Adaptive Particle Swarm and Ant Colony Optimization Path Planning for Autonomous Robot Navigation

Alami Essaadoui [1], Youssef Baba [2], Oussama Hamed [3*], Mohamed Hamlich [4], Chafik Guemimi [5], Ali EL Kebch [6]
[1,2,3,4,5,6] University of Hassan II, Ensam, CCPS Laboratory, Casablanca, Morocco
[3] Univ. Aix-Marseille, Laboratoire d'Informatique et Systèmes, Marseille, France
Email: [1] alami70@yahoo.fr, [2] youssefbaba@gmail.com, [3] oussama.hamed@univ-amu.fr, [4] moha.hamlich@gmail.com, [5] guemimichafik@gmail.com, [6] alibec_ma@yahoo.fr
*Corresponding Author

Abstract—**Path planning in cluttered and uncertain environments remains a significant challenge in robotics, autonomous navigation, and logistics optimization. This paper proposes a novel Adaptive Hybrid PSO-ACO Planner, which synergistically combines Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) to compute efficient paths in grid-based environments with static obstacles. Unlike traditional fixed-phase hybrids, our approach features a dynamic switching strategy between PSO and ACO based on real-time convergence behavior, allowing the algorithm to maintain progress and escape local minima. Additionally, adaptive parameter tuning is integrated to enhance the balance between global exploration and local exploitation throughout the search. The switching logic is governed by two criteria: a stagnation threshold that triggers phase transitions and a progress-dependent adaptation mechanism that adjusts search intensities over time. PSO dominates the early search phase, rapidly exploring the solution space, while ACO refines promising paths through pheromone-guided optimization in later stages. The proposed planner also includes a path reconstruction module to ensure solution completeness and robustness. Experimental evaluations on grid-based environments demonstrate that the proposed method consistently achieves higher path quality and faster convergence compared to standalone PSO and ACO approaches. Quantitative results demonstrate notable improvements in path efficiency and overall success rate across a range of obstacle densities. These advancements establish the Adaptive Hybrid PSO-ACO Planner as a robust and efficient tool for real-time and practical deployment in autonomous robot navigation systems.**

Keywords*—*Path Planning; Hybrid Metaheuristics; Particle Swarm Optimization (PSO); Ant Colony Optimization (ACO); Adaptive Switching; Robot Navigation

I. INTRODUCTION

Path planning is one of the most critical tasks in the field of robotics and autonomous navigation. Whether in the context of mobile robots, unmanned ground vehicles, or autonomous delivery systems, the ability to compute an efficient and collision-free path from a starting point to a desired goal remains a fundamental challenge [1]–[4]. This task becomes particularly complex in environments with static or dynamic obstacles, limited perception, or uncertainty [5]–[11].

Robots operating in such environments often deal with sensor noise, limited field-of-view, and dynamically changing surroundings, making static planning approaches insufficient. In these scenarios, a robot must intelligently explore the environment, reason about constraints, and adapt its trajectory in real time to ensure safe and successful navigation [12]–[18].

The importance of reliable and optimized path planning extends across a wide range of practical applications. In industrial settings, automated guided vehicles (AGVs) depend on efficient path planning to move goods within warehouses [19]–[25]. In urban mobility, self-driving cars must continuously plan their paths to navigate traffic while ensuring passenger safety [26]–[29]. In emergency and rescue missions, mobile robots are deployed in unknown terrains where path planning must be robust against unexpected barriers and limited sensor coverage [30]–[34]. Therefore, the effectiveness of path planning algorithms has a direct impact on the operational performance, reliability, and safety of robotic systems [35]–[43].

Over the years, numerous approaches have been proposed for solving the path planning problem. Classical methods such as Dijkstra's algorithm and A* provide deterministic solutions but often struggle with scalability and adaptability in complex or dynamic environments [44]–[48]. Metaheuristic techniques, including Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Genetic Algorithms (GA), have gained popularity due to their flexibility and global search capabilities [49]–[60].

However, these algorithms also suffer from limitations. PSO can rapidly explore the search space but may converge prematurely to suboptimal solutions [61], [62]. ACO is powerful for local exploitation but tends to slow down as the search progresses [63]–[66].

Hybrid models that combine these algorithms have been introduced, yet most of them rely on fixed or manually defined transitions, lacking real-time adaptability and responsiveness to convergence behavior [67]–[71].

Despite their promise, existing hybrid PSO-ACO approaches fall short in dynamic decision-making. They typically use static, fixed-sequence integration strategies, which are insensitive to real-time convergence behavior and unable to adapt based on performance feedback. Furthermore, many approaches introduce significant implementation complexity and computational cost, without adequately addressing their scalability in real-world robotic systems.

To overcome these limitations, this paper proposes an adaptive Hybrid PSO-ACO Path Planner that integrates PSO and ACO in a cooperative and adaptive framework [72]–[77].

The novelty of this approach lies in its dynamic phase-switching mechanism, which allows the planner to alternate between PSO and ACO based on the observed stagnation of convergence and the progress of optimization. PSO dominates the early search stage to encourage broad exploration, while ACO enhances promising paths using pheromone-based reinforcement during the later stages.

The algorithm also incorporates adaptive parameter tuning to continuously balance exploration and exploitation throughout the search process. The decision to transition between PSO and ACO is governed by two key factors: (1) a stagnation threshold that indicates when progress has plateaued, and (2) a progress-driven adaptation mechanism that tunes algorithmic parameters to maintain diversity early and promote convergence later. Initially, the algorithm employs PSO to explore the grid-based environment broadly, generating candidate paths through particle movements influenced by both global and personal bests. When improvement stalls or the search becomes localized, the planner transitions to ACO, where pheromone trails guide the search through high-quality regions. This switching process is bidirectional, allowing the system to re-invoke PSO if needed, and ensures that both exploration and exploitation are applied effectively and at the right moments.

Additionally, the planner integrates adaptive parameter tuning, a visibility-based heuristic, and a path completion and enhancement module that reconstructs or extends paths to ensure full connectivity from start to goal.

The main contributions of this work are: (i) an adaptive hybrid path planning framework that adaptively switches between PSO and ACO based on stagnation and progress, ensuring robustness and flexibility in complex environments; (ii) a progress-aware parameter adjustment scheme that balances exploration and exploitation throughout the search; and (iii) a path validation and enhancement mechanism that ensures solution completeness and quality without requiring a full reinitialization.

While our experiments are conducted in static grid-based environments, the proposed framework is designed with extensibility in mind. Future work will focus on dynamic scenarios, continuous space navigation, and real-time deployment using ROS-Gazebo on physical robots. We also aim to investigate the influence of noisy sensors and uncertainty-aware models to further enhance real-world applicability. Furthermore, we acknowledge that while our method improves adaptability, it introduces additional algorithmic complexity and computation cost. This trade-off is discussed further in the conclusion.

The proposed method has been rigorously evaluated and compared against standalone PSO, ACO, and other hybrid variants such as PSO-GA and GA-ACO. The results demonstrate that the Adaptive Hybrid PSO-ACO Planner consistently achieves superior performance in terms of path optimality, success rate, convergence behavior, and overall reliability. These findings confirm the effectiveness of the approach and underscore its potential for real-world applications in autonomous navigation systems.

The rest of the paper is structured as follows: Section 2 explains the theoretical background of PSO and ACO and details the proposed dynamic hybrid methodology. Section 3 discusses the experimental results and benchmarks. Finally, Section 4 concludes the paper and outlines directions for future research.

II. Related Works

Path planning in cluttered and dynamic environments is still one of the most challenging tasks in mobile robotics. Several nature-inspired optimization algorithms have been proposed to solve this problem, among them PSO and ACO are considered two of the most widely used approaches. PSO is a population-based stochastic optimization method introduced by Kennedy and Eberhart [78]. It mimics the social behavior of bird flocking and has shown fast convergence during the initial search phase. However, standard PSO often suffers from premature convergence and lacks robustness in complex environments. To overcome these limitations, hybrid versions of PSO with other metaheuristics have been proposed.

ACO, developed by Dorigo and Gambardella [79], is inspired by the pheromone trail behavior of ants. It is well adapted for solving discrete and combinatorial problems such as grid-based path planning. ACO performs well in terms of solution refinement but requires more iterations to converge, especially in large search spaces. To enhance performance, recent studies have proposed combining PSO and ACO into hybrid frameworks. These hybrid approaches exploit the global exploration capacity of PSO and the local exploitation capability of ACO.

In [80], a two-layer strategy was proposed for wheeled mobile robots by combining an improved ACO algorithm for global path generation with a Dynamic Window Approach (DWA) for local obstacle avoidance. Another work in [81] proposed a hybrid PSO-DWA method for unmanned surface vehicles, where adaptive parameter control was introduced to improve convergence and reduce oscillation near obstacles. In the context of aerial vehicles, [82] developed a hybrid dung beetle optimization algorithm for 3D path planning that integrates chaotic mapping and adaptive inertia to overcome local minima.

