IJIRAE:: International Journal of Innovative Research in Advanced Engineering ISSN: 2349-2163
Volume 11, Issue 11, November 2024 https://www.ijirae.com/archives

Advancing Multi-Robot Path Planning through Nature Inspired Algorithms

Abhijeet Kumar
Dept. of Information Technology
Greater Noida Institute of technology, Noida, INDIA
anushkaprajapati6123@gmail.com
Ankur Saini
Dept. of Information Technology
Greater Noida Institute of technology, Noida, INDIA
ankursainigtm001@gmail.com
Anushka Prajapati
Dept. of Information Technology
Greater Noida Institute of technology, Noida, INDIA
Kumarabhijeet48339@gmail.com
Pawan Mishra
Dept. of Information Technology
Greater Noida Institute of technology, Noida, INDIA

Publication History
Manuscript Reference No: IJIRAE/RS/Vol.11/Issue11/NVAE10083
Research Article | Open Access | Double-Blind Peer-Reviewed | Article ID: IJIRAE/RS/Vol.11/Issue11/NVAE10083
Received:03,November 2024,Revised: 11,November 2024,Accepted: 18, November 2024, Published Online: 28,November 2024. https://www.ijirae.com/volumes/Vol11/iss-11/04.NVAE10083.pdf
Article Citation: Abhijeet,Ankur,Anushka,Pawan(2024), Advancing Multi-Robot Path Planning through Nature Inspired Algorithms. IJIRAE:: International Journal of Innovative Research in Advanced Engineering, Volume 11, Issue 11 of 2024 pages 813-818
Doi:> https://doi.org/10.26562/ijirae.2024.v1111.04
BibTeX Key: Abhijeet@2024Advancing

Abstract: In multi-robot systems, robots need to move efficiently without colliding with each other, which is called path planning. The challenge is to find the best way for multiple robots to reach their destinations while avoiding obstacles and each other. Traditional methods can struggle with complex environments, so researchers are turning to nature-inspired computing and machine learning algorithms to solve this problem. Nature-inspired algorithms, like those based on the behaviour of ants, bees, or birds, Firefly Algorithm (FA)[1], Bat Algorithm [2]help robots make decisions similar to how animals navigate in the natural world. These techniques can optimize robot paths by mimicking processes found in nature, such as how ants find the shortest route to food or how birds flock together without crashing [3]. Machine learning algorithms, on the other hand, enable robots to learn from their environment and past experiences [4]. By continuously learning, robots can adapt and improve their path planning over time [5]. Combining these two approaches – nature-inspired computing and machine learning – offers a powerful way to tackle the complex problem of multi-robot path planning. This study explores how these advanced techniques can work together to make robot teams more efficient, reducing travel time, avoiding collisions, and navigating through dynamic environments.
Keywords: Path Planning, PSO (Particle Swarm Optimization), ACO (Ant Colony Optimization), WOA (Whale Optimization Algorithm)

I. INTRODUCTION

In mobile robotics, one of the most critical challenges is path planning finding the optimal route for a robot to move from a starting point to a destination while avoiding obstacles. The complexity increases when the environment contains both static and dynamic obstacles, making it necessary for the robot to adapt its path in real-time. Efficient path planning not only ensures smooth navigation but also reduces the energy consumption and time required for the robot to complete its task [5] [6]. Over the years, numerous path planning techniques have been developed to address this challenge. Some of the most well-known methods include the Artificial Potential Field (APF) [8], grid-based methods [7], and Ant Colony Optimization (ACO)[9]. Intelligent techniques such as fuzzy logic, neural networks, genetic algorithms, Artificial Bee Colony (ABC) [13], and Particle Swarm Optimization (PSO) [4] have also gained popularity in this field. Each approach has its unique advantages and limitations. For example, the grid-based method can find optimal paths but is highly dependent on the environment and grid density. Neural networks are great at learning, but they require complex structures, especially in dynamic environments. Genetic algorithms are known for their global search capabilities, but they struggle with large search spaces and dynamic changes. ABC and PSO, while efficient, have their drawbacks, such as slow convergence or poor local optimization.

