Applied Computer Science 2026, vol. 22, no. 1, pp. 140–153
https://doi.org/10.35784/acs_7891

Submitted: 2025-06-09 | Revised: 2025-08-11 | Accepted: 2025-08-29

Keywords: multi-robot exploration, optimization, ant colony optimization, artificial pheromones, reinforcement learning

Nabila RAHMOUNE [1]*, *Adel RAHMOUNE* [1]

[1] LIMOSE Laboratory, Computer Science Department, Faculty of Sciences, University M'Hamed Bougara of Boumerdes, Boumerdes, Algeria., n.rahmoune@univ-boumerdes.dz, ad.rahmoune@univ-boumerdes.dz

* Corresponding author: n.rahmoune@univ-boumerdes.dz

Efficient multi-robot exploration of unknown environments using inverted ant colony optimization and reinforcement learning

Abstract

Collaborative environmental exploration by a fleet of mobile robots is of growing interest, especially in the context of unknown environments. Exploration algorithms find diverse and critical applications, such as search and rescue, underwater surveillance, and space observation. However, despite significant advances in the field, a persistent gap between research results and their translation into real-world applications is a major obstacle to the deployment of effective solutions. This paper proposes a hybrid approach, called IACO-RL, which combines inverse ant colony optimization (IACO) with reinforcement learning (RL) to improve exploration efficiency. This method aims to maximize space coverage and minimize exploration time, with the additional goal of accurately locating mines hidden in the environment. The IACO algorithm directs robots to scarce or unexplored areas by reversing the classical pheromone deposition mechanism, thus promoting efficient spatial dispersal. For its part, the RL module allows each agent to learn autonomously from its interactions with the environment, thus enhancing its adaptability and local decision-making capacity. Experimental results, obtained through simulations in different environmental scenarios, show that the IACO-RL approach outperforms single methods in terms of coverage, speed and mine detection capacity. These performances confirm the relevance of this hybridization and highlight that effective mine detection results directly from the efficiency of the exploration performed by the multi-robot system.

1. INTRODUCTION

Autonomous exploration of unknown environments remains a fundamental challenge in mobile robotics. Manual control of robots for data collection is often inefficient due to communication latency, limited scalability, and risk of interference. As an alternative, autonomous navigation strategies have been developed that allow robots to iteratively select target areas to maximize spatial coverage. In this work, the term target refers to any object or location that the robots are tasked to detect during the exploration process. Among these targets, mines represent a specific category of dangerous objects that require special attention in our scenarios.

This paper introduces an original hybrid algorithm called Inverted Ant Colony Optimization combined with Reinforcement Learning (IACO-RL). Unlike classical ACO-RL combinations, our approach inverts the conventional pheromone mechanism to guide robots preferentially to less explored regions, thereby maximizing spatial coverage and minimizing redundant visits. Crucially, this inverted pheromone principle is applied consistently within both the metaheuristic optimization and reinforcement learning components, resulting in a novel hybrid method that enhances both global coordination and individual adaptability in multi-robot exploration.

However, these strategies often depend on predefined waypoints or full knowledge of the environment, which limits their adaptability in dynamic or unmapped scenarios. Furthermore, increasing the dimensionality of state and action spaces significantly increases the complexity of optimizing exploration strategies.

To address these limitations, a novel hybrid algorithm, Inverted Ant Colony Optimization with Reinforcement Learning (IACO-RL), is proposed. This approach combines a modified version of Ant Colony Optimization (IACO), which prioritizes less visited areas to encourage spatial dispersion, with Reinforcement Learning (RL), which allows robots to refine their decisions through reward feedback. The goal is to enable mobile robots to efficiently explore unknown environments without prior knowledge of the number or location of other robots, mines, or obstacles. Each robot operates autonomously using local information, while a shared

map ensures global coordination and avoids redundant exploration. This collaborative framework aims to maximize coverage, improve target detection, and reduce exploration time. An effective navigation strategy is essential to guide each robot through multiple potential paths, avoid obstacles, and reach goals under real-world constraints. Optimizing robot trajectories improves the quality of exploration and minimizes travel distance and time. In multi-robot exploration, coordination is critical to avoid collisions and redundant motions, and to ensure successful mission completion in minimal time. Path planning plays a key role in this process by computing safe and efficient trajectories that adapt to environmental changes (Tong et al., 2022).

Autonomous exploration methods are typically classified into three main categories (AbuJabal et al., 2024): (i) machine learning algorithms, such as supervised learning for structured environments and reinforcement learning for adaptive decision making; (ii) bio-inspired algorithms, which mimic natural behaviors to optimize navigation; (iii) hybrid techniques, which combine multiple strategies to improve efficiency, robustness, and adaptability.

Recent studies have focused on learning-based methods for autonomous navigation. For example, Chen et al. (2020) use graph neural networks (GNNs) to efficiently model spatial environments. Hybrid approaches such as GNN-DRL or Graph-STNN combining spatial, temporal, and historical data to improve decision making. Multi-agent deep reinforcement learning (DRL) has shown promising results in decentralized coordination (Wang et al., 2025), while other learning-based strategies use convolutional neural networks (Wu et al., 2024) or macro-action prediction (Tan et al., 2022) to improve planning. GNN-based decentralized control further improves adaptability and team behavior. These approaches differ from traditional map-based methods by allowing direct learning of optimal actions (Wang et al., 2024).

Path planning strategies are generally classified into global and local methods (Hou et al., 2024). Global approaches, such as Dijkstra, A*, RRT*, and PRM, require full knowledge of the environment to precompute trajectories and are particularly suitable for static environments. In contrast, local planning methods such as the Dynamic Window Approach (DWA), Artificial Potential Fields (APF), and Timed Elastic Band (TEB) rely on on-board sensor data to operate in real time, making them more effective in dynamic or partially known environments.

Nature-inspired metaheuristics have also been widely applied to robotic path planning, with swarm intelligence techniques providing robustness and adaptability through decentralized behavior. Examples include genetic algorithms (GA) (Suresh et al., 2022), ant colony optimization (ACO) (Miao et al., 2021), particle swarm optimization (PSO) (Sui et al., 2023), gray wolf optimizer (GWO) (Gul et al., 2021), and artificial bee colony (ABC) (Liang & Lee, 2015). Several enhanced variants improve performance through reward mechanisms, energy awareness, or search balancing.

