Paper Title\* (use style: *paper title*)

\*Note: Sub-titles are not captured in Xplore and should not be used

Dr. Upma Jain   
*Computer Science and Engineering*  
*Graphic Era (Deemed to be) University*Dehradun, India  
line 5: email address or ORCID  
  
Ashish Upadhyay  
*Computer Science and Engineering*  
*Graphic Era (Deemed to be) University*Dehradun, India  
upadhyay.ashish246@gmail.com  
  
Parth Sarthi  
*Computer Science and Engineering*  
*Graphic Era (Deemed to be) University*Dehradun, India  
parthsarthi.mpi@gmail.com  
  
Nikita Kumari  
*Computer Science and Engineering*  
*Graphic Era (Deemed to be) University*Dehradun, India  
nikita00413k@gmail.com

Puru Singh  
*Computer Science and Engineering*  
*Graphic Era (Deemed to be) University*Dehradun, India  
Puru.singh@outlook.in

*Abstract*— *This work explores the path planning in mobile robotics and provides an approach for single robot path planning using nature inspired algorithms. Among various nature inspired algorithms PSO is widely adopted by the researchers due to its ease of use and promising results for path planning. This work provides a concatenated PSO and GWO based approach for single robot path planning. The efficiency of the proposed approach is verified by evaluating its performance in a simulation environment for different target locations and comparing it to performance of other approaches.*

Keywords— Particle Swarm Optimization (PSO), Path Planning, Grey Wolf Algorithm (GWO)

# Introduction

Path planning is the vital process of rationally setting up a route for an object to travel from point A to point B while avoiding collisions. Given the evolving financial stresses we face, this process resembles a vigilant navigator seeking the safest and most efficient way to reach a destination with or without obstacles.

Path planning impacts several sectors, including military operations, search and rescue missions, and gaming. In the military field, path planning is indispensable as it reduces operational risks, thereby optimizing efficiency. In search and rescue missions, it provides vital aid to responders, enabling them to rescue and assist people in need. In gaming, it allows for the intelligent control of virtual characters, providing them with an understanding of the terrain. Without the critical capabilities of path planning techniques, addressing numerous real-life problems would be impossible.

Path planning algorithms are the solution to problem-seeking methods. Some of the top ones are A\*, Dijkstra’s algorithm, Particle Swarm Optimization (PSO), and the Grey Wolf Optimizer (GWO). A\* utilizes a heuristic formula to calculate the cost to a random point and from there to the goal, minimizing the effort of moving. In contrast, Dijkstra’s algorithm searches all potential routes sequentially from the source point to find the shortest path, which can be time-consuming. PSO is inspired by the swarming behavior of creatures, where particles (solutions) collaborate to explore the problem space, running synchronously to find the solution. The Grey Wolf Optimizer (GWO) imitates the leadership structure and hunting mechanisms of real wolves. By mimicking wolves' societal organization and foraging strategy, GWO balances exploration and exploitation, making it more suitable for complex environments.

Although A\* and Dijkstra’s algorithms are commonly applied, they are not without exceptions. If the heuristic function is imprecise, A\* may generate suboptimal solutions and is memory-sensitive, making it resource-intensive, especially in large-scale environments with complex terrains. Additionally, A\* struggles in dynamic scenarios where optimal pathways constantly change due to shifting obstacles or environmental conditions. Dijkstra’s algorithm, while straightforward and error-free in calculating the shortest path, is intensive and inefficient with large datasets or complex maps because it exhaustively searches every single node. In contrast, PSO excels at tuning parameters and escaping local optima effectively. It is time-efficient and navigates complex problem areas to find optimal solutions. The collective wisdom and harmony of particle coordination and favored search directions allow rapid trial and error until reasonable routes are found.

The Grey Wolf Optimizer (GWO) further simplifies the evolutionary process with its unique representation of the apex hierarchy structure and collective hunting strategy among grey wolves. This algorithm alternates between exploration and exploitation phases, providing robust performance in diverse environments. GWO's adaptability and dynamic nature enhance the efficiency of navigation tools, especially in situations requiring both plan adjustments and precise maneuvers.

In this paper, a combined approach that utilizes the features of Particle Swarm Optimization (PSO) technique and the Grey Wolf Optimizer (GWO) is provided to enhance the performance of PSO for path planning.

Especially it will research in what manner the Integration of PSO and GWO algorithms together with cubic spline methods can be approached to achieve highly performing and optimal trajectories. This paper attempts to achieve this goal by delving into the operational aspects and synergy of PSO, GWO, as well as cubic spline smoothing methods to enable the required improved precision and efficiency to enhance robotic navigation systems in complex environment.