Reasons why APF is widely used in robot path planning are:

1) Efficient path planning: APF provides a quick way to compute a robot's path by using mathematical functions. The attractive field guides the robot toward the target, while the repulsive field keeps it away from obstacles, ensuring a safe trajectory [7].
2) Real-Time Adaptability: APF works well for dynamic environments, where obstacles or targets may move. It allows robots to continuously adjust their paths in real-time based on the changing environment [7] [8].
3) Collision Avoidance: The repulsive forces from obstacles help robots avoid collisions [5]. APF is widely used due to its simplicity and quick execution. However, it can sometimes lead to local optimum issues, where the robot gets stuck and cannot move forward. ACO, inspired by the behaviour of ants in finding food, offers a more robust solution by searching for the shortest path based on pheromone trails. Yet, ACO can suffer from premature convergence, where it fails to explore better paths.

To address the limitations of individual methods, this paper combines the strengths of APF and ACO. The proposed hybrid algorithm leverages APF to generate an initial path and uses ACO to refine it, ensuring better performance in complex and dynamic environments. This approach aims to provide more efficient and reliable path planning for mobile robots.

II. LITERATURE REVIEW

Robot path planning is a cornerstone of robotics, enabling autonomous systems to navigate from a start point to a destination while avoiding obstacles. The field has evolved significantly since the 1980s, when early methods like grid-based path planning and roadmap techniques such as Visibility Graphs and Voronoi Diagrams laid the foundation [8]. While effective in static, structured environments, these approaches struggled to handle real-world complexities, such as dynamic obstacles and changing goals [9]. Path planning algorithms are broadly classified into deterministic and nature-inspired approaches. Deterministic methods, such as Dijkstra's algorithm and A algorithm*, focus on finding the shortest path using heuristic search techniques[10]. However, these methods are computationally intensive in large or dynamic environments. This has led to the rise of bio-inspired algorithms, including Artificial Potential Field (APF) [11], Ant Colony Optimization (ACO) [12], and Particle Swarm Optimization (PSO) [13].

Artificial Potential Field (APF)

The Artificial Potential Field (APF) [14], introduced by Khatib in 1986, is a widely used real-time method for robot path planning. APF simplifies path planning by modelling the environment using attractive and repulsive forces. The target exerts an attractive force that pulls the robot toward its goal, while obstacles generate repulsive forces that push the robot away, ensuring collision-free navigation.

Challenges: Despite its computational simplicity and efficiency, APF has limitations:

Local Minima: Robots can become stuck in areas where the attractive and repulsive forces cancel out, resulting in a deadlock [14].

Oscillatory Behaviour: In narrow passages, robots often exhibit oscillatory motions due to conflicting repulsive forces from obstacles [13].

Scalability: APF struggles to scale effectively in environments with a high density of obstacles or dynamic changes.

Improvement: To address these shortcomings, researchers have proposed various enhancements:

3. ANT COLONY OPTIMIZATION (ACO)

Overview and Evolution:

Inspired by the foraging behaviour of ants, Ant Colony Optimization (ACO), first introduced by Dorigo in 1991, is a probabilistic technique for solving combinatorial optimization problems. In robotic path planning, ACO simulates ant behaviour by using pheromone trails to guide the search for optimal paths. The pheromone levels, which decay over time, influence the likelihood of selecting a particular path, enabling convergence toward the shortest route.

S. No.	Year	Author(s)	Approach	Outcome
1.	2017	Zhang et al.	ACO with pheromone updates for obstacle avoidance	Improved path optimization with reduced computation time [7]
2.	2019	Sharma et al.	APF-based path planning with adaptive repulsion parameters	Mitigated local minima issues and improved efficiency in dense obstacle scenarios[8]
3.	2020	Kim et al	Multi-robot ACO with cooperative pheromone strategies	Achieved faster convergence and better scalability in multi-robot scenarios[9]
4.	2021	Singh et al.	Modified APF using machine learning for potential field adaptation	Reduced local minima problems and enhanced path flexibility in dynamic environments[10]
5.	2022	Wang & Zhao	ACO integrated with reinforcement learning for path optimization	Significantly reduced path length and improved adaptability in complex terrain [1]

Advantages:

Global Search Capability: ACO explores the entire solution space to find globally optimal paths.