Hybrid approaches that combine the strengths of different algorithms have shown high efficiency in robot path planning (Matos et al., 2025). Examples include RRT*-B-spline for smooth trajectories (Eshtehardian & Khodaygan, 2023), ACO-A* for fast path finding (Zhang et al., 2023), and combinations such as BSA Kriging or ACO-ILS with PPO for optimized scheduling and engineering design (Dong et al., 2022). Further studies have explored hybridizations of swarm intelligence with other metaheuristics, such as ACO-PSO (Shi et al., 2007), GA-RL (Dou et al., 2015), PSO-HSA (Abhishek et al., 2020), and WSA-APF (Hamed et al., 2022), all of which enhance adaptability in dynamic environments.

Specific combinations have been used to address different navigation challenges: ACO-DWA for reactive navigation (Wang et al., 2022), ACO-GA for improved coverage and obstacle avoidance (Liu et al., 2020), and a hybrid ACO-GA method for multi-robot coverage based on sensor range (Gong & Lee, 2023). Advanced ACO variants such as MMAS (Shafiq et al., 2022) and ACO-SFLA (Pu et al., 2024) incorporate improved pheromone update mechanisms and global-local search balancing. Other works integrate ACO with modified A* (Zhang et al., 2023a), PSO (Zhang et al., 2023b), or ABC (Li et al., 2023) to handle complex or hazardous environments.

The main objective of this paper is to develop a new hybrid method for exploring an unknown environment, including static or dynamic obstacles, using a fleet of mobile robots. The proposed IACO-RL approach aims at guiding these robots, whose motion energy is limited, to achieve maximum spatial coverage. The method is based on the hybridization of a metaheuristic inspired by Ant Colony Optimization (IACO), in an inverse version, and Reinforcement Learning (RL). This combination allows for a more efficient exploration of the environment and consequently leads to a fast and accurate detection of targeted hazardous substances, in particular mines.

2. METHOD

The proposed approach combines a bio-inspired Inverted Ant Colony Optimization (IACO) strategy with Reinforcement Learning (RL) techniques to enhance exploration of unknown environments. The inverted pheromone mechanism directs robots to less or never explored areas, promoting natural dispersal, reducing redundant paths, and improving overall area coverage. In parallel, each robot learns from its interactions with the environment by receiving positive rewards for successful mine detection or effective obstacle avoidance, and negative rewards for inefficient movements. These reward signals allow for the gradual adaptation of decision making and the reinforcement of effective behaviors. The interaction between collective guidance based on inverted pheromone cues and individual learning driven by environmental rewards balances inter-robot coordination with autonomous adaptability. This synergy improves trajectory coherence, strengthens agent cooperation, and increases overall exploration efficiency, especially in complex or dynamically changing environments.

2.1. Environment modeling

Mathematical modeling of the problem and the environment is a crucial step in robot trajectory planning. Among the commonly used methods for representing the environment, we distinguish grid-based approaches, topological methods, and visibility graphs (Li et al., 2022). In this study, a 2D grid-based approach is used to represent the environment. In the grid map, cells are binarized according to equation (1), where a value of 0 indicates a free space, while a value of 1 represents an obstacle. Obstacles can be divided into two categories: static and dynamic. Static obstacles include mines, which are the primary target to be located, as well as other immobile objects that robots must avoid while moving. Dynamic obstacles are the robots themselves and require careful coordination to avoid collisions.

$$G(i,j) = \begin{cases} 0, & free\ space \\ 1, & obstacle \end{cases} \quad (1)$$

where (i,j) $\in C$, the set of all grid cells defined as $C = \{(\mathrm{i,j}) \mid \mathrm{i} = 1, \ldots, \mathrm{Mr};\ \mathrm{j} = 1, \ldots, \mathrm{Nc}\}$
Mr and Nc represent the total number of rows and columns of the grid, respectively.

2.2. Problem modeling

Let a fleet of N robots operate in a discrete space C (a grid of cells) with:
- $(x, y) \in C(x, y)$: the position of a robot in the grid,
- $O \subset C$: the set of obstacle,
- $\tau(x, y)$: the pheromone concentration at cell *(x,y)*,
- E_i: the energy level of robot i, bounded by E_{max} ,
- $T \subset G$: the set of targets cells to be discovered,
- V_i : the set of cells visited by robot i.
- $\mathrm{N(x,y)} \subset C$:nthe set of reachable neighboring cells from position (x, y).

The objective function is to maximize the spatial coverage while minimizing the computational time (total number of steps) and the redundancy of revisiting the same cells.

$$\max \left(\sum_{(x,y)\in G} f(x,y) - \lambda_1 \sum_{i=1}^{N} t_i - \lambda_2 \sum_{(x,y)\in G} R(x,y)\right) \quad (2)$$

With $f(x, y) = 1$ if at least one robot has visited the cell (x,y); otherwise, it is 0,
where :

t_i – the total time taken by robot i,
$R(\mathrm{x,y})$ – the number of times a cell (x,y) has been revisited,
λ_1 , λ_2 – are weighting factors to balance coverage and computational efficiency.

Under the constraints:
1. Movement Constraints: a robot *i* can move from one cell to another if the cell is free and accessible:

$(x', y') \in \{ (x+1, y), (x-1, y), (x, y+1), (x, y-1), (x+1, y+1), (x-1, y-1), (x+1, y-1),$

$(x-1, y+1)\}$ with the condition that$(x', y') \notin O(x', y')$,

where:

O – represents the set of obstacles.

And it selects the cell with the lowest pheromone concentration:

$$(x', y') = \underset{(x', y') \in N(x, y)}{arg\ min\ \tau(x', y')} \tag{3}$$

where: $N(x, y)$ represents the set of accessible neighbouring cells.

2- Energy constraints: the robot's energy decreases with each movement:

$$\mathrm{E_i} \leftarrow \mathrm{E_i} - \Delta \mathrm{E_{move}} \quad \text{if } \mathrm{E_i} > 0 \text{ , else the robo stops} \tag{4}$$

If a robot finds a target $(\mathrm{x}', \mathrm{y}')$ in T, it receives an energy reward:

$$E_i = E_i + R_{reward} \quad if\ (x', y') \in T \tag{5}$$

2.3. Theoretical foundation of IACO-RL

The proposed hybrid algorithm, Inverted Ant Colony Optimization with Reinforcement Learning (IACO-RL), guides robots to prioritize exploration of less visited areas by reversing the conventional pheromone mechanism: instead of following high pheromone concentrations, robots choose actions that lead to cells with minimal pheromone concentrations. Pheromone concentrations at each cell are dynamically updated after visits through evaporation and incremental reinforcement. Reinforcement learning is implemented via Q-learning, where each robot maintains a Q-table estimating action utilities, updating values using the Bellman equation based on positive rewards for target detection and penalties otherwise (Bellman & Kalaba, 1965).