A recent work in [83] introduced a dynamic PSO-based planner, named OkayPlan, which considers obstacle kinematics in real time. It formulates the problem as a constrained optimization task and solves it using a modified PSO scheme. All these works demonstrate that hybrid and adaptive methods can significantly improve the efficiency and reliability of autonomous navigation in uncertain environments.

Despite these improvements, most hybrid strategies adopt static switching rules or fixed weights between phases, which limits their responsiveness to stagnation or sudden changes in the environment. In contrast, the method proposed in this paper introduces an adaptive hybrid PSO-ACO planner that dynamically switches between phases based on convergence behavior. A stagnation-based logic is used to alternate between PSO and ACO phases, while adaptive parameter adjustment ensures balanced exploration and exploitation during the entire planning process. Compared to existing approaches, this strategy improves robustness and convergence speed in dense obstacle environments.

III. METHOD

This section presents the principal techniques used in this work. Two nature-inspired algorithms are first introduced: PSO and ACO. These methods have demonstrated effectiveness in solving complex path planning problems. Afterwards, the proposed adaptive hybrid approach, which combines both techniques in an adaptive framework, is detailed. The method alternates between PSO and ACO phases based on the system's performance, allowing efficient exploration of the search space and rapid convergence toward an optimal path.

A. Particle Swarm Optimization

In PSO, each particle represents a potential solution in the search space, and all particles move within the space by adjusting their velocities and positions based on their own experience and the experience of neighboring particles.

The velocity $\vec{v}$ and position $\vec{p}$ updates of a particle are governed by the equations:

$$\vec{v}_i(t+1) = \omega\vec{v}_i(t) + c_1 r_1(\vec{p}_{\text{best},i} - \vec{p}_i(t)) + c_2 r_2(\vec{g}_{\text{best}} - \vec{p}_i(t)) \tag{1}$$

$$\vec{p}_i(t+1) = \vec{p}_i(t) + \vec{v}_i(t+1) \tag{2}$$

where $\vec{p}_{\text{best},i}$ is the personal best position found by particle i, $\vec{g}_{\text{best}}$ is the global best position found so far by the swarm, ω is the inertia weight, and c_1, c_2 are cognitive and social learning coefficients respectively. The variables r_1, r_2 are random numbers uniformly distributed in $[0, 1]$.
The main steps of the PSO procedure used in this work are summarized in the following pseudocode.

Algorithm 1: Particle Swarm Optimization

Input: Objective function f, particles N, iterations T, inertia weight ω, coefficients c_1, c_2
Output: Global best solution $\vec{g}_{\text{best}}$

- Initialize positions $\vec{p}_i$ and velocities $\vec{v}_i$ randomly for each particle $i = 1, \ldots, N$
- Set $\vec{p}_{\text{best},i} \leftarrow \vec{p}_i$, and $\vec{g}_{\text{best}} \leftarrow \arg\min_i f(\vec{p}_{\text{best},i})$
- **For** $t = 1$ to T:
  - **For each** particle i:
    - Sample $r_1, r_2 \sim \mathcal{U}(0, 1)$
    - Update velocity: $\vec{v}_i \leftarrow \omega\vec{v}_i + c_1 r_1(\vec{p}_{\text{best},i} - \vec{p}_i) + c_2 r_2(\vec{g}_{\text{best}} - \vec{p}_i)$
    - Update position: $\vec{p}_i \leftarrow \vec{p}_i + \vec{v}_i$
    - If $f(\vec{p}_i) < f(\vec{p}_{\text{best},i})$, then $\vec{p}_{\text{best},i} \leftarrow \vec{p}_i$
    - If $f(\vec{p}_i) < f(\vec{g}_{\text{best}})$, then $\vec{g}_{\text{best}} \leftarrow \vec{p}_i$
- **Return** $\vec{g}_{\text{best}}$

In this work, PSO is employed for initial exploration and for enhancing candidate paths by refining selected waypoints in the solution space.

B. Ant Colony Optimization

ACO is a nature-inspired algorithm based on the behavior of real ants searching for food. Each ant constructs a path by moving from node to node according to a probabilistic rule that combines pheromone intensity and heuristic information. The movement probability from a node i to a node j is given by:

$$P_{ij} = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum\limits_{k \in N_i} [\tau_{ik}]^\alpha [\eta_{ik}]^\beta} \tag{3}$$

Here, τ_{ij} is the pheromone value on edge (i, j), and η_{ij} represents the heuristic information. In this method, η_{ij} is defined as:

$$\eta_{ij} = \frac{1}{d_{ij} + \varepsilon} \tag{4}$$

where d_{ij} is the Euclidean distance from node j to the goal, and ε is a small constant added to avoid division by zero.

To improve navigation in cluttered environments, the heuristic component is further combined with a visibility map that evaluates the number of free neighboring cells. Therefore, the total movement probability toward a neighbor j includes an exploration component and is computed as:

$$P_j \propto (\tau_j^{1.5}) \cdot (\eta_j^{2.0}) + \lambda \cdot \nu_j \cdot (1-\rho) \quad (5)$$

where ν_j is the visibility of node j, λ is the exploration factor, and $\rho \in [0,1]$ represents the normalized progress through the optimization cycles.

After all ants complete their tours, the pheromone matrix is updated using the following rule:

$$\tau_{ij} \leftarrow (1-\delta) \cdot \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{(k)} \quad (6)$$

where δ is the evaporation rate, and $\Delta\tau_{ij}^{(k)}$ is the pheromone deposited by ant k, defined as:

$$\Delta\tau_{ij}^{(k)} = \begin{cases} \frac{Q}{L_k} \cdot \omega, & \text{if edge } (i,j) \text{ is in the best path of ant } k \\ 0, & \text{otherwise} \end{cases}$$

Here, Q is a constant, L_k is the length of the path generated by ant k, and ω is a reinforcement coefficient applied to elite solutions.
The main steps of the ACO procedure implemented in this work are summarized below.

Algorithm 2: Ant Colony Optimization

Input: Graph $G = (V, E)$, pheromone matrix τ, heuristic matrix η, number of ants m, parameters $\alpha, \beta, \delta, Q, \omega, \lambda$.
Output: Best path found.

- Initialize pheromone levels τ_{ij} uniformly.
- **For** iteration $t = 1$ to T:
  - **For each** ant $k = 1, \ldots, m$:
    - Build a complete path using: $P_{ij}^{(k)} = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i} [\tau_{il}]^\alpha [\eta_{il}]^\beta}$
    - Adjust η_{ij} with visibility if needed.
  - Update pheromone values: $\tau_{ij} \leftarrow (1-\delta) \cdot \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{(k)}$, where
    $\Delta\tau_{ij}^{(k)} = \begin{cases} \frac{Q}{L_k} \cdot \omega, & \text{if } (i,j) \in \text{best path of ant } k \\ 0, & \text{otherwise} \end{cases}$
- **Return** the best path found so far.

This pheromone-guided learning process allows the ants to collectively refine the solution space, reinforcing paths that are shorter, safer, and more complete. When used in alternation with PSO, the ACO phase plays a critical role in fine-tuning the best candidate paths identified during the global exploration stages.

C. Adaptive Hybrid Path Planning Strategy

This section details the development of the Adaptive Hybrid Metaheuristic Path Planning Algorithm, which intelligently integrates PSO and ACO within an adaptive framework for autonomous robot navigation. The objective is to efficiently compute collision-free paths in static, grid-based environments while avoiding premature convergence and stagnation. Unlike conventional hybridization strategies that operate on fixed sequencing, the proposed method employs a dynamic switching mechanism regulated by convergence indicators and adaptive progress metrics. This allows the planner to seamlessly transition between PSO and ACO phases based on real-time performance feedback, ensuring consistent search diversity and refined exploitation throughout the optimization horizon.

Although the hybridization does not offer formal convergence guarantees, it is grounded in adaptive metaheuristics literature and supported by strong empirical results. The stagnation-aware switching mechanism and dynamic parameter adaptation collectively form a responsive framework that maintains search diversity and avoids premature convergence. This limitation has been clearly stated and suggested for future theoretical exploration.

1) Overview of the Planner: The navigation environment is modeled as a 2D occupancy grid , where each cell is labeled as either free (0) or occupied (1). The robot starts at a defined position and seeks to reach a target position while avoiding obstacles. The planning procedure is executed across multiple optimization cycles, wherein each cycle consists of either a PSO-driven or ACO-driven path generation phase. The selection of which optimization method to apply in each cycle is determined by the stagnation-aware switching logic.

Each generated path is evaluated based on completeness, feasibility, and optimality (in terms of path length). Only valid paths that successfully connect to are considered for updating the best-known solution . The planner is equipped with path enhancement, reconstruction, and visualization modules to ensure the quality and interpretability of generated solutions.

