group. The beta wolf ranks second in the hierarchy and assists the alpha with various tasks. The delta wolf submits to both the alpha and beta wolves but exercises authority over the omega wolves. Within this social structure, the group includes scouts, guards, elders, hunters, and caretakers, with the omega wolf positioned at the lowest rank. Figure 1 represents the social structure of grey wolves.

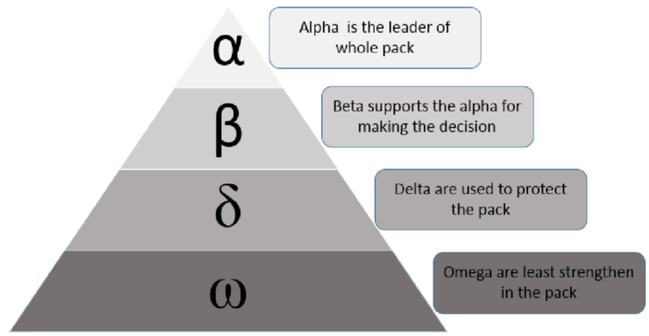

Fig. 1: social hierarchy of grey wolves

The group hunting strategy is another fascinating social behavior of gray wolves. In this approach, gray wolves first locate their prey and, under the alpha wolf's leadership, encircle it. In the mathematical model of this hunting strategy, it is assumed that the alpha, beta, and delta wolves possess the most accurate knowledge of the prey's location. Consequently, in the GWO algorithm, the positions of the wolves are updated using the top three solutions (alpha, beta, and delta). Notably, omega wolves are not included in the GWO code [39]. The mathematical model of the gray wolves' hunting mechanism is outlined below. A single iteration of the GWO algorithm includes four core steps: hierarchy division, searching, encircling, and attacking. Initially, the wolf population is categorized hierarchically based on each member's fitness, labeled as α, β, δ, and ω. The algorithm's optimization mainly depends on the top three best solutions from each generation. Mathematical models describing the grey wolves' search and tracking behaviors are as follows:

Searching: Grey wolves explore the search area for possible solutions by utilizing a mathematical model that mimics their hunting behavior. Encircling: When a potential solution or prey is located, grey wolves cooperate to surround it, creating an encircling formation to increase the chances of capture. Attacking: After forming the encirclement, grey wolves select the most suitable individual within the circle to launch the attack, aiming to enhance the population's overall fitness.

As shown in Figure 2, the solid circles represent the positions of α, β, δ, and ω wolves on a two-dimensional plane, while P marks the prey's relative position. The three essential phases of grey wolf hunting—approach, surround, and attack—are initiated once the prey's location is determined, with the alpha wolf (α) leading the pursuit, supported by β and δ. The positions of α, β, and δ, closest to the prey, guide the direction towards P. By calculating each wolf's fitness, the optimal, good, and suboptimal solutions are identified, and the other wolves' positions are set accordingly. In this context, each grey wolf represents a possible solution, with the α wolf's position being the optimal solution and the positions of β and δ reflecting good and suboptimal solutions.

The mathematical model that defines the search and tracking behaviors of grey wolves during prey pursuit is as follows:

$$\vec{D} = \left|\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)\right| \tag{1}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \tag{2}$$

where $\vec{D}$ represents the distance vector between the grey wolf and the prey, $\vec{X}p$ denotes the position vector of the prey, $\vec{X}$ represents the position vector of the grey wolf, t stands for the iteration number, and $\vec{A}$ and $\vec{C}$ are coefficient vectors.