The interaction between IACO and Q-learning is mutually influential: the global pheromone map guides robots to underexplored regions, while Q-learning allows individual adaptation of movement policies based on experience, balanced by an ε-greedy strategy to avoid local optima. In addition, each robot has a limited energy reserve that decreases with each movement, emphasizing efficient trajectory planning. Energy depletion halts exploration and affects both pheromone updates and learning, integrating resource management into decision making. This hybrid framework effectively combines global coordination, local learning, and energy awareness to improve exploration efficiency and robustness. The following details formalize the pheromone updating, action selection, Q-learning updates, and obstacle avoidance mechanisms implemented in the algorithm.

Let $\tau(\mathrm{x}, \mathrm{y})$ denote the pheromone concentration at position(x, y). The robots' movement is influenced by both the pheromone concentration and the Q-learning algorithm, which integrates energy-aware decision-making. The pheromone concentration at a position (x, y) is updated after each robot's visit:

$$\tau(\mathrm{x}, \mathrm{y}) = (1-\rho)\tau(\mathrm{x}, \mathrm{y}) \ + \ \Delta\tau(\mathrm{x}, \mathrm{y}) \tag{6}$$

Where ρ is the evaporation rate (0<ρ<1) and Δτ(x,y) corresponds to the pheromone increment added after the robot's visit, typically a small positive value. To select its next action, the robot chooses the path with the minimum pheromone concentration. At position (x,y), the action a is determined by identifying a*chosen to minimize the concentration by the function :

$$a^* = \underset{a \in A}{arg\ min\ \tau(x_a, y_a)} \tag{7}$$

Where A is the set of possible actions, and (x_a, y_a) is the cell reached by taking action a from the current position (x, y). To prevent trapping in cycles, we use an ϵ-greedy policy to incorporate a degree of random exploration. A random action is selected with probability ϵ, while the optimal action is chosen with probability $1 - \varepsilon$.

Reinforcement learning based on Q-learning allows robots to optimize their movements using a Q-table $Q(x, y, a)$, that estimates the utility of each action. The robot receives a reward $R(\mathrm{x}, \mathrm{y})$ defined as:

$$\mathrm{R}(x, \mathrm{y}) = \begin{cases} 1 & \text{if a target is found,} \\ -0.1 & \text{therwise} \end{cases} \quad (8)$$

In addition to the reward, the robot's energy level is affected by its actions. When a target is found, the robot receives an energy boost of R_{Reward}which allows it to extend its exploration. Conversely, any action that does not lead to a goal results in an energy reduction of $R_{Penalty}$which models the energy cost associated with unsuccessful movements. After performing an action a, the value Q(x,y,a) is updated according to Bellman's equation (Bellman & Kalaba, 1965; Fayaz et al., 2022). To incorporate the immediate reward and the estimation of future rewards. The update follows the learning rule:

$$Q(x, y, a) = Q(x, y, a) + \alpha\big(R(x', y') + \gamma max_{a'}\, Q(x', y', a') - Q(x, y, a)\big) \quad (9)$$

Where α represents the learning rate, which controls the speed at which the model adapts, while γ is the discount factor, which weights the importance of future rewards.

To ensure safe navigation, the robot only considers actions that lead to free positions, i.e. positions that are not occupied by obstacles. Positions corresponding to obstacles are excluded from the set of possible actions A, thus guaranteeing that robots avoid collisions by not moving into obstructed cells.

2.4. Algorithm IACO-RL

The proposed hybrid IACO-RL algorithm combines the inverse ant colony optimization principle originally introduced in this work with a Q-learning reinforcement learning framework to optimize multi-robot exploration in unknown environments.

In the IACO component, robots preferentially choose paths with lower pheromone concentration to encourage exploration of less visited areas. Initially, the pheromone map is set to low uniform values. Unlike standard ACO, where frequently visited cells accumulate higher pheromone levels, our inverse approach reduces pheromone levels for cells that are frequently revisited. This mechanism prevents path stagnation and promotes coverage diversity. Pheromone updates are performed according to equation (6), which implements this inverse reinforcement rule and directly influences the action selection process.

To complement this global exploration strategy, Q-learning is used to refine local decision making based on past experience. Using a ε-greedy policy, each robot consults its Q-table $Q(x, y, a)$to choose actions that balance exploration and exploitation. The learning rate α, discount factor γ, and exploration rate ε are initialized as described in Section 3.1. The reward structure assigns +1 for the first discovery of a target and -0.1 for unproductive moves, thus encouraging efficient discovery while limiting redundant paths. Q-values are updated according to equation (9), and these updates subsequently affect the likelihood of selecting certain paths in future steps.

The two components operate in a feedback loop: the pheromone map generated by IACO biases the ε-greedy action selection in Q-learning, while the evolving Q-values change the probability of selecting low-pheromone paths. This interaction ensures that global coverage goals and local optimization criteria remain aligned throughout the mission.

Finally, energy constraints are embedded in both modules. Every robot starts with a finite energy budget $E_{max}$with each movement reducing the remaining energy. When energy is depleted, the robot stops exploring, forcing the IACO-RL framework to make path selection decisions that maximize area coverage before battery depletion.

Algorithm IACO-RL

Step 1: **Initialize environment**

- Define environment size, number of targets, robots, and obstacles
- Set initial positions of targets, robots, and obstacles
- Initialize pheromone grid with low default values
- Initialize Q-table Q(x, y, a) for Q-learning
- Set Q-learning parameters: learning rate α, discount factor γ, exploration rate ε

Step 2: **Main exploration loop**

For each episode **Do**
For each robot **Do**
Initialize robot energy level E ← E_{max}
While (E > 0) and (max steps not reached) **Do**
1. Action Selection (Exploration vs. Exploitation)
Generate a random number r ∈ [0,1]
If r < ε **Then**
Select a random action a among accessible positions (positions not occupied by obstacles)
Else
Select action a* that minimizes the pheromone concentration by equation 7, considering only accessible positions.
EndIf
2. Validate selected position
Let (x', y') be the position reached by action a
If (x', y') is occupied by an obstacle **Then**
Reject action a
Select another feasible action among accessible positions
Repeat validation
EndIf
3. Movement execution
Move robot to position (x', y')
Decrease energy: E ← E - ΔE_move
Mark position (x', y') as visited
4. Reward assignment
If a target is discovered for the first time at (x', y') **Then**
Add target to discovered list
Assign reward R(x', y') = +1
Else
Assign reward R(x', y') = -0.1
EndIf
5. Energy constraint handling
If E ≤ 0 **Then**
Stop robot movement
EndIf
6. Pheromone update (IACO)
Update pheromone τ(x,y) according to Equation (6)
7. Q-learning update
Update Q(x,y,a) using Equation (9)
EndWhile
EndFor
EndFor

3. RESULTS AND DISCUSSION

To quantitatively evaluate the efficiency and robustness of the IACO-RL algorithm, the method was implemented using Python 3.12.8, chosen for its efficiency and rapid prototyping capabilities. The algorithm was evaluated on a personal computer equipped with an Intel® Core™ i5 processor, 4GB of system memory, and the Windows® 10 Professional 64-bit operating system. To quantitatively assess the efficiency and robustness of the IACO-RL algorithm, several performance metrics were considered:

- Exploration rate (%): Percentage of the environment explored.

$$\text{Exploration Rate} = \frac{(\text{Area Explored})}{\text{Total Area}} \times 100 \quad (10)$$

- Exploration Time (s): Time elapsed from the start of exploration to the end of coverage.