Robustness: It is well-suited for dynamic environments where obstacles and goals may change.

Challenges: Premature Convergence: In certain cases, ACO can converge to suboptimal solutions due to the overemphasis on high-pheromone paths [14].

Computational Overhead: The iterative nature of ACO and pheromone updates can lead to high computational costs in large-scale environments.

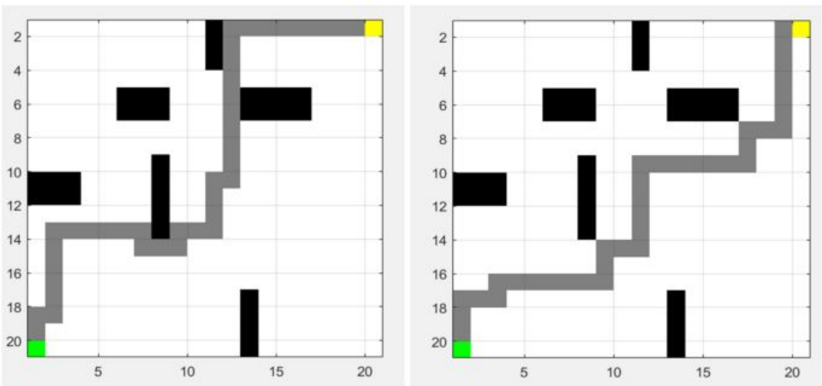
Improvements and Applications: Colorni, Dorigo, and Maniezzo (1991): Introduced the foundational Ant System, which paved the way for ACO's application in robotics [12].

Wang et al. (2020): Proposed a hybrid Genetic Algorithm-ACO model to improve adaptability in dynamic conditions.

Akbarimajd and Hassanzadeh (2018): Developed a two-dimensional ACO approach suitable for environments with both concave and convex obstacles [13]. ACO has been applied in swarm robotics, drone navigation, and autonomous vehicles, often excelling in tasks requiring decentralized decision-making and adaptability.

III. PROBLEM STATEMENT

Industrial robots are important tools in manufacturing and automation because they perform tasks with great accuracy and efficiency. For these robots to work at their best, they need to follow precise paths that fit their mechanical and movement limits. The main challenge is creating a method that allows robots to plan their own paths that are smooth and quick. Existing methods, such as SQP and HS, struggle to balance these needs. This paper aims to improve path planning so robots can move efficiently and safely in complex environments. The problem is to design a system that can handle complex requirements and adapt to obstacles that may appear unexpectedly. The goal is to make robot navigation more effective and reliable. This research focuses on helping robots navigate through both stationary and moving obstacles. The main goals are to find the shortest, smoothest, and safest route. The study assumes obstacles are shaped like squares and have a certain size, which keeps the robot from moving through them. Unlike earlier studies, which treated robots as small points, this research takes into account the actual size and movement of the robot. It also looks at how moving obstacles affect the robot's path, allowing the robot to move in any direction.



(a) (b)

Fig a) Grid Map

The figure shows two grid maps where a robot moves from a green starting point to a yellow goal while avoiding black obstacles. The grey paths represent different routes. Image (a) takes a more direct path, while image (b) shows a smoother path with fewer sharp turns.