# LITERATURE REVIEW

Path Planning has been applied in guiding the mobile robots to reach a particular objective from very simple trajectory planning to the selection of a suitable sequence of action.

Path Planning for mobile robots is divided into two categories: [1] Global Path Planning and Local Path Planning.

The Global Path Planning requires a map of environment to calculate the best route. Depending on the analysis of the map, some methods are based on Roadmaps are proposed.

To transform the Global Path into suitable waypoints, the Local Path Planning creates new waypoints taking into consideration the dynamic obstacles and the vehicles constraints.

## The Spectrum of Global Path Planning Algorithms

Global path planning is the core of robot navigation, with the goal of charting a collision-free trajectory while traveling the lowest distance possible. Various techniques have been developed and used to optimize global pathways. This spectrum includes:

1. *A∗ Algorithm* [2]: A heuristic approach capable of finding the optimal path by analysing the graph in a more informed way compared to classic search algorithms.
2. *Ant Colony Optimization* (ACO) [3]: A probabilistic method of tackling computational problems but with disadvantages such as delayed convergence and a proclivity to converge to local optima.
3. *Genetic Algorithm* (GA) *[4]:* A genetic algorithm is used to find the optimal path for a mobile robot to move in a static environment expressed by a map with nodes and links.
4. *Salp Swarm Algorithm* *(SSA)* [5]*:* Inspired by the swarming behaviour of salps when navigating and foraging in oceans, SSA is tested on several mathematical optimization functions to observe and confirm their effective behaviours in finding the optimal solutions for optimization problems.
5. *Particle Swarm Optimization (PSO)* [6]: An evolutionary algorithm that has been used to solve wide range of engineering and technological challenges. It has been regularly changed to improve its convergence properties.
6. *Improved Particle Swarm Optimisation* [7]: In this paper, an improved version of the traditional PSO was implemented. The inertia of the particles is varied throughout the iterations, the values of and are also changed based upon this modified value of the inertial weight. The formula used for the modification of inertia with each iteration is given as

(2.1)

(2.2)

Where and are the upper and lower limits of the inertia, *it* is the current iteration and *MaxIt* is the maximum number of iterations, is the modified inertia.

The value of and is calculated by equations 2.3 and 2.4:

(2.3)

(2.4)

1. *Improved PSO-GWO Algorithm*[8]: In this paper, Grey Wolf optimisation was used in combination with improved PSO. The modification of inertial weight after each iteration is adaptive based on the current and the of previous iteration. The formula for this inertia is given by:

(2.5)

(2.6)

where and are the upper and lower limits of inertia respectively, *k* is the current iteration and *MaxIter* is the maximum number of iterations, is the global best for current iteration, and is the global best of previous iteration.

## Contributions and Structure

Identify applicable funding agency here. If none, delete this text box.

In this paper, we suggest a new strategy that combines the exploitation ability of particle swarm optimization (PSO) and the exploration ability of the grey wolf optimizer (GWO).

# METHODOLOGY

This section outlines the approach taken to implement PSO along with GWO in a concatenated manner with adaptive inertial weight to find the optimal path from start to end coordinates while avoiding collision with the obstacles.

## Obstacle Setup

In this work, we have chosen circular obstacles of arbitrary positions and radii that are spread unevenly in the search space. The goal is to avoid collision with these obstacles when the robot navigates form the start to end coordinates.

## Random Initialization of Points

After the obstacles as defined and setup in the search space, random initialization of the swarm takes place. A method generates an arbitrary number of particles in the search space, each with its own x and y coordinates as well as an initial velocity. The generation of the random particles is done such that they do not exceed the search space.

## Particle Swarm Optimization Algorithm

In Particle Swarm Optimization (PSO), the execution begins with a set of particles in a search space where each particle denotes a possible solution. In our implementation, each particle consists of a list of points, which when joined together form a path from the start to end coordinates while avoiding the obstacles in between. Each particle has its own inertia to control the convergence of the algorithm. The aim is to try and minimize this path length while avoiding any collisions.

The algorithm runs for a set number of iterations. In each iteration it calculates the local and global best positions for each particle and adjusts the velocities and positions of all the other particles.

The velocity and positions of the particles are updates by the given formulae

(3.1)

(3.2)

is the velocity of the particle, is the inertia, and are two random numbers in the range [0,1], and are cognitive and social components for the particle and are taken as 2.05 in this work, is the global best position, is the position of the current paeticle, ans is the personal best of the current particle. is the position of the current particle and is its velocity.

Throughout each iteration, the particles navigate through the search space, constantly refining their positions and speeds to get closer to the best solutions. Moreover, the algorithm ensures the safety of particle movements by carefully examining their positions in relation to any obstacles in the environment. This precautionary measure prevents collisions between particles and obstacles, guaranteeing smooth navigation towards potential solutions.