– Target Detection Rate (%): Percentage of mines successfully located.

$$\text{Target Detection Rate} = \frac{(\text{Number of Targets Detected})}{\text{Total Number of Targets}} \times 100 \quad (11)$$

– Redundancy Rate: Number of revisits to already explored cells.

$$\text{Redundancy Rate} = \left(\frac{\text{Number of Revisits}}{\text{Total Number of Visits}}\right) \times 100 \quad (12)$$

3.1. Validation tests

Extensive testing was conducted across different categories and scenarios of the environment to determine the optimal values of these parameters, leading to improved results in terms of exploration rate and computation time.

The common initial conditions for all experiments were set as follows:

– Pheromone_grid = 0.1.
– Evaporation_rate (ρ) = 0.5
– Pheromone_increment (Δτ) = 1
– Alpha = 0.1
– Gamma = 0.9

A reference test scenario was defined with the following parameters:

– grid_size = (10, 10)
– num_targets = 5
– num_robots = 2
– num_obstacles = 15
– targets = {(1, 1), (6, 3), (2, 9), (7, 4), (1, 6)}
– robots_initial_positions = [(2, 5), (9, 4)]
– num_episodes = 10
– num_iterations = 50

The results demonstrate the effectiveness of the hybrid approach in a constrained environment containing 15 obstacles. A coverage rate of 89% of the free space excluding obstacle-occupied cells was achieved, highlighting the system's ability to efficiently explore navigable areas while avoiding restricted zones. Moreover, all targets were successfully detected, with an average execution time of only 0.02 seconds, underscoring the computational efficiency of the method.

This configuration reveals the strong coordination among the robots and the relevance of the decisions made throughout the exploration process. The synergy between the global guidance mechanisms of IACO and the local adaptability provided by RL contributes to a fast and structured traversal of the environment. Figure 1 illustrates the resulting exploration, showing the robots' trajectories and the obstacle distribution across the simulated grid.

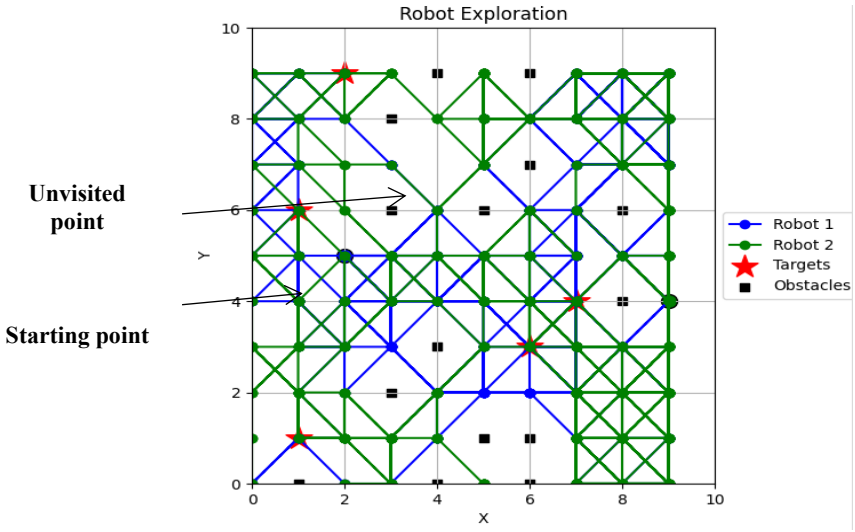

Fig. 1. Exploration map generated by IACO-RL

Table 1 summarizes the configuration parameters, including the number of robots, grid size, distribution of obstacles and mines, and energy reserves for different scenarios. Table 2 shows the results in terms of exploration rate, execution time, and mine detection capability.

Tab. 1. Configuration parameters for exploration

Configuration	Robots number	Map size	Obstacles number	Mines number	Episodes number	Energy
Config. 1	3	50 × 50	20	5	10	80
Config .2	4	100 × 100	50	10	10	150
Config .3	5	150 × 150	80	20	15	400
Config .4	5	200 × 200	120	30	15	550

Tab. 2. Average performance results from exploration tests

Configuration	Exploration rate %	Execution time (s)	Detection Rate (%)	Redundancy Rate (%)
Config. 1	93.80	0.58	100	10.5
Config. 2	94.36	2.85	100	13.7
Config. 3	92.62	6.58	89	15.2
Config. 4	90.98	10.27	91	17.6

The average coverage rate of the unknown environment reaches 92.94%, which facilitates the detection of most targets. Moreover, the average revisit rate of previously explored areas is limited to 14.25%, indicating an overall efficient and optimized exploration process, achieved within an average execution time of 5.07 seconds. In addition to these results, further tests were conducted under three different scenarios to evaluate the adaptability and robustness of the hybrid approach under varying environmental conditions:

1. Fixed positions of mines and robots.
2. Fixed robot positions with random mine placement.
3. Random positions of both mines and robots.

These scenarios range from static to highly dynamic environments and are designed to challenge the system's ability to maintain high performance under various constraints.

The results are shown in Figures 2 and 3. In all three scenarios, the exploration rate remains consistently high, exceeding 98.5%.

Figure 2 shows the combined effect of the number of iterations and the number of robots on the exploration performance. It shows a clear trade-off: increasing the number of iterations allows the system to achieve optimal coverage with fewer robots, and conversely, more robots can compensate for fewer iterations. The

execution time increases only moderately as more robots are introduced, demonstrating the effective coordination and scalability of the algorithm.

Figure 3 focuses on the effect of the number of iterations alone. The exploration rate improves rapidly with more iterations and plateaus near 99%, indicating that beyond a certain threshold, additional iterations yield minimal gains. This behavior underscores the efficiency of the algorithm, which achieves near-optimal performance without unnecessary computational overhead. Overall, these results confirm the robustness and efficiency of the hybrid method. It consistently achieves near-complete coverage in diverse settings while balancing robot deployment, execution time, and iteration count, making it well suited for real-world multi-robot exploration tasks.