Coefficient vectors $\vec{A}$ and $\vec{C}$ can be expressed as follows:

$$\vec{A} = 2a \cdot \vec{r}_1 - a \tag{3}$$

$$\vec{C} = 2 \cdot \vec{r}_2 \tag{4}$$

In the equation, the convergence factor is represented by $\vec{a}$, which decreases linearly from 2 to 0 with each iteration. The magnitudes of $\vec{r_1}$ and $\vec{r_2}$ are randomly generated numbers between 0 and 1. When grey wolves encounter their prey during the search process, under the guidance of α, β, and δ will encircle the prey. However, since the exact location of the optimal prey is unknown in the abstract space, to simulate the authentic behavior of grey wolves, the top three best solutions discovered thus far are preserved to determine the positions and stimulate other individuals to update their search positions [21–23]. The mathematical model corresponding to this approach is defined as follows:

$$\begin{cases} \vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}\right| \\ \vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}\right| \\ \vec{D}_\delta = \left|\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}\right| \\ \vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha \\ \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta \\ \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot \vec{D}_\delta \end{cases} \tag{5}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{6}$$

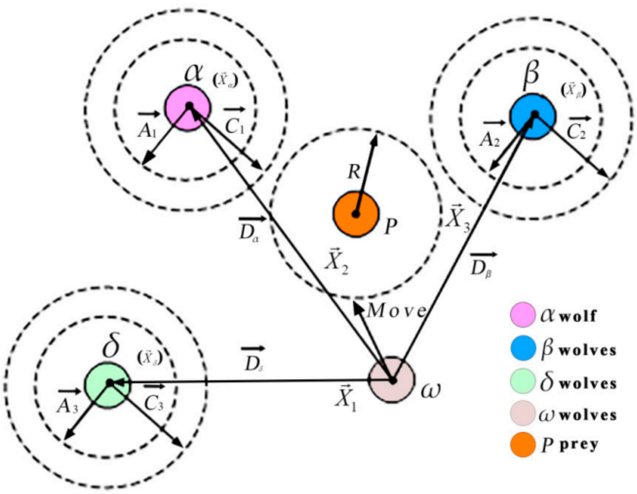

Fig. 2: GWO algorithm principle

Given below the pseudocode whows the functioning of Grew wolf optimization:

Initialize the positions of gray wolves
Calculate the cost values of gray wolves
Save the best gray wolf as alpha wolf
Save the second best gray wolf as beta wolf
Save the third best gray wolf as delta wolf
while (iteration ¡ maximum iteration)
Decrease using Eq. (14)
for each gray wolf
Generate the coefficient vectors for alpha, beta, delta
Calculate the distance vectors using Eqs. (1-3)
Calculate the trial vectors using Eqs. (4-6)
Update the position of gray wolf using Eq. (7)
end for
Calculate the cost values of updated gray wolves
for each gray wolf
if (gray wolf ¡ alpha wolf)
update alpha wolf
else if (gray wolf ¡ beta wolf)
update beta wolf
else if (gray wolf ¡ delta wolf)
update delta wolf
end if
end for
Update the elite antlion
increase iteration one
end while
return alpha wolf

Robot path planning

The robot path planning problem is a NP-hard optimization problem and this problem is often solved by meta-heuristic algorithms in the literature. The main aim in solving this problem is that the mobile robot should reach from the start point to the target position in the shortest path without touching any obstacles. It consists of the start and target positions, the size of obstacles, the shape of obstacles, the number of obstacles, the zone's boundaries. The objective function of path planning problem is given below:

$$J = \min_{x,y} Q(1 + \beta V) \tag{7}$$

where β is violation coefficient (100), V indicates the violation cost, Q denotes the total distance between start and target points. In calculating the violation for the candidate solution, the following pseudo code was used.

Algorithm 2: Pseudo code of violation's calculation.

Violation ← 0
for each obstacle

Calculate distance vector between the obstacle's center and path

$a \leftarrow \max\left(1 - \frac{\text{distance}}{\text{radius}_{\text{obs}}}, 0\right)$

Violation $\leftarrow$ Violation $+\text{mean}(a)$

end for

Our Approach

Aim

The objective of this experiment is to assess the effectiveness of the Grey Wolf Optimizer (GWO) and a modified version in solving path planning problems. Specifically, we aim to minimize the path length from a start point to a goal point while avoiding randomly placed obstacles.

The primary goals are:

- To compare path lengths and convergence rates of the standard GWO and the modified GWO.
- To evaluate the algorithms' ability to maintain a minimum safe distance from obstacles.
- To examine whether the modified version offers performance gains over the standard GWO.