As the algorithm progresses through each cycle of iterations, it fine-tunes the positions of particles, drawing from both their individual journeys and the collective wisdom of the swarm. This iterative journey brings particles closer and closer to the best possible solutions. Along the way, they learn from both their personal experiences and the insights gained from the entire group, guiding them towards optimal outcomes.

## Integration of Cubic Spline Interpolation

This section plots a smooth path using a spline interpolation technique. It takes the best location points as input and extracts the X and Y coordinates. Then, it includes the start and end points in the spline calculation. Using Cubic Spline interpolation, it generates smooth X and Y coordinates.

A group of math symbols

Description automatically generated

Fig 1. Showing the Range for the Cubic Spline Formula

## Concatenation approach: Implementing GWO after PSO

After applying PSO, the GWO algorithm is used to improve the search. This concatenated method overcomes the individual weaknesses inherent in both algorithms and enhances the overall optimization process. The wolves in this concatenated optimization technique are chosen from the top three particle subgroups of the PSO output: alpha, beta, and delta wolves. These elite entities represent the top solutions found previously and can be used to guide and correct the other members of the swarm.

It is the alpha, beta, and delta particles of the GWO method which are responsible for the movement of the remaining particles, called the omega wolves. These particles update their positions relative to their distance from the alpha, beta, and delta particles.

The formula for updating the positions of the particles with respect to the alpha, beta, and delta wolves is given by:

(1)

where 𝑓(𝛼, 𝛽, 𝛿) represents the function that updates the location of all the wolves based on the positions of the , , and wolves respectively. The distance vectors and positions of the , , and wolves are calculated by the given equations:

(3.3)

(3.4)

(3.5)

(3.6)

, , are the distance vectors for , , and wolves respectively. , , and have the values 2\* where is a random number in the range [0,1]. *X* is the position vector of the current wolf. a is initialized with 2 but decreases over each iteration. is a random number between [0,1]. , , are position vectors calculated by , , and wolves, the average of which is the new position of the current w.

The steps of the PSO+GWO algorithm with adaptive inertial weight for path planning are as follows:

1. The obstacles are set up in the environment and the parameters for PSO and GWO are initialized.
2. Random initialization of the swarm is done.
3. PSO is executed for each particle for the given number of iterations, the inertial weight is adjusted by using the equations 2.5 and 2.6.
4. Velocity and positions of the particles are updated depending upon the global best and particle best calculated, by using the equations 3.1 and 3.2.
5. If collision is detected on this new position, the position is shifted to avoid it and PSO is again implemented for the same particle again.
6. After PSO, GWO is implemented on the same swarm, which have their positions updated by PSO. The positions of the particles are updated by using the equations 3.3 and 3.4.
7. Collision detection and avoidance are the same as in step 5.
8. The final particle is returned after the maximum number of iterations have been done and plotted, along with the generation of cubic spline between the path coordinates.

**A screenshot of a phone

Description automatically generated**

Fig 2. Flowchart Showing the Proposed Methodological Approach.

# RESULT AND DISCUSSION

The performance of proposed concatenated PSO+GWO approach approach has been compared with other methods PSO[] and improved pso[] in a simulated enviornemnt.

For testing the performance of algorithm, a 25 \* 25 grid environment was considered with three circular obstacles in search environment. The values for c1 and c2 for PSO were taken as 2.05 for the fixed inertia method. The velocity was taken as 4 for all the algorithms. the values for max inertia and min inertia were taken to be 0.9 and 0.4 for adaptive PSO+GWO and 0.7 and 0.4 for variable inertia PSO respectively. The findings also indicate that the use of PSO + GWO with adaptive inertia was effective in the optimization exercises compared to other methods such as PSO with fixed inertia, PSO + GWO with fixed inertia, as well as PSO with variable inertia. The start coordinates were (0,0) for all the test simulations. The number of iterations was 150 and the swarm size was set to be 150. The code was run 20 times for each destination. The main outcomes are summarized in Table 4.1 below.

TABLE 4.1 COMPARISON OF PATH LENGTHS FOR VARIOUS ALGORITHMS.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Destination | PSO with fixed inertia (units) | PSO + GWO with fixed inertia (units) | PSO with variable inertia (units) | PSO+GWO with adaptive inertia (units) |
| (20,20) | 33.54 | 32.37 | 34.82 | 31.38 |
| (15,22) | 31.67 | 30.52 | 37.71 | 30.63 |
| (12,12) | 23.51 | 21.12 | 23.77 | 20.76 |
| (15,5) | 30.04 | 28.89 | 18.88 | 16.09 |
| (5,20) | 30.08 | 28.69 | 32.95 | 26.78 |