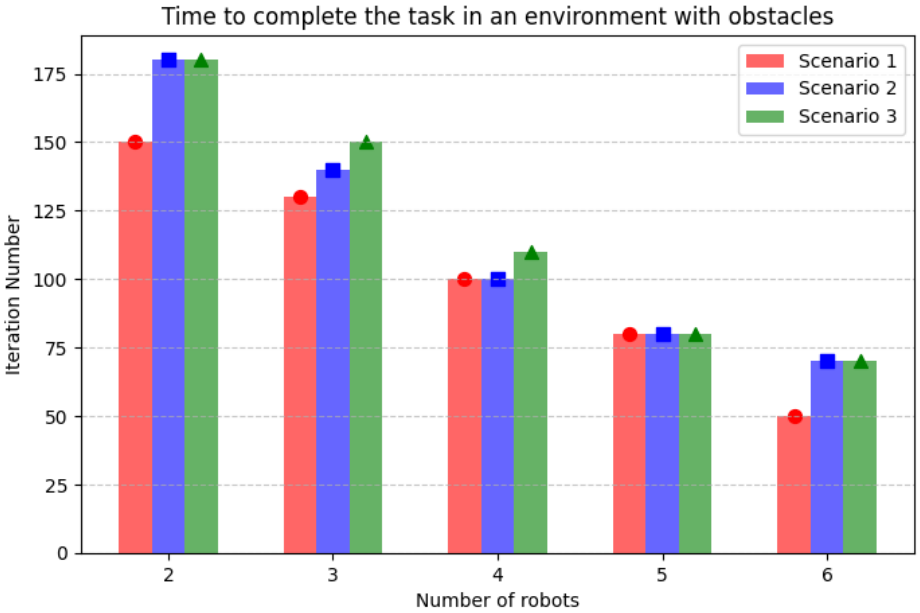

Fig. 2. Impact of iterations on the robots number

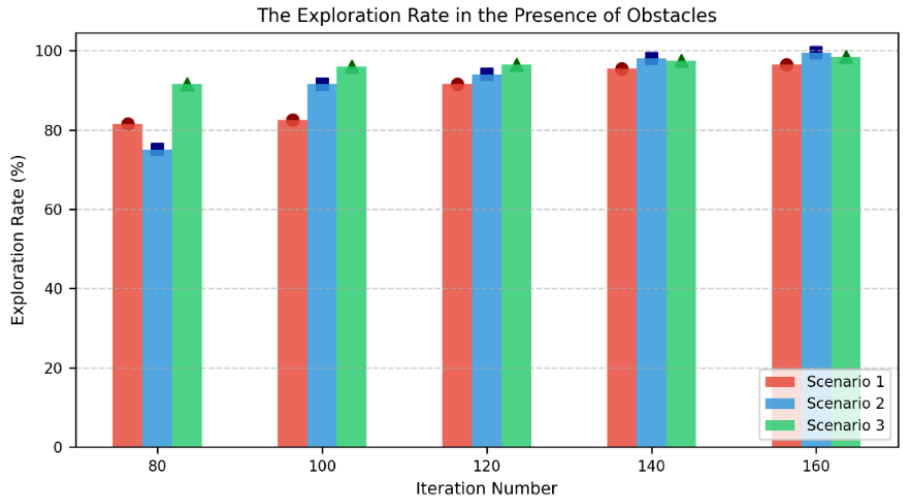

Fig. 3. Impact of iterations on the exploration rate

Tab. 3. Performance Comparison of the Proposed Hybrid Method (IACO-RL) Against Standard ACO and Pure RL

Configuration	Method	Exploration rate %	Execution time (s)	Detection Rate (%)	Redundancy Rate (%)
Config. 1	ACO	54.08	0.52	60	41.3
	RL	38.12	0.50	40	48.2
	IACO-RL	93.80	0.58	100	10.5
Config. 2	ACO	61.45	2.69	70	39.7
	RL	46.73	2.42	50	49.8
	IACO-RL	94.36	2.85	100	13.7
Config. 3	ACO	66.92	6.58	75	34.2
	RL	51.87	5.91	55	41.5
	IACO-RL	92.62	6.58	89	15.2
Config. 4	ACO	68.20	9.95	76.66	32.6
	RL	56.03	9.31	63.33	38.1
	IACO-RL	90.98	10.27	91	17.6

Table 3 shows that the proposed hybrid method, IACO-RL, consistently outperforms the individual ACO and RL methods in all tested configurations. IACO-RL achieves much higher exploration rates, always above 90%, while ACO and RL alone have lower rates, often below 70%. The hybrid method also detects more targets with almost perfect accuracy, compared to the lower detection rates of ACO and RL. In addition, the hybrid method reduces the redundancy rate, i.e. it revisits less already explored areas (less than 18%), while ACO and RL have higher redundancy rates, sometimes up to 48%. The execution time is similar for all methods, showing that the improved performance of IACO-RL does not increase the computational time. These results confirm that the IACO-RL hybrid method achieves better exploration and target detection without sacrificing speed or efficiency. To better illustrate these results, Figure 4 graphically compares the performance metrics of IACO-RL, ACO, and RL across the four configurations.