Methodology

– **Experimental Setup**

The path planning problem is simulated within a bounded 2D space with randomly generated start and goal points and obstacles. The obstacles are placed randomly in the environment, respecting a minimum safe distance constraint from both the start and goal points and from each other.

– **Algorithms Used**

- Standard GWO: This algorithm follows the original GWO process for updating the positions of the agents.
- Modified GWO:The modification introduces an adjusted weight for each leader's influence on the position update, emphasizing the alpha wolf's influence more than that of the beta and delta wolves.

– **Assumptions**

- Obstacles are considered as circles of $radius = SafeDistance$.
- The robot/entity which moves from start to goal is considered a point object.

– **Parameters**

- Population size: Number of wolves in the pack.
- Maximum iterations: Upper limit for the number of iterations.
- Safe distance threshold: Minimum allowable distance from obstacles.
- Space Bounds: Limits of the 2D space.
- Number of obstacles: Total obstacles in the environment.

– **Difference between Standard and Modified GWO**

- Standard Grey Wolf Optimizer: The position update equation is used as-is from Equation (6), giving equal importance to the alpha, beta, and delta wolves.
- Modified Grey Wolf Optimizer:The modified position update equation is given by:

$$\vec{X}(t+1) = 0.6\vec{X}_1 + 0.3\vec{X}_2 + 0.1\vec{X}_3 \quad (8)$$

This equation places more influence on the alpha wolf, prioritizing its position over the beta and delta wolves.

– **Algorithm Setup**

- Selecting Start and Goal Positions: Start and goal positions are chosen randomly within the defined bounds of the environment to ensure the path planning problem has clearly defined endpoints. This initial random selection allows for flexibility in the setup.
- Generating Obstacles in the Environment: To simulate a realistic environment for path planning, a set number of obstacles is generated within the boundaries. Each obstacle is randomly placed in the environment, maintaining a safe minimum distance from both the start and goal points, as well as from other obstacles. This configuration ensures that the agent's path is challenged by obstacles while guaranteeing feasibility.
- Calculating Fitness of Agent Positions: The fitness function evaluates each agent's position based on proximity to the goal and penalizes it for being near obstacles within the safe distance.
- Grey Wolf Optimizers: The optimizers are configured according to the equations and the differences mentioned in the previous subsections.

– **Evaluation Metric**

The algorithms are evaluated based on path length and the fitness score convergence rate, incorporating both distance to the goal and penalties for proximity to obstacles .

Results

We ran the GWO and MGWO for two sets of parameters and obtained the following results.

Parameter Set 1

Parameter Name	Value
Pop_size	50
Max_T	75
Dim	2
Num_obstacles	30
safe_distance	10
xbound	[0, 200]
ybound	[0,200]