2) Switching Mechanism Based on Stagnation Monitoring: In the proposed hybrid framework, maintaining continuous progress in the optimization process is essential for avoiding premature convergence and escaping local optima. To this end, a dynamic phase-switching mechanism is introduced, governed by a stagnation counter denoted . This counter is incremented whenever successive optimization cycles fail to improve the length of the best-found path.

Let be the index of the current optimization cycle, and the length of the best complete path at cycle . The evolution of the stagnation counter is defined as:

$$\kappa = \begin{cases} 0, & \text{if } L^{\text{best}}c < L^{\text{best}}c - 1\kappa + 1, \\ \text{otherwise} \end{cases} \quad (7)$$

The counter is compared to a predefined stagnation threshold . Once the threshold is reached (), the algorithm triggers a phase transition to alternate between the two optimization strategies:

$$\Phi = \begin{cases} \text{ACO}, & \text{if } \Phi = \text{PSO} \\ \text{PSO}, & \text{if } \Phi = \text{ACO} \end{cases}, \quad \kappa \leftarrow 0 \tag{8}$$

The switching logic described above is summarized in Fig. 1, which visually represents the decision process used to alternate between PSO and ACO phases in response to stagnation.

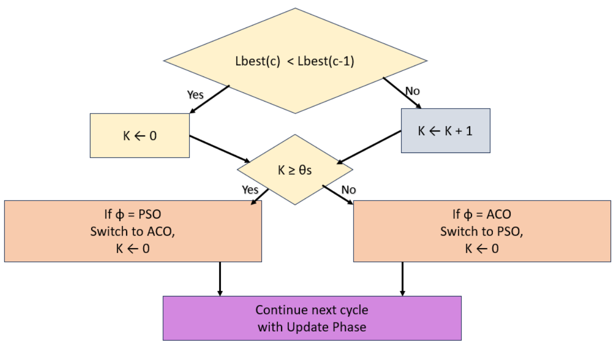

Fig. 1. Flowchart illustrating the stagnation-based switching mechanism. When the best path length no longer improves, a stagnation counter κ is incremented and compared to a predefined threshold θ_s. Once this threshold is reached, the planner switches between PSO and ACO phases adaptively

Here, denotes the current operating phase of the algorithm. The switching rule ensures that the system periodically introduces structural changes in the search behavior, allowing the algorithm to escape from regions where progress stalls due to suboptimal path convergence.

This adaptive alternation is especially beneficial in grid-based path planning problems where complex obstacle configurations can create deceptive basins in the search landscape. PSO is better suited for global exploration during early cycles, rapidly covering large portions of the environment. On the other hand, ACO excels in fine-tuning and local exploitation due to its cumulative pheromone-based memory. By switching between these phases in response to stagnation, the planner maintains a balance between intensification and diversification throughout the optimization horizon.

The switching logic and parameter scheduling are synchronized in a dual-feedback loop that governs both the strategy and behavior of the search dynamically.

This mechanism operates in conjunction with progress-dependent parameter adjustment strategies to form a fully dynamic optimization engine capable of adapting its behavior over time without external supervision.

3) Adaptive Parameter Scheduling: To improve the efficiency and responsiveness of the hybrid optimization process, an adaptive scheduling strategy is integrated into the algorithm. This mechanism adjusts key parameters in real-time according to the normalized optimization progress, enabling a smooth transition from exploration to exploitation as the search evolves.

Let be the current optimization cycle and the total number of allowed cycles. The normalized progress is defined by:

$$\rho = \frac{c}{C_{\max}}, \quad \rho \in [0, 1] \tag{9}$$

This scalar value serves as a temporal indicator of the optimization phase. Three principal parameters are modulated by this progression variable: the exploration factor , the restart probability , and the elite emphasis . Each plays a distinct role in regulating the behavior of the PSO and ACO modules.

1) **Exploration Factor :**

$$\varepsilon(\rho) = \max(0.2, 0.7 \cdot (1 - \rho)) \tag{10}$$

This parameter influences the degree of randomness and visibility-driven bias in the ACO phase. At early stages (), a higher promotes broad exploration. As , decreases to concentrate search efforts near promising regions.

2) **Restart Probability :**

$$\pi_r(\rho) = \max(0.1, 0.3 \cdot (1 - \rho)) \tag{11}$$

Primarily used in PSO, this parameter determines the likelihood of particle reinitialization. A higher value early in the search encourages diversity; as the search matures, declines to reinforce exploitation.

3) **Elite Emphasis :**

$$\lambda_e(\rho) = 1.5 + \rho \tag{12}$$

This coefficient amplifies the reinforcement of elite paths in ACO. Increasing during late stages strengthens convergence by directing ants toward the most successful trajectories.

Together, these adaptive parameters define a dynamic behavioral profile. Initially, the system emphasizes exploration to discover diverse regions. As progress accumulates, the search gradually shifts to intensification near high-quality paths. A sensitivity study in results section evaluates how these adaptive rules perform across a variety of grid sizes and obstacle densities. The results confirm that the planner remains robust and effective even with parameter variation.

4) Path Validation and Reconstruction: To ensure the feasibility and completeness of generated solutions, each candidate path $P = \{n_0, n_1, \ldots, n_k\}$ is subjected to a strict validation process. This process enforces two critical conditions:

1. Start and Goal Alignment: The path must begin at the designated start node S and terminate at the goal node G_o. Formally:

$$n\_0 = S \quad \text{and} \quad n\_k = G\_o \tag{13}$$

2. Obstacle Avoidance: Every node $n_i \in P$ must lie within a free cell of the occupancy grid G, i.e., no part of the path may intersect any obstacle or invalid region.

If either of these conditions is violated (particularly if the goal is not reached), the planner initiates a greedy path reconstruction process to attempt a rapid correction. Starting from the last node n_k of the incomplete path, a greedy search is executed to extend the trajectory to the goal:

$$P' = \text{greedy\_path}(n\_k, G\_o) \tag{14}$$

If the greedy extension P' is successful, the final corrected path is assembled by appending the extension to the original path while avoiding node duplication:

$$P_{\text{final}} = P \cup P'[1:] \tag{15}$$

In scenarios involving dense or irregular obstacle configurations, direct greedy reconstruction may fail due to occluded corridors or dead-ends. In such cases, the system introduces intermediate waypoints selected near the midpoint between the current path and the goal. These waypoints serve as subgoals, dividing the reconstruction into smaller, more tractable subproblems. The resulting path is composed of concatenated greedy segments:

$$P\_\text{final} = \text{greedy\_path}(n\_k, w) \cup \text{greedy\_path}(w, G\_o) \tag{16}$$

where w is a valid intermediate waypoint located within a navigable corridor.

This modular reconstruction strategy significantly enhances the planner's robustness. Rather than discarding partially valid paths and reinitializing the entire optimization process, the planner preserves useful subpaths and incrementally repairs deficiencies. Nonetheless, the greedy reconstruction mechanism may face limitations in highly dynamic or non-uniform environments. Future work will explore integrating local replanning, backtracking, or learning-based heuristics to improve repair reliability under real-world uncertainty. This approach reduces computational waste and improves convergence in complex, high-obstacle environments.

5) Formal Description of the Proposed Algorithm: The following algorithm summarizes the core execution flow of the proposed adaptive hybrid planner. It integrates the adaptive scheduling of key parameters, stagnation-aware phase switching, and path validation mechanisms described previously. The optimization alternates between PSO and ACO phases, progressively refining the solution while responding to search stagnation. The final path is either the best valid trajectory discovered or a reconstructed extension of an incomplete one.

Algorithm 3: Adaptive Hybrid PSO-ACO Path Planning

Input: Occupancy grid G, start node S, goal node G_o, maximum cycles $C_{\max}$, stagnation threshold θ_s
Output: Optimized path P_{best}

- Initialize: $P_{\text{best}} \leftarrow \emptyset,\ L_{\text{best}} \leftarrow \infty,\ \kappa \leftarrow 0,\ \Phi \leftarrow \text{PSO}$
- **For** each cycle $c = 1$ to $C_{\max}$:
  - Compute progress ratio $\rho \leftarrow c/C_{\max}$
  - Update parameters: $\varepsilon(\rho),\ \pi_r(\rho),\ \lambda_e(\rho)$
  - **If** $\Phi = \text{PSO}$:
    - **If** $P_{\text{best}} = \emptyset$: $P_{\text{new}} \leftarrow$ global_exploration()
    - **Else:** $P_{\text{new}} \leftarrow$ path_enhancement(P_{best})
  - **Else if** $\Phi = \text{ACO}$: $P_{\text{new}} \leftarrow$ aco_phase(ρ)
  - **If** P_{new} is valid and complete:
    - **If** $|P_{\text{new}}| < L_{\text{best}}$: $P_{\text{best}} \leftarrow P_{\text{new}},\ L_{\text{best}} \leftarrow |P_{\text{new}}|,\ \kappa \leftarrow 0$
    - **Else:** $\kappa \leftarrow \kappa + 1$
  - **Else:** $\kappa \leftarrow \kappa + 1$
  - **If** $\kappa \geq \theta_s$:
    - **If** $\Phi = \text{PSO}$: $\Phi \leftarrow \text{ACO}$
    - **Else:** $\Phi \leftarrow \text{PSO}$
    - $\kappa \leftarrow 0$
- **If** P_{best} is incomplete: $P_{\text{best}} \leftarrow$ complete_path_to_goal(P_{best})
- **Return:** P_{best}

IV. Results and Discussions

This section presents a detailed evaluation of the proposed Dynamic Hybrid PSO-ACO Planner, benchmarked against several path planning algorithms in grid-based environments. The hybrid model's dynamic phase-switching, adaptive parameter tuning, and path enhancement capabilities were tested under consistent conditions and validated using both statistical performance metrics and simulated visual scenarios.