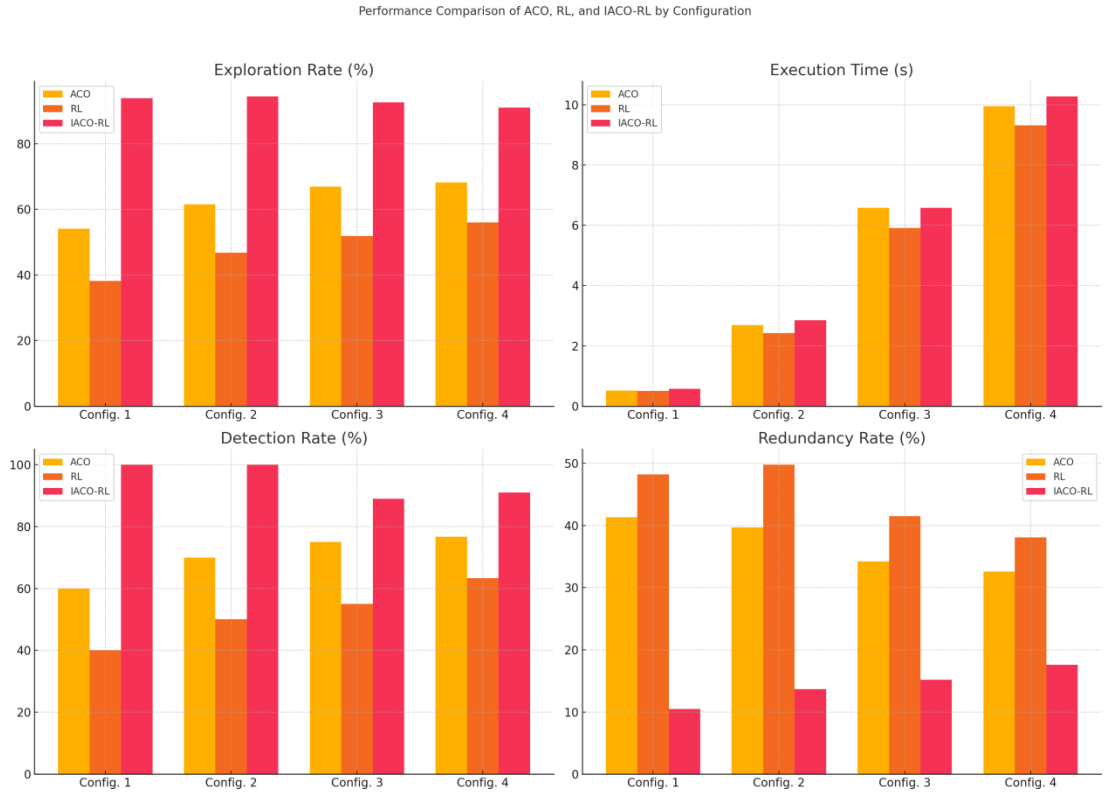

Fig. 4. Comparison of exploration performance between hybrid and individual methods (IACO-RL, ACO and RL)

To evaluate the robustness of the obtained results, each configuration and method (ACO, RL, and IACO-RL) was evaluated over 20 independent simulation runs with randomized target and obstacle placements. Figure 5 shows the mean performance values together with the corresponding standard deviations, represented

as error bars. The results show that the proposed IACO-RL method not only achieves higher exploration rates and detection accuracies than both baseline approaches, but also exhibits lower variability across runs, demonstrating consistent performance in different randomized environments. The relatively small standard deviations across all metrics further confirm the stability and reliability of the proposed approach. Figure 5 illustrates this trend, showing that IACO-RL maintains superior and stable performance compared to ACO and RL across all configurations.

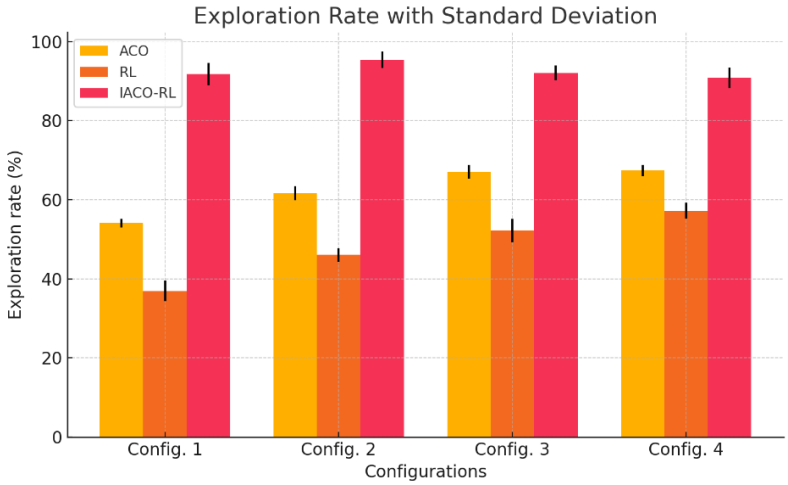

Fig. 5. Statistical analysis of exploration for ACO, RL, and IACO-RL

3.2. Analysis and comparison

The performance of robotic exploration methods is generally evaluated using several criteria, such as the number of robots deployed, the exploration rate, the exploration time, and the size and nature of the environment. This study focuses on two key criteria: spatial coverage and exploration speed. It is important to note that these results depend on factors such as the number of iterations and the energy allocated to the robots. In this context, Table 4 presents a comparative analysis based on criteria common to several methods reported in the literature.

Tab. 4. Comparison of proposed approach with existing methods

Authors	Number of robots	Exploration rate (%)	Exploration time(s)	Environment size
(Bendahmane & Tlemsani,2023)	3	92	5	30 x 30 to 50 x 50
(Zhang & Noguchi, 2016)	2	90	2	15x15
(Tran & Garratt, 2023)	2	84	0.02	10x10*
(Romeh et al., 2023)	2- 4	95	1.5-3	30x30 to 50x50
(Sanghvi et al., 2024)	4	78	3	10x10 to 20x20
(Zhang et al, 2023c)	2	85	0.1-1	20 x 20 to 50 x 50
Proposed method	2 2- 4 4	97.42 95.89 93.83	0.02 0.94 2.89	10x10* 30x30* 100x100*

Environnement* with obstacles.

The proposed method demonstrates excellent exploration capabilities in environments of different sizes, ranging from 10x10 to 100x100. It achieves a high coverage rate, reaching up to 97.42%, while maintaining remarkable time efficiency, with execution times ranging from 0.02 to 2.89 seconds. These results are obtained with a limited number of robots (2 to 4), highlighting the effective coordination and energy management of the algorithm. Moreover, the proposed approach remains competitive with existing methods in the literature, both in terms of coverage rate and execution time, confirming its relevance to real-world and diverse exploration scenarios.

These results are further illustrated in Figure 6, which provides a comparative overview of the exploration rates and execution times between the proposed method and several other approaches reported in the literature.

This visual comparison highlights the consistent performance of the proposed method across different environment sizes, confirming its competitiveness in both coverage and computation time.