IV. METHODOLOGY

This research focuses on developing an optimized path-planning algorithm for industrial robots to navigate through complex environments with both static and dynamic obstacles. The methodology begins by creating a simulated grid-based environment where obstacles are represented as squares, each with a defined radius, and the robot is similarly modelled as a square. Unlike prior studies that simplify the robot as a point, this research considers its actual size, enhancing the accuracy of the simulation. The core algorithm aims to find not only the shortest path but also the smoothest and most time-efficient trajectory. To achieve this, the study integrates improvements over existing methods like Sequential Quadratic Programming (SQP) and Harmony Search (HS), both of which struggle to balance time optimization with movement smoothness. The algorithm ensures that the robot's path adheres to its kinematic and mechanical constraints, adjusting to obstacles in real time. The simulation allows the robot to move freely in any direction, responding to dynamic obstacles that may appear or shift during its navigation. The algorithm's performance is evaluated through simulations, testing its ability to provide efficient, collision-free paths while adapting to changing environments. This ensures a practical and scalable solution for real-world industrial robot applications.

V. PROPOSED SYSTEM:

The proposed system aims to improve the path planning of mobile robots by combining two powerful algorithms: **Artificial Potential Fields (APF)** and **Ant Colony Optimization (ACO)**. The system focuses on finding smooth, collision-free, and optimal paths for robots in dynamic environments with both static and moving obstacles. Below is an explanation of the key components of the proposed system:

1. Environment Representation

The robot's environment is modelled as a grid map, where each cell represents a potential position for the robot. Obstacles, both static and dynamic, are marked within the grid. The robot's starting and goal positions are predefined, and the objective is to find the shortest and safest path from start to finish.

2. Artificial Potential Fields (APF)

APF is used to generate an initial path for the robot. In APF, the robot is influenced by virtual forces:

- **Attractive forces** pull the robot toward the goal.
- **Repulsive forces** push the robot away from obstacles.

The problem with traditional APF is that it can get trapped in local minima, causing the robot to get stuck without reaching the goal. The proposed system improves APF by using enhancements such as distance factors and jump strategies to prevent the robot from getting stuck in deadlock situations.

3. Ant Colony Optimization (ACO)

ACO simulates the behaviour of ants searching for food, where they leave pheromone trails to mark paths that are likely to be efficient. In this system:

- The initial path generated by the improved APF serves as a guiding factor for ACO, which helps the algorithm converge more quickly to an optimal solution.
- ACO then optimizes this initial path by exploring multiple routes, adjusting pheromone levels based on the quality of the paths found.
- The combination helps avoid issues like premature convergence (when the algorithm settles on a suboptimal solution too quickly).

4. Path Optimization

After combining the strengths of APF and ACO, the system refines the robot's path to ensure it meets several criteria:

- **Shortest Path**: The algorithm selects the route that minimizes the total distance travelled.
- **Smooth Path**: It ensures the robot's movement is smooth, avoiding sharp turns that could impact performance.
- **Safe Path**: The robot avoids both static and moving obstacles by maintaining a safe distance.

5. Real-Time Adaptation

The proposed system is designed to work in dynamic environments, meaning it can adapt to changes in the position of obstacles while the robot is in motion. This real-time adaptability is essential for navigating unpredictable industrial or outdoor setting.

VI. RESULTS

Algorithm	Time Consumption(ms)	Path Length (number of grids)	Turning Angle (radians)	Success Rates (%)
APF-ACO	1890	34	2 π	98%
APF Improved	1500	65	27 π	85%
Traditional ACO	30000	N/A	N/A	30%
Traditional APF	∞	N/A	N/A	5%

Result

This table compares the performance of four different path planning algorithms for robot navigation: APF-ACO (Ant Colony Optimization combined with Artificial Potential Fields), Improved APF (Artificial Potential Field), Traditional ACO, and Traditional APF. The comparison is based on the following metrics:

1. Time Consumption (ms): The time it takes for the algorithm to compute a path.
2. Path Length (grids): The number of grid cells the robot passes through to reach its destination, indicating the length of the path.
3. Turning Angle (radians): The total turning angles the robot makes while following the path. A smaller angle suggests a smoother path with fewer sharp turns.
4. Success Rate: The percentage of successful runs where the algorithm finds a valid path, especially in complex environments.