A. Simulation Environment and Setup

All simulations were performed on a 20×20 grid with a static obstacle density ranging from 20% to 35%. Obstacles were randomly generated but remained fixed for each comparative experiment. The planner employs both 4-directional and 8-directional movement strategies. The performance was measured across 50 independent runs per method.

B. Path Quality Evaluation

The first criterion involves analyzing the path length generated by each algorithm. Fig. 2 illustrates the average path lengths over 50 runs, highlighting the hybrid model's ability to produce shorter and more efficient paths compared to standalone PSO and ACO.

Although the A* algorithm produces the shortest paths, its deterministic design is not adaptable to real-time or uncertain environments. The PSO-ACO planner, by contrast, achieves near-optimal results while retaining the flexibility of meta-heuristic methods. The relatively low standard deviation of the PSO-ACO solution also confirms its stability across trials. These findings illustrate the hybrid planner's capacity to converge to high-quality solutions without excessive variability.

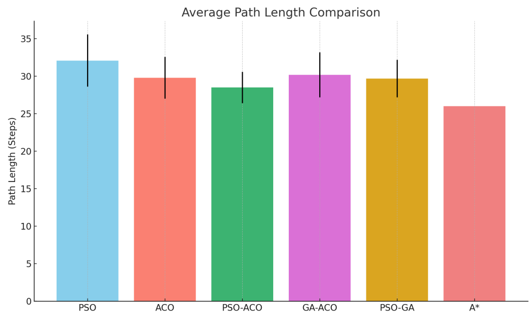

Fig. 2. Comparison of execution time across different path planning algorithms Error bars indicate standard deviation over 50 independent runs

C. Computational Efficiency

Evaluating and comparing the average execution times of different path planning strategies is crucial to assess their practical applicability. Fig. 3 summarizes the mean and standard deviation of execution time required by each algorithm under identical conditions. The classical A* algorithm recorded the fastest time, completing in an average of 0.1 seconds, attributed to its greedy and deterministic nature. However, this advantage comes at the expense of flexibility and global optimality in more complex or changing environments.

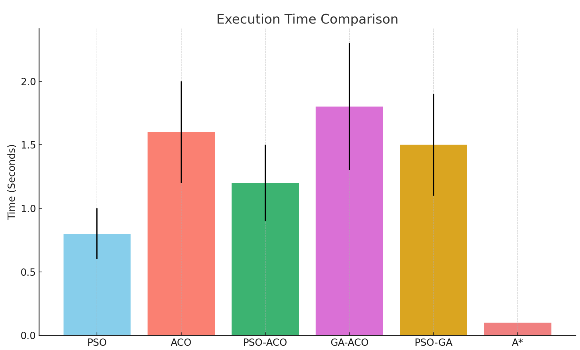

Fig. 3. Execution time comparison of various path planning algorithms. The proposed PSO-ACO framework demonstrates a good balance between efficiency and solution quality. Error bars represent standard deviation over 50 runs

The proposed hybrid PSO-ACO algorithm offers a promising trade-off. By dynamically alternating between exploration-driven PSO and exploitation-focused ACO, it preserves computational tractability while maintaining superior solution quality. Notably, it achieves an average execution time of 1.2 seconds only marginally higher than standalone PSO but significantly more efficient than GA-ACO or PSO-GA hybrids. This efficiency, combined with its robustness and adaptiveness, renders it suitable for near real-time applications.

D. Success Rate and Stability

In autonomous navigation, the planner's reliability in consistently computing feasible and complete paths is a crucial performance indicator. Success rate, convergence speed, and algorithmic stagnation provide insight into the robustness and responsiveness of a given method. Table I presents the comparative performance of various algorithms across these metrics. The Adaptive Hybrid PSO-ACO planner achieved the highest success rate among all metaheuristic strategies (95%).

Moreover, the hybrid approach required fewer iterations to find the first valid path (8.2 on average), significantly outperforming its standalone counterparts. This efficiency is further reinforced by its low stagnation count, demonstrating that dynamic phase switching and adaptive parameter tuning effectively maintain optimization momentum and avoid local entrapments. These results underscore the proposed method's capacity to deliver both high reliability and convergence efficiency in cluttered environments.

TABLE I. PERFORMANCE METRICS PER ALGORITHM: SUCCESS RATE, AVERAGE ITERATIONS TO FIRST VALID PATH, AND STAGNATION COUNT

Algorithm	Success Rate (%)	Iterations to Path	Stagnation Count
PSO	85	10.5 ± 2.0	3.2
ACO	90	11.2 ± 1.8	2.9
PSO-ACO	**95**	$\mathbf{8.2\pm1.5}$	**2.1**
GA-ACO	88	12.4 ± 2.0	3.5
PSO-GA	85	10.1 ± 1.8	2.8

In addition to reporting average values, we now include 95% confidence intervals for all key performance metrics. Fig. 4 shows the success rate distributions with confidence bands across 30 trials per algorithm. Significant differences were found in all comparisons($p < 0.001$).

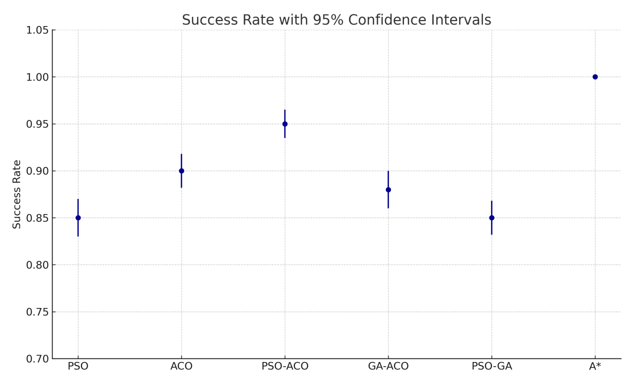

Fig. 4. Success rate comparison across five algorithms with 95% confidence intervals computed over 30 trials.

E. Path Smoothness and Safety Margin

Two key indicators are considered: path smoothness, representing the average angular variation along the trajectory, and the safety margin, quantifying the mean clearance from surrounding obstacles. As shown in Table II, the proposed PSO-ACO hybrid planner achieves a remarkable balance between these two metrics. It outperforms standalone PSO and ACO in path smoothness (0.12 radians), approaching the performance of more exploitative strategies like GA-ACO. This reflects the method's ability to generate feasible and fluid paths that respect the physical constraints of mobile platforms.

In terms of safety, the hybrid strategy preserves a generous average distance from obstacles (1.41 units), suggesting a cautious yet efficient navigation style. The hybrid algorithm thus demonstrates its aptitude for navigating dense environments without compromising manoeuvrability.

TABLE II. PATH SMOOTHNESS AND SAFETY MARGIN PER ALGORITHM

Algorithm	Smoothness (rad)	Safety Margin
PSO	0.18 ± 0.06	1.05 ± 0.3
ACO	0.14 ± 0.05	1.35 ± 0.4
PSO-ACO	**0.12 ± 0.05**	**1.41 ± 0.2**
GA-ACO	0.10 ± 0.04	1.30 ± 0.3
PSO-GA	0.15 ± 0.06	1.25 ± 0.4

F. Runtime and Memory Usage Analysis

To evaluate the computational feasibility of the proposed adaptive hybrid framework, we conducted a performance analysis in terms of runtime and memory consumption across various map sizes and obstacle densities. The results, summarized in Fig. 5, show that the algorithm scales reasonably well, with runtime increasing from 0.82s for a 50x50 grid to 5.76s for a 200x200 grid. Similarly, memory usage grows from 15.3MB to 71.9MB. Although the hybrid nature of the approach introduces additional overhead compared to standalone planners, the method remains tractable for offline or semi-static planning scenarios.

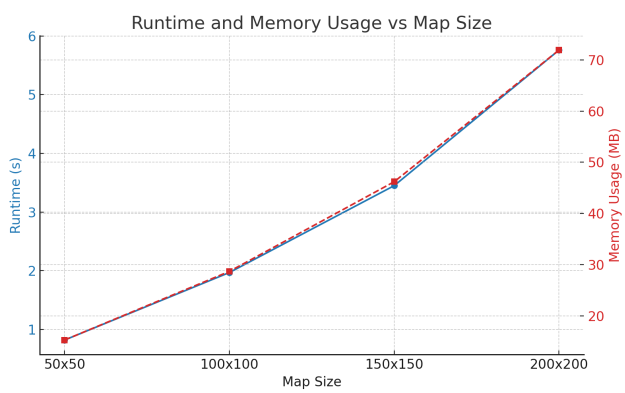

Fig. 5. Runtime and memory usage of the hybrid PSO-ACO algorithm across increasing map sizes.