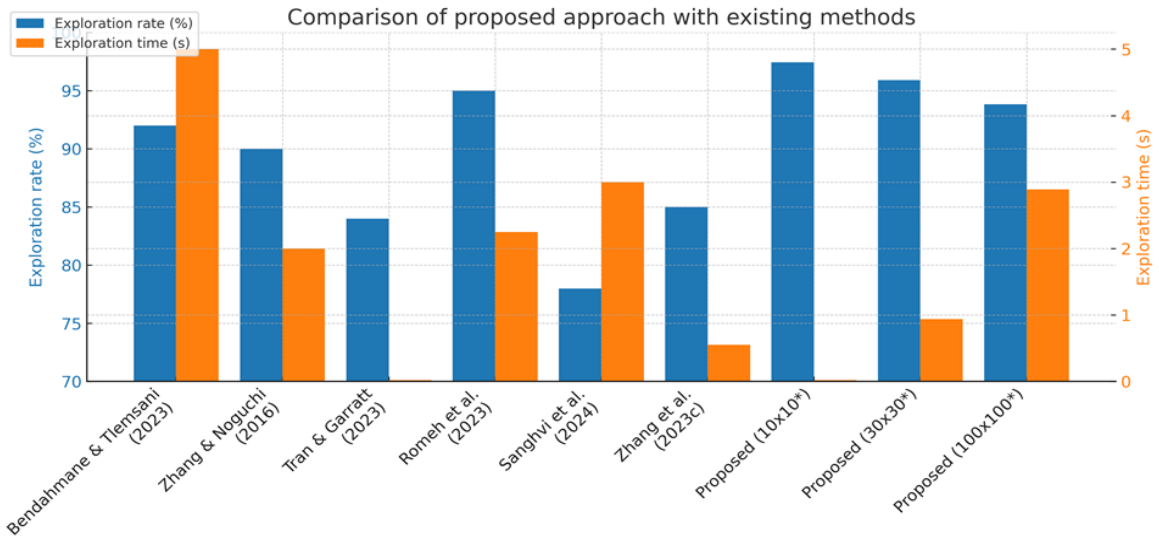

Fig. 6. Comparative analysis of exploration rate and time across different methods

4. CONCLUSIONS

In this paper, we propose a hybrid algorithm that combines Inverted Ant Colony Optimization (IACO) with Reinforcement Learning (RL). Our contributions are: (1) the IACO approach, which inverts the classical ACO principle by directing agents to less visited areas to maximize coverage and reduce redundancy; and (2) its hybridization with RL, which applies the same inverted pheromone principle to action selection to improve decision making. Experiments show that IACO-RL outperforms ACO and RL alone, achieving exploration rates above 90%, near-perfect detection accuracy, and redundancy rates below 18%, with comparable computational times. Statistical evaluation over 20 randomized runs confirms both the superiority and the stability of the method. Moreover, comparison with recent exploration approaches in similar environments shows that IACO-RL achieves up to 97.42% coverage with execution times as low as 0.02 s, remaining competitive or superior in both performance and efficiency even with only 2-4 robots. These results highlight IACO-RL as a high-performance, energy-efficient, and robust solution for exploration in complex and diverse environments. Future work will address larger and more dynamic environments, real-world robot deployment, process parallelization for real-time performance, and integration with advanced planning and control strategies for multi-UGV systems in 3D scenarios.

Conflicts of Interest

The authors declare no conflict of interest.

REFERENCES

Abhishek, B., Ranjit, S., Shankar, T., Eappen, G., Sivasankar, P., & Rajesh, A. (2020). Hybrid PSO-HSA and PSO-GA algorithm for 3D path planning in autonomous UAVs. SN *Applied Sciences*, *2*(11), 1805. https://doi.org/10.1007/s42452-020-03498-0

AbuJabal, N., Baziyad, M., Fareh, R., Brahmi, B., Rabie, T., & Bettayeb, M. (2024). A comprehensive study of recent path-planning techniques in dynamic environments for autonomous robots. *Sensors*, *24*(24), 8089. https://doi.org/10.3390/s24248089

Bellman, R., & Kalaba, R. E. (1965). Dynamic programming and modern control theory. *New York: Academic Press*, *81*.

Bendahmane, A., & Tlemsani, R. (2023). Unknown area exploration for robots with energy constraints using a modified Butterfly optimization algorithm. *Soft Computing*, *27*(7), 3785-3804. https://doi.org/10.1007/s00500-022-07530-w

Chen, F., Martin, J. D., Huang, Y., Wang, J., & Englot, B. (2020). Autonomous exploration under uncertainty via deep reinforcement learning on graphs. *ArXiv, abs/2007.12640*. https://doi.org/10.48550/ARXIV.2007.12640

Dong, L., Yuan, X., Yan, B., Song, Y., Xu, Q., & Yang, X. (2022). An improved grey wolf optimization with multi-strategy ensemble for robot path planning. *Sensors*, *22*(18), 6843. https://doi.org/10.3390/s22186843

Dou, J., Chen, C., & Yang, P. (2015). Genetic scheduling and reinforcement learning in multirobot systems for intelligent warehouses. *Mathematical Problems in Engineering*, *2015*, 597956. https://doi.org/10.1155/2015/597956

Eshtehardian, S. A., & Khodaygan, S. (2023). A continuous RRT*-based path planning method for non-holonomic mobile robots using B-spline curves. *Journal of Ambient Intelligence and Humanized Computing*, *14*(7), 8693-8702. https://doi.org/10.1007/s12652-021-03625-8

Fayaz, S. A., Jahangeer Sidiq, S., Zaman, M., & Butt, M. A. (2022). Machine learning: An introduction to reinforcement learning. In P. Agrawal, C. Gupta, A. Sharma, V. Madaan, & N. Joshi (Eds), *Machine Learning and Data Science* (1st edn, pp. 1–22). Wiley. https://doi.org/10.1002/9781119776499.ch1

Gong, J., & Lee, S. (2023). Hierarchical area-based and path-based heuristic approaches for multirobot coverage path planning with performance analysis in surveillance systems. *Sensors*, *23*(20), 8533. https://doi.org/10.3390/s23208533

Gul, F., Rahiman, W., Alhady, S. S. N., Ali, A., Mir, I., & Jalil, A. (2021). Meta-heuristic approach for solving multi-objective path planning for autonomous guided robot using PSO–GWO optimization algorithm with evolutionary programming. *Journal of Ambient Intelligence and Humanized Computing*, *12*(7), 7873-7890. https://doi.org/10.1007/s12652-020-02514-w

Hamed, O., Hamlich, M., & Ennaji, M. (2022). Hunting strategy for multi-robot based on wolf swarm algorithm and artificial potential field. *Indonesian Journal of Electrical Engineering and Computer Science*, *25*(1), 159. https://doi.org/10.11591/ijeecs.v25.i1.pp159-171

Hou, J., Jiang, W., Luo, Z., Yang, L., Hu, X., & Guo, B. (2024). Dynamic path planning for mobile robots by integrating improved sparrow search algorithm and dynamic window approach. *Actuators*, *13*(1), 24. https://doi.org/10.3390/act13010024

Li, C., Huang, X., Ding, J., Song, K., & Lu, S. (2022). Global path planning based on a bidirectional alternating search A* algorithm for mobile robots. *Computers & Industrial Engineering*, *168*(C), 108123.