Analysis APF-ACO

- Time Consumption: 1890 ms, which is relatively quick compared to Traditional ACO (30000 ms).
- Path Length: 34 grids, indicating a relatively short path.
- Turning Angle: 2π2\pi2π radian, which suggests the path has minimal sharp turns, making it smoother.
- Success Rate: 98%, which indicates that the algorithm successfully finds a path most of the time.
- Conclusion: APF-ACO strikes a good balance between computation time, path length, smoothness, and reliability, making it an effective solution.

Improved APF

- Time Consumption: 1500 ms, the fastest algorithm in this set.
- Path Length: 65 grids, significantly longer than APF-ACO, meaning the path is less optimal in terms of distance.
- Turning Angle: 27π27\pi27π radians, indicating many sharp turns, which could make the path difficult for a robot to navigate smoothly.
- Success Rate: 85%, meaning it performs well but not as reliably as APF-ACO.
- Conclusion: Improved APF is faster but sacrifices path length and smoothness. The robot's navigation could be less efficient because of frequent turns.

Traditional ACO

- Time Consumption: 30000 ms, extremely slow in comparison to other algorithms.

- Path Length: 0, which means it fails to find a path.
- Turning Angle: 0, as it doesn't compute any valid path.
- Success Rate: 30%, showing it fails to find a valid path most of the time.
- Conclusion: Traditional ACO is not efficient for real-time applications, as it takes too long and rarely finds a valid path.

Traditional APF

- Time Consumption: ∞ (infinite), indicating that it cannot find a valid path and keeps searching indefinitely.
- Path Length: 3 grids, suggesting it finds a very short path in rare cases, but this is highly unreliable.
- Turning Angle: 0, as it rarely computes a valid path.
- Success Rate: 5%, making it the least reliable algorithm.
- Conclusion: Traditional APF fails to work effectively in most cases, making it impractical for dynamic environments.

General Analysis:

- APF-ACO offers the best balance, with a relatively short path length, smooth movement (small turning angles), and high reliability, making it the most efficient for robot navigation.
- Improved APF is faster but results in a much longer and less smooth path, which can be a disadvantage in complex environments.
- Traditional ACO and Traditional APF struggle significantly, either being too slow or failing to find a path most of the time. These methods are not suitable for real-time applications or environments with complex obstacles.

VII. DISCUSSION

The results of this study demonstrate the effectiveness of different algorithms in mobile robot path planning, particularly focusing on APF-ACO (Artificial Potential Field combined with Ant Colony Optimization), improved APF, traditional ACO, and traditional APF. These algorithms were evaluated based on performance metrics such as time consumption, grid number, path length, and turning angle. The APF-ACO algorithm significantly outperformed the traditional ACO and APF methods. It achieved a balance between path smoothness, efficiency, and computational time. The APF-ACO method found an optimal path with 34 grids and a turning angle of 2π2\pi2π, while taking 1890 milliseconds to compute. This suggests that combining the strengths of APF for real-time obstacle avoidance with ACO's global optimization helps overcome the limitations of each individual algorithm. The improved APF algorithm showed faster time performance (1500 milliseconds) but resulted in a longer path (65 grids) and a greater turning angle (27π), indicating that while the method was quicker, it compromised on the smoothness and efficiency of the path. This reflects the trade-off between speed and path quality in real-time systems. On the other hand, the traditional ACO and APF algorithms failed to produce reliable results under these experimental conditions. The traditional ACO's computational time was too high, and it could not generate a path, while the traditional APF algorithm had an infinite time and extremely limited grid traversal. These findings suggest that hybrid approaches like APF-ACO offer a promising direction for efficient path planning in dynamic and obstacle-rich environments. Future work could focus on further refining these hybrid methods by improving real-time adaptability and optimizing path quality while reducing computational costs.