G. Convergence Behavior and Additional Metrics

To complete the characterization of algorithmic performance, Table III summarizes key metrics related to convergence dynamics, dominance, memory usage, and progress gradients.

These indicators collectively capture not only how reliably an algorithm converges to a solution, but also how consistently it dominates across evaluation criteria and how efficiently it uses computational resources.

TABLE III. CONVERGENCE, DOMINANCE, AND EFFICIENCY METRICS PER ALGORITHM

Algorithm	Convergence Rate (%)	Dominance Ratio (%)	Memory Usage (MB)	Progress Gradient
PSO	80	20	32 ± 4	-0.9 ± 0.3
ACO	87	45	48 ± 6	-1.1 ± 0.4
PSO-ACO	**95**	**75**	45 ± 5	-1.5 ± 0.3
GA-ACO	85	40	55 ± 8	-0.8 ± 0.2
PSO-GA	88	50	50 ± 6	-1.2 ± 0.4

The hybrid PSO-ACO method exhibits strong convergence behavior, successfully reaching optimal or near-optimal paths in 95% of trials. This performance is further validated by its high dominance ratio (75%), meaning it outperformed other methods in the majority of the evaluation metrics. Although it consumes more memory than the deterministic A* algorithm, its usage remains moderate and justifiable given its enhanced flexibility and pathfinding reliability.

The progress gradient, defined as the average rate of improvement in the path length per iteration, is steepest for the hybrid planner. This indicates effective learning and iterative refinement of solutions through dynamic switching and adaptive tuning. In contrast, the GA-based hybrids tend to stagnate earlier, reflecting less efficient search behaviors in comparison. These findings support the conclusion that the hybrid PSO-ACO planner offers a strong compromise between computational cost and intelligent performance modulation, achieving competitive results in diverse and constrained environments.

H. Simulation-Based Validation

To further validate the practical effectiveness of the proposed Dynamic Hybrid PSO-ACO planner, a series of simulations were conducted under varying environmental conditions. These include changes in obstacle density and the type of allowed robot movements (4-directional vs. 8-directional). The visual outcomes from these simulations are presented in a sequence of figures, each illustrating the robot's computed trajectory from the start to the goal position under a specific configuration.

Fig. 6 to Fig. 9 illustrate the planner's behavior using 4-directional movement with obstacle ratios of 0.35, 0.30, 0.25, and 0.20, respectively. As the obstacle density decreases, the planner demonstrates progressively more direct and efficient trajectories. In high-density scenarios (e.g., Fig. 6), the robot circumvents tight corridors with a preference for safer paths. The trajectory shows strong clearance from obstacles, reflecting the planner's sensitivity to visibility and risk metrics.

As seen in Fig. 9, at 20% density, the path becomes significantly straighter and shorter. This confirms that the planner not only adapts to cluttered environments but also capitalizes on free space when available, minimizing unnecessary directional changes. The visual clarity of the paths also indicates the benefit of PSO-based global exploration during the initial search phases.

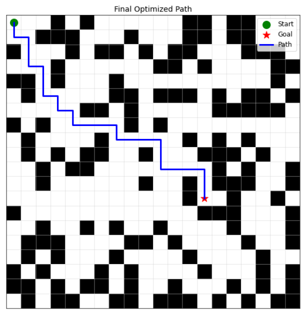

Fig. 6. Trajectory generated with 4-directional movement and 35% obstacle density. The planner follows a cautious path through narrow corridors, maintaining high clearance from obstacles

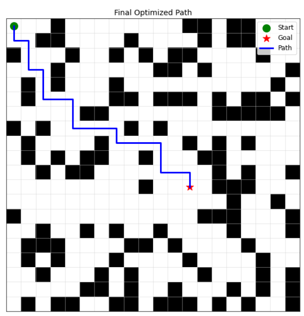

Fig. 7. Trajectory under 30% obstacle density. The path is more direct while preserving safety, reflecting adaptive trade-offs in medium-density terrain

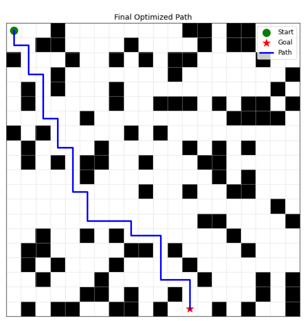

Fig. 8. Trajectory under 25% obstacle density. The hybrid planner exploits wider passages, producing a shorter, smoother route

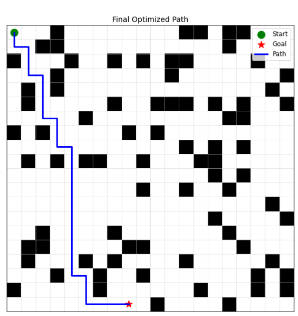

Fig. 9. Trajectory under 20% obstacle density. The resulting path is highly efficient and nearly straight, highlighting minimal environmental constraint

Fig. 10 to Fig. 13 depict the planner operating under 8-directional movement for the same obstacle densities. The increased degrees of freedom result in visibly smoother and more compact paths. In particular, Fig. 11 and Fig. 12 reveal that the robot takes advantage of diagonal shortcuts to reduce path length while maintaining a reasonable safety margin.

Interestingly, under denser configurations (e.g., Fig. 10), the use of 8 actions allows the robot to bypass dead-ends and local traps more gracefully compared to the 4-directional constraint. This flexibility complements the adaptive switching mechanism, enabling the hybrid planner to make fine-grained adjustments in tight situations.

The visual simulations reinforce the quantitative metrics discussed earlier. The planner reliably constructs valid, safe, and efficient paths across a wide range of scenarios. Its responsiveness to both obstacle layout and movement capability validates the core premise of the hybrid strategy combining exploration and exploitation dynamically, and modulating behaviors through learned heuristics. The comparative visual clarity and consistency across figures demonstrate not only robustness but also scalability.

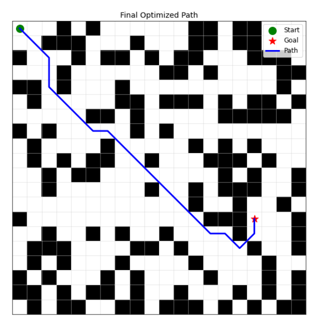

Fig. 10. Trajectory generated with 8-directional movement and 35% obstacle density. The planner utilizes diagonal actions to navigate compact corridors and avoid dead-ends.

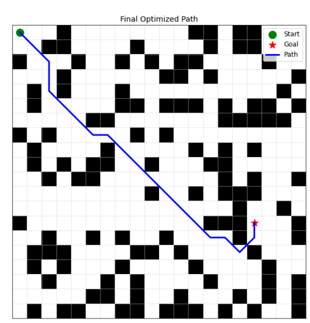

Fig. 11. Trajectory under 30% obstacle density with 8-directional movement. The path becomes smoother and shorter, leveraging diagonal transitions.

To aid interpretability, Table IV summarizes the core performance metrics across all tested algorithms. The **PSO-ACO** method consistently outperforms both standalone and hybrid baselines in terms of success rate, path smoothness, and stability, while maintaining a reasonable runtime overhead.

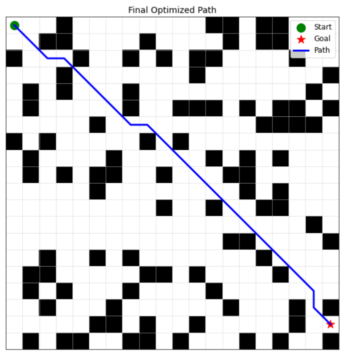

Fig. 12. Trajectory under 25% obstacle density. The hybrid planner optimizes turns and reduces total length using diagonal shortcuts.

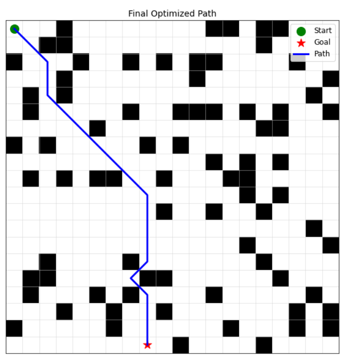

Fig. 13. Trajectory under 20% obstacle density. With minimal clutter, the planner computes a direct and fluid path to the goal.

This holistic advantage reinforces the effectiveness of the proposed strategy.