Li, G., Liu, C., Wu, L., & Xiao, W. (2023). A mixing algorithm of ACO and ABC for solving path planning of mobile robot. *Applied Soft Computing*, *148*, 110868. https://doi.org/10.1016/j.asoc.2023.110868

Liang, J.-H., & Lee, C.-H. (2015). Efficient collision-free path-planning of multiple mobile robots system using efficient artificial bee colony algorithm. *Advances in Engineering Software*, *79*, 47-56. https://doi.org/10.1016/j.advengsoft.2014.09.006

Liu, Z., Chen, B., Zhou, H., Koushik, G., Hebert, M., & Zhao, D. (2020). Mapper : Multi-agent path planning with evolutionary reinforcement learning in mixed dynamic environments. *ArXiv, abs/2007.15724*. https://doi.org/10.48550/arXiv.2007.15724

Matos, D. M., Costa, P., Sobreira, H., Valente, A., & Lima, J. (2025). Efficient multi-robot path planning in real environments : A centralized coordination system. *International Journal of Intelligent Robotics and Applications*, *9*(1), 217-244. https://doi.org/10.1007/s41315-024-00378-3

Miao, C., Chen, G., Yan, C., & Wu, Y. (2021). Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Computers & Industrial Engineering*, *156*, 107230. https://doi.org/10.1016/j.cie.2021.107230

Pu, X., Song, X., Tan, L., & Zhang, Y. (2024). Improved ant colony algorithm in path planning of a single robot and multi-robots with multi-objective. *Evolutionary Intelligence*, *17*(3), 1313-1326. https://doi.org/10.1007/s12065-023-00821-7

Romeh, A. E., Mirjalili, S., & Gul, F. (2023). Hybrid vulture-coordinated multi-robot exploration : A novel algorithm for optimization of multi-robot exploration. *Mathematics*, *11*(11), 2474. https://doi.org/10.3390/math11112474

Sanghvi, N., Niyogi, R., & Milani, A. (2024). Sweeping-based multi-robot exploration in an unknown environment using webots: *16th International Conference on Agents and Artificial Intelligence*, 248-255. https://doi.org/10.5220/0012343400003636

Shafiq, M., Ali, Z. A., Israr, A., Alkhammash, E. H., Hadjouni, M., & Jussila, J. J. (2022). Convergence analysis of path planning of multi-uavs using max-min ant colony optimization approach. *Sensors*, *22*(14), 5395. https://doi.org/10.3390/s22145395

Shi, C., Ying-yong, B., Zi-guang, L., & Jun, T. (2007). Solving path planning problem by an ACO-PSO hybrid algorithm. *International Conference on Intelligent Systems and Knowledge Engineering*. Atlantis Press. https://doi.org/10.2991/iske.2007.91

Sui, F., Tang, X., Dong, Z., Gan, X., Luo, P., & Sun, J. (2023). ACO+PSO+A* : A bi-layer hybrid algorithm for multi-task path planning of an AUV. *Computers & Industrial Engineering*, *175*, 108905. https://doi.org/10.1016/j.cie.2022.108905

Suresh, K. S., Venkatesan, R., & Venugopal, S. (2022). Mobile robot path planning using multi-objective genetic algorithm in industrial automation. *Soft Computing*, *26*(15), 7387-7400. https://doi.org/10.1007/s00500-022-07300-8

Tan, A. H., Bejarano, F. P., Zhu, Y., Ren, R., & Nejat, G. (2022). Deep reinforcement learning for decentralized multi-robot exploration with macro actions. *IEEE Robotics and Automation Letters, 8*(1), 272-279. https://doi.org/10.1109/LRA.2022.3224667

Tong, X., Yu, S., Liu, G., Niu, X., Xia, C., Chen, J., Yang, Z., & Sun, Y. (2022). A hybrid formation path planning based on A* and multi-target improved artificial potential field algorithm in the 2D random environments. *Advanced Engineering Informatics, 54*, 101755. https://doi.org/10.1016/j.aei.2022.101755

Tran, V. P., Garratt, M. A., Kasmarik, K., & Anavatti, S. G. (2023). Dynamic frontier-led swarming: Multi-robot repeated coverage in dynamic environments. *IEEE/CAA Journal of Automatica Sinica, 10*(3), 646-661

Wang, Q., Li, J., Yang, L., Yang, Z., Li, P., & Xia, G. (2022). Distributed multi-mobile robot path planning and obstacle avoidance based on aco–dwa in unknown complex terrain. *Electronics*, *11*(14), 2144. https://doi.org/10.3390/electronics11142144

Wang, R., Lyu, M., & Zhang, J. (2025). A multi-robot collaborative exploration method based on deep reinforcement learning and knowledge distillation. *Mathematics*, *13*(1), 173. https://doi.org/10.3390/math13010173

Wang, Y., Zhou, Z., Dai, W., Guo, C., Zhu, P., & Liu, P. (2024). Multi-robot obstacle-avoidance formation based on graph neural networks and imitation learning. *2024 China Automation Congress (CAC)* (pp. 5499-5504). IEEE. https://doi.org/10.1109/CAC63892.2024.10864862

Wu, J., Li, H., Li, B., Zheng, X., & Zhang, D. (2024). Optimization of robotic path planning and navigation point configuration based on convolutional neural networks. *Frontiers in Neurorobotics*, *18*, 1406658. https://doi.org/10.3389/fnbot.2024.1406658

Zhang, C., & Noguchi, N. (2016). Cooperation of two robot tractors to improve work efficiency. *Advances in Robotics & Automation*, *5*(2), 1000146. https://doi.org/10.4172/2168-9695.1000146

Zhang, D., Luo, R., Yin, Y., & Zou, S. (2023a). Multi-objective path planning for mobile robot in nuclear accident environment based on improved ant colony optimization with modified A*. *Nuclear Engineering and Technology*, *55*(5), 1838-1854. https://doi.org/10.1016/j.net.2023.02.005

Zhang, D., Yin, Y., Luo, R., & Zou, S. (2023b). Hybrid IACO-A*-PSO optimization algorithm for solving multiobjective path planning problem of mobile robot in radioactive environment. *Progress in Nuclear Energy*, 159, 104651. https://doi.org/10.1016/j.pnucene.2023.104651

Zhang, Y., Wen, Y., & Tu, H. (2023). A method for ship route planning fusing the ant colony algorithm and the A* search algorithm. *IEEE Access*, *11*, 15109-15118. https://doi.org/10.1109/ACCESS.2023.3243810

Zhang, Y., Zhang, Z., & Wang, S. (2023c). Adaptive clustering quasi-line search path planning algorithm based on sampling. *IEEE Transactions on Vehicular Technology*, *72*(2), 1720-1734. https://doi.org/10.1109/TVT.2022.3212982