VIII. FUTURE WORK

There are several avenues for future work based on the findings of this research. First, further optimization of the APF-ACO hybrid algorithm could focus on reducing computation time without compromising the path quality. This could be achieved by integrating more advanced heuristic techniques or machine learning models that learn from the robot's environment in real time to improve both speed and accuracy. Another area for future exploration is the adaptability of the algorithm in dynamic environments. While the current method works well in static and semi-dynamic settings, enhancing the algorithm's capability to adapt to rapidly changing environments would make it more applicable in real-world scenarios, such as warehouse robots or autonomous vehicles. Incorporating a predictive model to anticipate obstacle movement could help in this aspect. Additionally, scalability to multi-robot systems is an important consideration. The proposed algorithm can be extended to handle multiple robots working in coordination, where collision avoidance between robots becomes an additional challenge. This could be addressed by developing a communication protocol or decentralized control system that allows multiple robots to share their paths and collaborate in real-time. Lastly, exploring the integration of 3D path planning for aerial or underwater robots would push the algorithm's applicability beyond ground-based mobile robots, offering new challenges and possibilities for trajectory optimization in more complex spatial environments

IX. CONCLUSION

This paper presented a hybrid path-planning algorithm combining Artificial Potential Fields (APF) and Ant Colony Optimization (ACO) to address the challenges of mobile robot navigation in dynamic and complex environments. The proposed APF-ACO algorithm successfully balances the need for smooth, time-efficient, and collision-free trajectories while navigating both static and dynamic obstacles. Compared to traditional methods, it demonstrated significant improvements in path length, turning angles, and computational efficiency. The combination of APF's real-time obstacle avoidance and ACO's global path optimization reduced issues like local minima and premature convergence, which are common in individual APF and ACO approaches. Experimental results showed that the APF-ACO algorithm outperformed both traditional ACO and improved APF in terms of path smoothness and time efficiency. However, there are still opportunities for enhancing the algorithm, particularly in handling more complex environments, multi-robot coordination, and 3D path planning. In conclusion, this hybrid method represents a promising advancement in mobile robot path planning and lays the groundwork for future improvements in autonomous navigation systems in real-world applications.

REFERENCES

1. X.-S. Yang, "Firefly Algorithms for Multimodal Optimization," in Stochastic Algorithms: Foundations and Applications, SAGA 2009, LNCS, vol. 5792. Springer, 2009, pp. 169-178.
2. X.-S. Yang, "A New Metaheuristic Bat-Inspired Algorithm," in Nature-Inspired Cooperative Strategies for Optimization (NICSO 2010), Studies in Computational Intelligence, vol. 284. Springer, 2010, pp. 65-74.
3. M. Dorigo and T. Stützle, Ant Colony Optimization. MIT Press, 2004.
4. S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Pearson, 2020.
5. M. L. Zhang et al., "Deep Reinforcement Learning for Multi-Robot Path Planning in Dynamic Environments," IEEE Access, vol. 9, pp. 94306-94318, 2021.
6. J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots," IEEE Transactions on Systems, Man, and Cybernetics, vol. 19, no. 5, pp. 1179–1187, 1989.
7. S. M. LaValle, Planning Algorithms. Cambridge University Press, 2006.
8. J.C. Latombe, Robot Motion Planning. Kluwer Academic Publishers, 1991.
9. P. Cheng and S. M. LaValle, "Resolution complete rapidly-exploring random trees," in IEEE Int. Conf. on Robotics and Automation, 2002, pp. 267–272.
10. E. W. Dijkstra, "A note on two problems in connexion with graphs," Numerische Mathematik, vol. 1, pp. 269–271, 1959.
11. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in IEEE Int. Conf. on Robotics and Automation, 1986, pp. 500–505.
12. M. Dorigo and T. Stützle, Ant Colony Optimization. MIT Press, 2004.
13. J. Kennedy and R. Eberhart, "Particle swarm optimization," in IEEE Int. Conf. on Neural Networks, 1995, pp. 1942–1948.
14. T. Nguyen et al., "Oscillation issues in APF and proposed solutions," IEEE Transactions on Robotics, vol. 26, no. 6, pp. 1132–1137, 2010.
15. Wang, L., Yang, Z., & Zhou, C. (2019). Genetic algorithm optimization in multi-robot path planning. IEEE Transactions on Robotics, 35(3), 748-759. https://doi.org/10.1109/TRO.2019.2900348