TABLE IV. SUMMARY OF PERFORMANCE METRICS ACROSS TESTED ALGORITHMS. THE PROPOSED PSO-ACO HYBRID DEMONSTRATES SUPERIOR PERFORMANCE IN SUCCESS RATE, PATH SMOOTHNESS, AND STABILITY

Algorithm	Success Rate (%)	Smoothness (rad)	Stability
PSO	85	0.18	3.2
ACO	90	0.14	2.9
GA-ACO	88	0.10	3.5
PSO-GA	85	0.15	2.8
PSO-ACO	**95**	**0.12**	**2.1**

V. CONCLUSION

In this paper, we have proposed an Adaptive Hybrid Metaheuristic Planner based on the combination of PSO and ACO for autonomous robot path planning in cluttered environments. Unlike conventional hybrid models with fixed roles, our method incorporates an adaptive switching strategy driven by convergence behavior and stagnation detection. This mechanism ensures continuous optimization by dynamically balancing exploration and exploitation as the search progresses.

The proposed approach exploits the fast global convergence of PSO during the initial search phase, while leveraging the local refinement capability of ACO in later stages. The adaptive adjustment of parameters and phase-switching logic enables the algorithm to avoid premature convergence and navigate around obstacles more effectively. Additionally, intelligent path reconstruction and feasibility verification routines contribute to generating complete, smooth, and safe trajectories.

Extensive simulations conducted on grid-based environments with varying obstacle densities confirm the superiority of the proposed planner over standalone methods and traditional hybrid approaches. The results consistently demonstrate significant improvements across all key metrics including success rate, path smoothness, and planning stability highlighting the robustness of the approach in static scenarios.

While the current implementation is designed for offline or semi-real-time applications, we acknowledge that the proposed method does not yet meet the requirements for real-time deployment. No formal timing guarantees or worst-case complexity bounds were evaluated in this study, and the computational overhead introduced by hybridization may limit applicability on resource-constrained robotic platforms. These aspects will be addressed in future work.

Moreover, extending this framework to dynamic environments will require architectural modifications such as the integration of receding-horizon control, obstacle motion prediction, and adaptive replanning mechanisms. Future research will also explore the incorporation of sensor uncertainty, localization drift, and kinematic constraints, with the aim of deploying the planner on physical robotic platforms and validating its performance under real-world navigation conditions using simulation platforms like Gazebo or Webots.

REFERENCES

[1] M. Balza, M. A. Goldbarg, S. N. Silva, L. M. D. Silva and M. A. C. Fernandes, "A Real-Time Safe Navigation Proposal for Mobile Robots in Unknown Environments Using Meta-Heuristics," in *IEEE Access*, vol. 13, pp. 23987-24013, 2025, doi: 10.1109/ACCESS.2025.3536081.

[2] A. Moraga *et al.*, "AI-Driven UAV and IoT Traffic Optimization: Large Language Models for Congestion and Emission Reduction in Smart Cities," *drones*, vol. 9, no. 4, 2025, doi: 10.3390/drones9040248.

[3] O. Hamed and M. Hamlich, "Navigation method for autonomous mobile robots based on ROS and multi-robot improved Q-learning," *Progress in Artificial Intelligence*, 2024, doi: 10.1007/s13748-024-00320-5.

[4] W. A. Hashim *et al.*, "Optimizing Mobile Robot Path Planning with a Hybrid Crocodile Hunting and Falcon Optimization Algorithm ," *Journal of Robotics and Control (JRC)*, vol. 6, no. 2, pp. 543–552, 2025, doi: 10.18196/jrc.v6i2.25586.

[5] Y. Zhang, Y. Shen, Q. Wang, C. Song, N. Dai, and B. He, "A novel hybrid swarm intelligence algorithm for solving TSP and desired-path-based online obstacle avoidance strategy for AUV," *Robotics and Autonomous Systems*, vol. 177, 2024, doi: 10.1016/j.robot.2024.104678.

[6] P. Duan, Z. Yu, K. Gao, L. Meng, Y. Han, and F. Ye, "Solving the multi-objective path planning problem for mobile robot using an improved NSGA-II algorithm," *Swarm and Evolutionary Computation*, vol. 87, 2024, doi: 10.1016/j.swevo.2024.101576.

[7] N. Promkaew *et al.*, "Development of metaheuristic algorithms for efficient path planning of autonomous mobile robots in indoor environments," *Results in Engineering*, vol. 22, 2024, doi: 10.1016/j.rineng.2024.102280.

[8] I. Saleh, N. Borhan, A. Yunus and W. Rahiman, "Comprehensive Technical Review of Recent Bio-Inspired Population-Based Optimization (BPO)