Table 1. Parameter Set 1

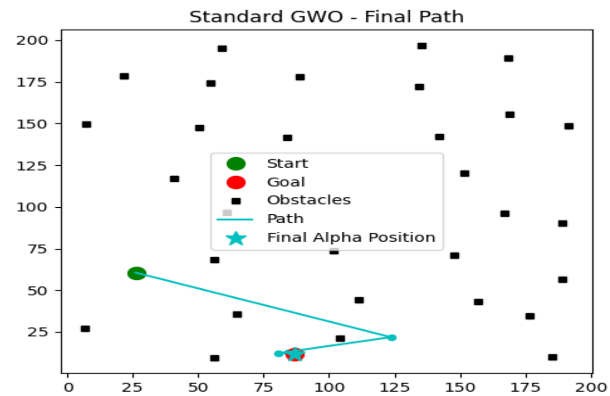

Fig. 3: GWO Path for Parameter Set 1

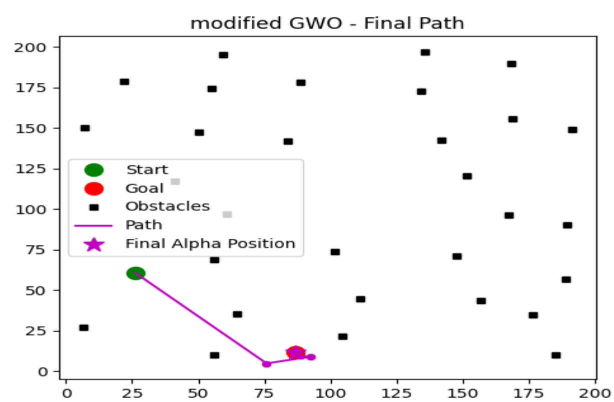

Fig. 4: MGWO Path for Parameter Set 1

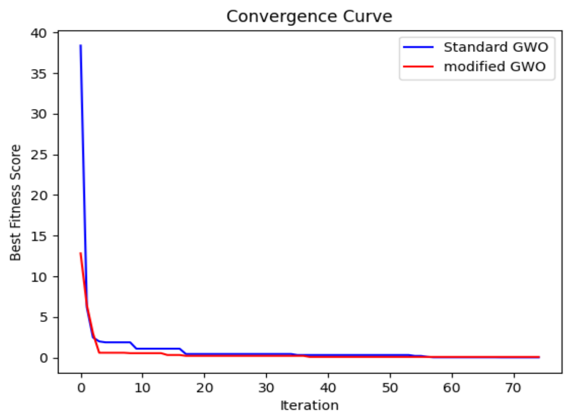

Fig. 5: Convergence Comparison for Parameter Set 1

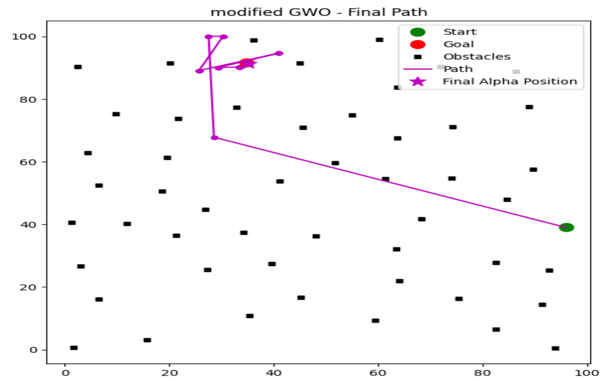

Fig. 7: MGWO Path for Parameter Set 2

Path Type	Value
Standard Path Length	165.33170595517302
Modified Path Length	100.00410729865557

Table 2. Path Lengths for Parameter Set 1

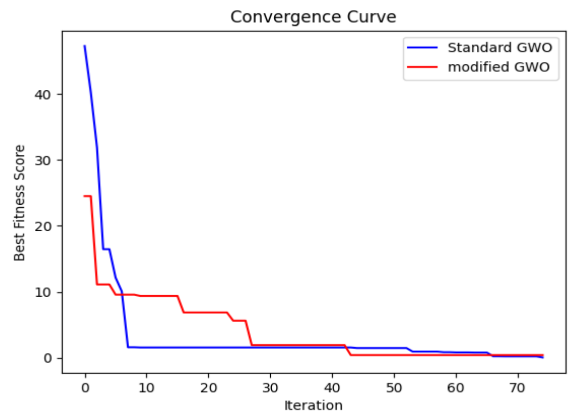

Fig. 8: Convergence Comparison for Parameter Set 2

Parameter Set 2

Parameter Name	Value
Pop_size	10
Max_T	75
Dim	2
Num_obstacles	50
safe_distance	5
xbound	[0, 100]
ybound	[0, 100]

Path Type	Value
Standard Path Length	179.83532826232798
Modified Path Length	154.78314868288112