A graph with dots and a dotted line

Description automatically generated

Fig. 3. (a) Path visualization for PSO with fixed inertia

A graph of a graph with circles and a dotted line

Description automatically generated

Fig. 3. (b) Path visualization for PSO + GWO with fixed inertia

A graph of a graph with circles and a dotted line

Description automatically generated

Fig. 3. (c) Path visualization for PSO with variable inertia

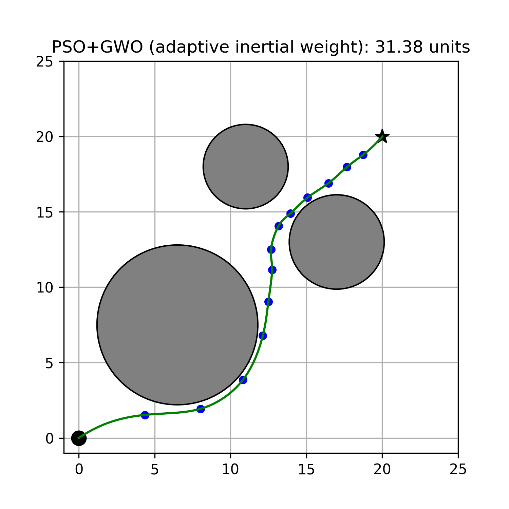


Fig. 3. (d) Path visualization for PSO + GWO with adaptive inertia

Another key parameter is the path length since shorter pathway implies higher efficiency and lower energy cost. The adaptive inertia approach with PSO+GWO yielded the shortest paths followed by the approach of PSO+GWO with fixed inertia.

By looking into the individual capabilities of both the PSO and the GWO, it can be deduced why the PSO + GWO with adaptive inertia is significantly more effective. PSO is popular with high convergence speed which helps well in path planning problems in terms of adaptation rate. Ecologically, it tends to exhibit premature convergence more often, especially when working in complex settings. GWO performs well in exploration and keeps the algorithm from premature convergence to a local minimum. Adaptive inertial based PSO and GWO are complementary where PSO has fast convergence rate while GWO has better exploration characteristics, which are integrated into a single framework.

# CONCLUSION

The study has successfully proven that the use of the combination of Particle Swarm Optimization (PSO) and Grey Wolf Optimization (GWO) does optimize path plans and is also good at handling obstacles for robotics. Analysis reveals that the concatenated PSO+GWO algorithm demonstrably surpasses both PSO and GWO algorithms in respect to accuracy, and path length.

The concatenated approach of PSO+GWO algorithm with adaptive inertia performed the best among all the other approaches it was compared with.

The PSO + GWO algorithm will be enhanced in future by using other problem-solving strategies. Besides, the research will involve operating the prototype in a broad spectrum of the real-world context, to understand the practical side and conduct studies on the influencing factors (optimal implementation).

##### References

1. K. Hao, J. L. Zhao, B. B. Wang, Y. L. Liu, and C. Q. Wang, "The application of an adaptive genetic algorithm based on collision detection in path planning of mobile robots," *Computational Intelligence and Neuroscience*, vol. 2021, 2021.
2. Z. Zhang, Y. Wan, Y. Wang, X. Guan, W. Ren, and G. Li, "Improved hybrid A\* path planning method for spherical mobile robot based on pendulum," in *International Journal of Advanced Robotic Systems*, vol. 18, no. 1, 2021.
3. C. Miao, G. Chen, C. Yan, and Y. Wu, "Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm," *Computers & Industrial Engineering*, vol. 156, pp. 107230, 2021.
4. R. Sarkar, D. Barman, and N. Chowdhury, “Domain knowledge based genetic algorithms for mobile robot path planning having single and multiple targets,” *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 7, 2022.
5. X. B. Cheng, L. C. Zhu, H. H. Lu, J. Z. Wei, and N. Wu, "Robot path planning based on an improved salp swarm algorithm," *Journal of Sensors*, vol. 2022, Article ID 2559955, 16 pages, 2022.
6. Yu, Z. Si, X. Li, D. Wang, and H. Song, "A novel hybrid particle swarm optimization algorithm for path planning of UAVs," *IEEE Internet of Things Journal*, vol. 9, no. 22, pp. 22547–22558, 2022.
7. Xun Li, Dandan Wu, Jingjing He, Muhammad Bashir, Ma Liping, "An Improved Method of Particle Swarm Optimization for Path Planning of Mobile Robot", *Journal of Control Science and Engineering*, vol. 2020, Article ID 3857894, 12 pages, 2020.
8. Cheng X, Li J, Zheng C, Zhang J and Zhao M (2021) An Improved PSO-GWO Algorithm with Chaos and Adaptive Inertial Weight for Robot Path Planning. *Front. Neurorobot.* 15:770361.