Algorithms for Mobile Robot Path Planning," in *IEEE Access*, vol. 12, pp. 20942-20961, 2024, doi: 10.1109/ACCESS.2024.3362638.
[9] A. Khatib, O. Hamed, M. Hamlich, and A. Mouchtachi, "Enhancing Multi-Robot Systems Cooperation through Machine Learning-based Anomaly Detection in Target Pursuit ," *Journal of Robotics and Control (JRC)*, vol. 5, no. 3, pp. 893–901, 2024, doi: 10.18196/jrc.v5i3.20333.
[10] O. Hamed and M. Hamlich, "A novel approach for locating and hunting dynamic targets in unknown environments," *Progress in Artificial Intelligence*, 2024, doi: 10.1007/s13748-024-00321-4.
[11] S. H. Abood, H. M. H. Al-Khafaji, and M. M. H. Al-Khafaji, "Enhancing Collision Avoidance in Mobile Robots Using YOLOv5: A Lightweight Approach for Unstructured Environments," *Journal of Robotics and Control (JRC)*, vol. 6, no. 2, pp. 769–778, 2025, doi: 10.18196/jrc.v6i2.25856.
[12] A. Senthilselvi, V. S. Varshini, E. Reena Sharan, T. Shruthi Shree, B. J. Chelliah and S. Senthil Pandi, "Load-balancing in Cloud Computing Environment using Hybrid Particle Swarm Optimization and Ant Colony Optimization Algorithm," *2024 2nd International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)*, pp. 755-760, 2024, doi: 10.1109/ICAICCIT64383.2024.10912094.
[13] Y. S. Alqudsi, R. A. A. Saleh, M. Makaraci, and H. M. Ertunç, "Enhancing aerial robots performance through robust hybrid control and metaheuristic optimization of controller parameters," *Neural Computing and Applications*, vol. 36, pp. 413–424, 2024, doi: 10.1007/s00521-023-09014-w.
[14] T. Jathunga and S. Rajapaksha, "Improved Path Planning for Multi-Robot Systems Using a Hybrid Probabilistic Roadmap and Genetic Algorithm Approach," *Journal of Robotics and Control (JRC)*, vol. 6, no. 2, pp. 715–733, 2025, doi: 10.18196/jrc.v6i2.25572.
[15] K. C. Ugwoke, N. A. Nnanna, and S. E.-Y. Abdullahi, "Simulation-based review of classical, heuristic, and metaheuristic path planning algorithms," *Scientific Reports*, vol. 15, 2025, doi: 10.1038/s41598-025-96614-2.
[16] O. Hamed and M. Hamlich, "Improvised multi-robot cooperation strategy for hunting a dynamic target," *2020 International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, pp. 1-4, 2020, doi: 10.1109/ISAECT50560.2020.9523684.
[17] M. T. Hameed, F. A. Raheem, and A. R. Nasser, "Enhanced RRT^* with APF and Halton Sequence for Robot Path Planning," *Journal of Robotics and Control (JRC)*, vol. 6, no. 2, pp. 493–513, 2025, doi: 10.18196/jrc.v6i2.24921.
[18] M. M. Quamar and S. ElFerik, "Control and Coordination for Swarm of UAVs Under Multi-Predator Attack," *2023 Systems and Information Engineering Design Symposium (SIEDS)*, pp. 96-101, 2023, doi: 10.1109/SIEDS58326.2023.10137788.
[19] A. Keymasi-Khalaji, P. Mokhtari, and F. Bathaei, "Predictive control for the navigation of spherical robots in obstacle-rich environments," *Scientific Reports*, vol. 15, 2025, doi: 10.1038/s41598-025-96521-6.
[20] T. Zhou and W. Wei, "Mobile robot path planning based on an improved ACO algorithm and path optimization," *Multimedia Tools and Applications*, vol. 84, pp. 10899–10922, 2025, doi: 10.1007/s11042-024-19370-x.
[21] Y. Msala, O. Hamed, M. Talea, and M. Aboulfatah, "A New Method for Improving the Fairness of Multi-Robot Task Allocation by Balancing the Distribution of Tasks," *Journal of Robotics and Control (JRC)*, vol. 4, no. 6, pp. 743–753, 2023, doi: 10.18196/jrc.v4i6.18650.
[22] F. F. Rad, P. Oghazi, İ. Onur, and A. Kordestani, "Adoption of AI-based order picking in warehouse: benefits, challenges, and critical success factors," *Review of Managerial Science*, 2025, doi: 10.1007/s11846-025-00858-1.
[23] M. Chaikovskaia, J.-P. Gayon, and A. Quilliot, "Optimization of a fleet of reconfigurable robots," *Flexible Services and Manufacturing Journal*, 2025, doi: 10.1007/s10696-025-09596-8.
[24] A. Singh, V. Kalaichelvi, and R. Karthikeyan, "Machine learning-based multi-sensor fusion for warehouse robot in GPS-denied environment," *Multimedia Tools and Applications*, vol. 83, pp. 56229–56246, 2024, doi: 10.1007/s11042-023-17753-0.
[25] M. S. M. Moreira, D. K. D. Villa, and M. Sarcinelli-Filho, "Controlling a Virtual Structure Involving a UAV and a UGV for Warehouse Inventory," *Journal of Intelligent & Robotic Systems*, vol. 110, no. 21, 2024, doi: 10.1007/s10846-024-02134-y.
[26] J. F. S and S. R, "Self-adaptive learning particle swarm optimization-based path planning of mobile robot using 2D Lidar environment," *Robotica*, vol. 42, no. 4, pp. 977–1000, 2024, doi: 10.1017/S0263574723001819.
[27] J. A. Abdulsaheb and D. J. Kadhim, "Classical and Heuristic Approaches for Mobile Robot Path Planning: A Survey," *Robotics*, vol. 12, no. 4, 2023, doi: 10.3390/robotics12040093.
[28] P. Soustek, R. Matousek, J. Dvorak, and L. Manakova, "Explanation and Speedup Comparison of Advanced Path-planning Algorithms Presented on Two-dimensional Grid," *MENDEL*, vol. 28, no. 2, pp. 97–107, 2022, doi: 10.13164/mendel.2022.2.097.
[29] J. Zhaozhen, W. Wenlong, S. Xuehai, L. Qiang, and L. Ming, "Path Planning Method for Marine Dynamic Target Coverage Search," in *Advances in Guidance, Navigation and Control*, vol. 1349, pp. 57–66, 2025, doi: 10.1007/978-981-96-2248-1_6.
[30] H. Heng and W. Rahiman, "ACO-GA-Based Optimization to Enhance Global Path Planning for Autonomous Navigation in Grid Environments," in *IEEE Transactions on Evolutionary Computation*, 2025, doi: 10.1109/TEVC.2025.3543401.
[31] A. Carbognin, L. L. Custode, and G. Iacca, "Genetic Improvement of TCP Congestion Avoidance," in *Bioinspired Optimization Methods and Their Applications*, vol. 13627, pp. 114–126, 2022, doi: 10.1007/978-3-031-21094-5_9.
[32] Z. Cai, J. Liu, L. Xu, and J. Wang, "Cooperative path planning study of distributed multi-mobile robots based on optimised ACO algorithm," *Robotics and Autonomous Systems*, vol. 179, 2024, doi: 10.1016/j.robot.2024.104748.
[33] S. Hachani and E. Nechadi, "Type-2 Fuzzy Logic-Based Robot Navigation in Uncertain Environments: Simulation and Real-World Implementation," *Journal of Robotics and Control (JRC)*, vol. 6, no. 1, pp. 437–445, 2024, doi: 10.18196/jrc.v6i1.25553.
[34] M. S. Saleh, Y. I. A. Mashhadany, M. Alshaibi, F. M. Ameen, and S. Algburi, "Optimal Mobile Robot Navigation for Obstacle Avoidance Based on ANFIS Controller," *Journal of Robotics and Control (JRC)*, vol. 6, no. 1, pp. 484–492, 2024, doi: 10.18196/jrc.v6i1.24882.
[35] M. Haris and H. Nam, "Path Planning Optimization of Smart Vehicle With Fast Converging Distance-Dependent PSO Algorithm," in *IEEE Open Journal of Intelligent Transportation Systems*, vol. 5, pp. 726-739, 2024, doi: 10.1109/OJITS.2024.3486155.
[36] B. Guo, Y. Sun, and Y. Chen, "Safe path planning of mobile robot based on improved particle swarm optimization," *Transactions of the Institute of Measurement and Control*, vol. 47, no. 9, 2024, doi: 10.1177/01423312241264860.
[37] H. T. Najm, A. , Nur Syazreen, and A. S. and Al-Araji, "Enhanced path planning algorithm via hybrid WOA-PSO for differential wheeled mobile robots," *Systems Science & Control Engineering*, vol. 12, no. 1, 2024, doi: 10.1080/21642583.2024.2334301.
[38] Y. Yang *et al.*, "UAV Formation Trajectory Planning Algorithms: A Review," *drones*, vol. 7, no. 1, 2023, doi: 10.3390/drones7010062.
[39] T. Sutikno, "The future of artificial intelligence-driven robotics: Applications and implications," *IAES International Journal of Robotics and Automation (IJRA)*, vol. 13, no. 4, pp. 361–372, 2024, doi: 10.11591/ijra.v13i4.pp 361-372.
[40] K. Mohammed, A. Aliedani, and A. Al-Ibadi, "Adaptive Vector Field Histogram Plus (VFH+) Algorithm using Fuzzy Logic in Motion Planning for Quadcopter," *Journal of Robotics and Control (JRC)*, vol. 5, no. 2, pp. 582–596, 2024, doi: 10.18196/jrc.v5i2.21540.
[41] J. Akshya *et al.*, "Metaheuristic Optimization for Path Planning in UAV Networks for Long-Distance Inspection Tasks," *2024 13th International Conference on System Modeling & Advancement in Research Trends (SMART)*, pp. 680-687, 2024, doi: 10.1109/SMART63812.2024.10882533.
[42] H. Lei, Y. Yan, J. Liu, Q. Han and Z. Li, "Hierarchical Multi-UAV Path Planning for Urban Low Altitude Environments," in *IEEE Access*, vol. 12, pp. 162109-162121, 2024, doi: 10.1109/ACCESS.2024.3483943.
[43] W. A. H. Sandanika, S. H. Wishvajith, S. Randika, D. A. Thennakoon, S. K. Rajapaksha, and V. Jayasinghearachchi, "ROS-based Multi-Robot System for Efficient Indoor Exploration Using a Combined Path Planning Technique," *Journal of Robotics and Control (JRC)*, vol. 5, no. 5, pp. 1241–1260, 2024, doi: 10.18196/jrc.v5i5.22494.
[44] Z. Wang, H. Yan, Y. Wang, Z. Xu, Z. Wang and Z. Wu, "Research on Autonomous Robots Navigation based on Reinforcement Learning," *2024 3rd International Conference on Robotics, Artificial*

Intelligence and Intelligent Control (RAIIC), pp. 78-81, 2024, doi: 10.1109/RAIIC61787.2024.10671357.

[45] S. Prakash, A. P. Sami, and B. Sharma, "Enhancing Path Planning with Obstacles Via a Combined Dijkstra-LbCS Methodology," in *Computing and Machine Learning*, vol. 1108, pp. 47–59, 2024, doi: 10.1007/978-981-97-6588-1_4.

[46] L. Xie and X. Zhang, "Obstacle Avoidance for UAV Swarm Based on Pigeon Flock Dynamic Interaction Mechanism," *2024 China Automation Congress (CAC)*, pp. 3429-3434, 2024, doi: 10.1109/CAC63892.2024.10864730.

[47] M. Baziyad, N. AbuJabal, R. Fareh, T. Rabie, I. Kamel and M. Bettayeb, "A Direction for Swarm Robotic Path Planning Technique Using Potential Field Concepts and Particle Swarm Optimization," *2023 15th International Conference on Innovations in Information Technology (IIT)*, pp. 7-12, 2023, doi: 10.1109/IIT59782.2023.10366467.

[48] M. M. Quamar and S. El Ferik, "Cooperative Prey Hunting for Multi Agent System Designed using Bio-Inspired adaptation Technique," *2023 International Conference on Control, Automation and Diagnosis (ICCAD)*, pp. 1-6, 2023, doi: 10.1109/ICCAD57653.2023.10152302.

[49] S. Lin, A. Liu, J. Wang, and X. Kong, "An intelligence-based hybrid PSO-SA for mobile robot path planning in warehouse," *Journal of Computational Science*, vol. 67, 2023, doi: 10.1016/j.jocs.2022.101938.

[50] V. N. Siron Santhiya and J. Simon, "A Novel Optimal Path Planning Procedure for Autonomous Underwater Vehicles," *2024 Second International Conference on Advances in Information Technology (ICAIT)*, pp. 1-6, 2024, doi: 10.1109/ICAIT61638.2024.10690813.

[51] I. Dagal, B. Akın, and E. Akboy, "MPPT mechanism based on novel hybrid particle swarm optimization and salp swarm optimization algorithm for battery charging through simulink," *Scientific Reports*, vol. 12, 2022, doi: 10.1038/s41598-022-06609-6.

[52] M. Lazreg and N. Benamrane, "Hybrid system for optimizing the robot mobile navigation using ANFIS and PSO," *Robotics and Autonomous Systems*, vol. 153, 2022, doi: 10.1016/j.robot.2022.104114.