Table 4. Path Lengths for Parameter Set 2

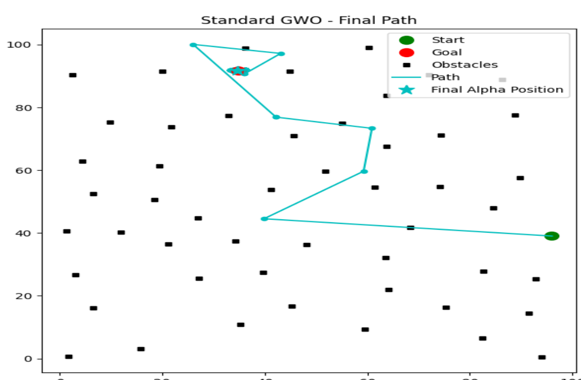

Fig. 6: GWO Path for Parameter Set 2

Here's Link to our GITHUB

The GITHUB repository includes the python codes for our GWO Optimizers, the results, parameters setting and standard IEEE Report for comparison purposes.

Conclusions

The results across both parameter sets show that MGWO outperforms the standard GWO in terms of path length and convergence rate. By emphasizing the influence of the alpha wolf, MGWO promotes more direct paths to the goal, avoiding obstacles more effectively. This suggests that the modified approach is a more efficient method for solving robot path planning challenges in complex environments.

References

[1] Lokman Dogan, U. YÜZGEÇ, "Robot Path Planning using Gray Wolf Optimizer",International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES'18)

[2] Lili Liu, Longhai Li,Heng Nian, Yixin Lu, Hao Zhao and Yue Chen , "Enhanced Grey Wolf Optimization Algorithm for Mobile Robot Path Planning"

[3] Xue Yang, Heng Yuxi, Lu Qiuhong,"Robot path planning based on the improved grey wolf optimization algorithm ".

[4] MIRJALILI S, MIRJALILI S M,LEWIS. A Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69:46-61.

[5] XU C F,DUAN H B,LIU F. Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path planning[J]. Aerospace Science and Technology, 2010, 14 (8):535-541.

[6] ZHANG S,ZHOU Y Q, LI Z M,et al. Grey wolf optimizer for unmanned combat aerial vehicle path planning[J]. Advances in Engineering Software, 2016, 99:121-136.

[7] YYahia, H.S.; Mohammed, A.S. Path planning in Unmanned Aerial Vehicles (UAVs): Overview, Challenges, and Solutions. Util. Math. 2023, 120, 455–471

[8] Liu, L.; Wang, X.; Yang, X.; Liu, H.; Li, J.; Wang, P. Path planning techniques for mobile robots: Review and prospect. Expert Syst. Appl. 2023, 227, 120254.

[9] SSabitri, P.; Muhammad, Y.; Sangman, M. Bio-Inspired Optimization-Based Path Planning Algorithms in Unmanned Aerial Vehicles: A Survey. Sensors 2023, 23, 3051

[10] S. Hosseini and A. Al Khaled, "A survey on the Imperialist Competitive Algorithm metaheuristic: Implementation in engineering domain and directions for future research," Applied Soft Computing Journal, vol. 24. pp. 1078–1094, 2014.

[11] Q. Luo, S. Zhang, Z. Li, and Y. Zhou, "A Novel Complex-Valued Encoding Grey Wolf Optimization Algorithm," Algorithms, vol. 9, no. 1, p. 4, 2015.

[12] M. Kohli and S. Arora, "Chaotic grey wolf optimization algorithm for constrained optimization problems," J. Comput. Des. Eng., 2017.

[13] C. Alexopoulos and P. M. Griffin, “Path planning for a mobile robot,” IEEE Trans. Syst. Man. Cybern., vol. 22, no. 2, pp. 1132–1139, 1992.

[14] N. Sariff and N. Buniyamin, “An overview of autonomous mobile robot path planning algorithms,” in Research and Development, 2006. SCOReD 2006. 4th Student Conference on, 2006, pp. 183–188.

[15] C. Hofner and G. Schmidt, “Path planning and guidance techniques for an autonomous mobile cleaning robot,” Rob. Auton. Syst., vol. 14, no. 2–3, pp. 199–212, 1995.

[16] Yarpiz, Optimal Robot Path Planning using PSO. http://yarpiz.com/403/ypa path-planning, Accessed 28 March 2018.