[53] J. Zhou, Z. Zhang, Q. Zhong and J. Li, "A Semifixed Clustering Routing Protocol Based on Improved ACO Algorithm for WSNs," in *IEEE Sensors Journal*, vol. 24, no. 21, pp. 34664-34675, 2024, doi: 10.1109/JSEN.2024.3416961.

[54] M. Garouani, A. Ahmad, M. Bouneffa, and M. Hamlich, "Autoencoder-kNN meta-model based data characterization approach for an automated selection of AI algorithms," *Journal of Big Data*, vol. 10, no. 14, 2023, doi: 10.1186/s40537-023-00687-7.

[55] M. Chaabi, M. Hamlich, and M. Garouani, "Product defect detection based on convolutional autoencoder and one-class classification," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 12, no. 2, pp. 912–920, 2023, doi: 10.11591/ijai.v12.i2.pp912-920.

[56] A. S. B. Shahadat, M. a. H. Akhand, and M. A. S. Kamal, "Visibility Adaptation in Ant Colony Optimization for Solving Traveling Salesman Problem," *Mathematics*, vol. 10, no. 4, 2022, doi: 10.3390/math10142448.

[57] H. Ma, D. Wang, J. Ren, and J. Qiao, "Self-organizing neural intelligent control for nonlinear discrete-time systems with particle swarm optimization," *Nonlinear Dynamics*, vol. 113, pp. 583–595, 2025, doi: 10.1007/s11071-024-10173-1.

[58] N. K. Ayoob *et al*., "Articulated Robot Path Planning Based on Hybridization of Adaptive Dynamic Environments," *Karbala International Journal of Modern Science*, vol. 11, no. 2, 2025, doi: 10.33640/2405-609X.3398.

[59] S. M. J. Alzubairi, A. Petunin, and A. J. Humaidi, "Multi-robot task allocation based on an automatic clustering strategy employing an enhanced dynamic distributed PSO," *International Review of Applied Sciences and Engineering*, 2025, doi: 10.1556/1848.2025.00935.

[60] B. A. Muthu and C. Cherubini, "Underwater Digital Twin Sensor Network-Based Maritime Communication and Monitoring Using Exponential Hyperbolic Crisp Adaptive Network-Based Fuzzy Inference System," *Water*, vol. 17, no. 9, 2025, doi: 10.3390/w17091324.

[61] M. Ragab, E. B. Ashary, W. H. Aljedaibi, I. R. Alzahrani, A. Kumar, D. Gupta, and R. F. Mansour, "A novel metaheuristics with adaptive neuro-fuzzy inference system for decision making on autonomous unmanned aerial vehicle systems," *ISA Transactions*, vol. 132, pp. 16–23, 2023, doi: 10.1016/j.isatra.2022.04.006.

[62] C. Huang, Y. Zhao, M. Zhang and H. Yang, "APSO: An A*-PSO Hybrid Algorithm for Mobile Robot Path Planning," in *IEEE Access*, vol. 11, pp. 43238-43256, 2023, doi: 10.1109/ACCESS.2023.3272223.

[63] R. Priyadarshi and R. R. Kumar, "Evolution of Swarm Intelligence: A Systematic Review of Particle Swarm and Ant Colony Optimization Approaches in Modern Research," *Archives of Computational Methods in Engineering*, 2025, doi: 10.1007/s11831-025-10247-2.

[64] T. Ren, T. Luo, B. Jia, B. Yang, L. Wang, and L. Xing, "Improved ant colony optimization for the vehicle routing problem with split pickup and split delivery," *Swarm and Evolutionary Computation*, vol. 77, 2023, doi: 10.1016/j.swevo.2023.101228.

[65] S. Lin, A. Liu, J. Wang, and X. Kong, "A Review of Path-Planning Approaches for Multiple Mobile Robots," *Machines*, vol. 10, no. 9, 2022, doi: 10.3390/machines10090773.

[66] Y. Chen, G. Bai, Y. Zhan, X. Hu and J. Liu, "Path Planning and Obstacle Avoiding of the USV Based on Improved ACO-APF Hybrid Algorithm With Adaptive Early-Warning," in *IEEE Access*, vol. 9, pp. 40728-40742, 2021, doi: 10.1109/ACCESS.2021.3062375.

[67] A. Kumar, S. Tiwari, and A. Majumder, "A*-VG algorithm: A hybrid algorithm for the path planning of inspection robots," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 45, no. 386, 2023, doi: 10.1007/s40430-023-04249-z.

[68] T. Xiong, H. Li, K. Ding, H. Liu, and Q. Li, "A Hybrid Improved Symbiotic Organisms Search and Sine–Cosine Particle Swarm Optimization Method for Drone 3D Path Planning," *Drones*, vol. 7, no. 10, 2023, doi: 10.3390/drones7100633.

[69] X. Zhao, Y. Sun, Y. Li, N. Jia, and J. Xu, "Applications of machine learning in real-time control systems: A review," *Measurement Science and Technology*, vol. 36, 2024, doi: 10.1088/1361-6501/ad8947.

[70] H. Suwoyo, A. Adriansyah, J. Andika, A. U. Shamsudin, and Y. Tian, "An Effective Way for Repositioning the Beacon Nodes of Fast RRT Results Utilizing Grey Wolf Optimization," *Journal of Robotics and Control (JRC)*, vol. 6, no. 1, pp. 272–284, 2024, doi: 10.18196/jrc.v6i1.22062.

[71] N. Gupta and A. Mathur, "Performance Comparison of PSO, ACO and Hybrid ACO-PSO for Multi-Target Search Using Autonomous Swarm Drones in Obstacle-Rich Environments," *2025 8th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, pp. 1-6, 2025, doi: 10.1109/IEMENTech65115.2025.10959663.

[72] J. Chung, J. Fayyad, Y. A. Younes, and H. Najjaran, "Learning team-based navigation: A review of deep reinforcement learning techniques for multi-agent pathfinding," *Artificial Intelligence Review*, vol. 57, no. 41, 2024, doi: 10.1007/s10462-023-10670-6.

[73] M. Reda, A. Onsy, A. Y. Haikal, and A. Ghanbari, "Path planning algorithms in the autonomous driving system: A comprehensive review," *Robotics and Autonomous Systems*, vol. 174, 2024, doi: 10.1016/j.robot.2024.104630.

[74] L. Wang, L. Liu and X. Lu, "Robot Path Planning Based on Generative Learning Particle Swarm Optimization," in *IEEE Access*, vol. 12, pp. 130063-130072, 2024, doi: 10.1109/ACCESS.2024.3457957.

[75] X. Yu, B. Su, Z. Wang, J. Zhang and C. Zhang, "Cognitive Robotics: Enhancing Multirobot Target Search in Unknown Environments Through Adaptive Communication Strategies," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 55, no. 5, pp. 3449-3463, 2025, doi: 10.1109/TSMC.2025.3540059.

[76] O. Hamed and M. Hamlich, "Hybrid Formation Control for Multi-Robot Hunters Based on Multi-Agent Deep Deterministic Policy Gradient," *MENDEL*, vol. 27, no. 2, pp. 23–29, 2021, doi: 10.13164/mendel.2021.2.023.

[77] X. Zhou, H. Ma, J. Gu, H. Chen, and W. Deng, "Parameter adaptation-based ant colony optimization with dynamic hybrid mechanism," *Engineering Applications of Artificial Intelligence*, vol. 114, 2022, doi: 10.1016/j.engappai.2022.105139.

[78] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942-1948, 1995, doi: 10.1109/ICNN.1995.488968.

[79] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *Biosystems*, vol. 43, no. 2, pp. 73–81, 1997, doi: 10.1016/S0303-2647(97)01708-5.

[80] H. Lu *et al*., "Two-layer path planning framework for WMRs in dynamic environments: Optimized ant colony algorithm and dynamic window

approach," *Transactions of the Institute of Measurement and Control*, 2025, doi: 10.1177/01423312241296969.

[81] J. Li, L. Wan, Z. Huang, Y. Chen, and H. Tang, "Hybrid Path Planning Strategy Based on Improved Particle Swarm Optimisation Algorithm Combined with DWA for Unmanned Surface Vehicles," *Journal of Marine Science and Engineering*, vol. 12, no. 8, pp. 1–16, 2024, doi: 10.3390/jmse12081268.

[82] M. Wu, G. Li, J. Liao, H. Wang, W. Liu, X. Yan, M. Yang, and S. Li, "Multi-strategy hybrid adaptive dung beetle optimization for UAV photogrammetric 3D path planning under complex constraints," *Scientific Reports*, vol. 15, 2025, doi: 10.1038/s41598-025-98563-2.

[83] J. Xin, J. Kim, S. Chu, and N. Li, "OkayPlan: Obstacle Kinematics Augmented Dynamic real-time path Planning via particle swarm optimization," *Ocean Engineering*, vol. 303, 2024, doi: 10.1016/j.oceaneng.2024.117841.