Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.083871

ARTICLE

EG-IGGO: An Evolutionary Game-Improved Greylag Goose Optimization Algorithm for Multi-Robot Path Planning

Ao Nie[1], **Wei Zhou**[1], **Yi Yu**[1] **and Wan Xu**[1,2,*]

[1]School of Mechanical Engineering, Hubei University of Technology, Wuhan, China
[2]Hubei Key Laboratory of Modern Manufacturing Quality Engineering, Hubei University of Technology, Wuhan, China
*Corresponding Author: Wan Xu. Email: xuwan@mail.hbut.edu.cn

Received: 12 April 2026; Accepted: 09 July 2026; Published: 15 September 2026

ABSTRACT: Currently, mobile robot path planning in unstructured forest environments remains a hot research topic in the robotics field. Studies applying the Greylag Goose Optimization (GGO) algorithm to multi-robot path planning under such scenarios are limited, and these approaches still face significant challenges, such as insufficient trajectory smoothness, frequent coordination conflicts, and relatively slow convergence to optimal solutions. To address these issues, this paper proposes an Evolutionary Game-Theoretic Improved GGO algorithm (EG-IGGO), designed to optimize path quality while ensuring robust obstacle avoidance capabilities. Specifically, two novel strategies—the population alignment strategy and the dual-source adaptive guidance strategy—are integrated into the original GGO framework. The population alignment strategy refines solution quality and enhances trajectory smoothness; the dual-source adaptive guidance strategy balances global exploration and local exploitation, thereby reducing collision risks and mitigating coordination conflicts. Furthermore, to accelerate convergence, an evolutionary game algorithm is introduced to dynamically adjust the probabilities of strategy selection, ensuring individuals consistently adopt the optimal strategy for efficient and robust optimal path search. Rigorous comparative evaluations were conducted using the CEC2022 benchmark functions, where the performance of EG-IGGO was assessed against advanced algorithms including PSO, GWO, GGO, FAPSO-GM, and FSDBWO. Results demonstrate that EG-IGGO outperforms these comparative algorithms and their variants across multiple metrics. Experimental validation in three forest environments with different complexity levels further confirms its effectiveness; compared with existing methods, EG-IGGO achieves a 21.37% improvement in convergence speed and a 28.25% improvement in path smoothness. This study provides a novel theoretical framework and a high-performance solution for multi-robot path planning in complex environments.

KEYWORDS: Evolutionary game; multi-robot; greylag goose optimization algorithm; path plan

1 Introduction

In forest disaster prevention and emergency rescue tasks, mobile robots have become a key force for improving inspection efficiency and ensuring rescue safety due to their flexibility and intelligence in complex woodland environments [1,2]. However, the unstructured forest environment poses unique and complex combinatorial challenges to robot navigation, which rarely occur simultaneously in structured environments. Dense vegetation occlusion severely limits the perception range and inter-robot visibility, making it difficult for robots to perceive the entire environment and coordinate their actions [3]. Spatial constraints caused by limited traversable passages force multiple robots to share narrow corridors, greatly increasing the risk of spatiotemporal conflicts. In addition, the inherent endurance and coverage limitations of a single robot necessitate the use of multi-robot systems [4], which in turn introduce additional coordination complexity:

robots must not only avoid collisions with static obstacles, but also avoid collisions with one another, while simultaneously optimizing path length and trajectory smoothness [5]. Therefore, multi-robot path planning in forest environments must satisfy constraints such as obstacle avoidance, inter-robot spacing, and velocity simultaneously. These constraints interact within a highly irregular and cluttered spatial structure, making the problem a highly challenging multi-constraint optimization task [6–9]. Therefore, the development of efficient and robust path planning methods for multi-robot systems in such environments is of great practical significance for improving the reliability of forest disaster response systems [10].

Over the years, numerous methods have been proposed to address the robot path planning problem, and these methods can generally be categorized into four major classes. Mathematical programming approaches, such as mathematical programming [11] and model predictive control [12], can guarantee optimality and strict collision avoidance. However, their computational complexity grows exponentially with the number of robots, making them impractical for multi-robot systems operating in large-scale or dynamically changing forest environments. Artificial potential field methods guide robot motion by constructing virtual attractive and repulsive fields [13]. Although they offer real-time responsiveness, they are inherently prone to becoming trapped in local minima, which is a critical drawback in cluttered forest terrain, where irregular obstacle distributions often create dead-end-like potential configurations. Graph-based search methods, including A* variants [14] and conflict-based search (CBS) algorithms [15], provide completeness guarantees, but suffer from severe scalability degradation as the environment becomes more complex, rendering exhaustive graph construction and traversal computationally infeasible in large unstructured forest spaces. Intelligent optimization algorithms, such as genetic algorithms [16], particle swarm optimization [17], and ant colony optimization [18], mimic natural phenomena to approximate optimal solutions at relatively low computational cost, and therefore have been widely applied in multi-robot path planning [19–22]. However, when these heuristic methods are applied to forest environments with high dimensionality, complex constraints, and multiple robots, they often encounter problems such as premature convergence to local optima, insufficiently smooth trajectories, and an inability to properly handle inter-robot coordination conflicts. These limitations motivate the need for an algorithm that can better achieve smooth trajectory generation, reliable obstacle avoidance, efficient inter-robot conflict resolution, and rapid convergence in complex and highly constrained forest environments.

As a typical intelligent optimization algorithm, the Greylag Goose Optimization (GGO) algorithm was proposed by El-Kenawy in 2024 [23]. Inspired by the group behavior and social structure variations of gray geese during migration and foraging, the algorithm employs a V-shaped formation strategy and a grouping strategy to achieve rapid convergence toward the optimal solution. Owing to its favorable balance between exploration and exploitation, GGO has attracted considerable research interest, and an increasing number of studies have been devoted to addressing its inherent limitations. Meanwhile, Shi proposed an improved simulated annealing algorithm, which incorporates a priority strategy to accelerate the optimal path selection process and enhance the coordination among robots [24]. Yu proposed a novel hybrid particle swarm optimization algorithm that incorporates a simulated annealing mechanism to avoid premature local convergence, enabling faster planning of higher-quality unmanned aerial vehicle paths and exhibiting better robustness in complex three-dimensional environments [25]. In addition, Hongmei proposed an enhanced multi-strategy integrated particle swarm optimization algorithm based on guided Latin hypercube initialization. By propagating high-quality solutions and incorporating a progressive fusion elite strategy, the proposed method further improves the overall efficiency and safety of amphibious unmanned vehicle path planning [26]. It can be observed that such studies generally improve the final solution quality by increasing algorithmic complexity and integrating multiple optimization mechanisms. However, while this line of improvement enhances the solving capability, it also tends to weaken the adaptability and robustness of the algorithm to some extent. Similarly, under the constraints of vibration suppression in robotic arms, trajectory smoothness also requires dedicated multi-objective trade-off modeling to achieve an acceptable level of performance [27]. Overall, these improvements and their underlying motivations reveal a series of limitations in optimization algorithms such as the original GGO, including a tendency toward premature convergence, weak population diversity maintenance, and insufficient strategy adaptability. These shortcomings become particularly pronounced when GGO is applied to complex, high-dimensional problems with rich constraints.

When these general limitations are specifically applied to multi-robot path planning in forest environments, they manifest as three concrete performance deficiencies. First, the elite-guided movement strategy in the exploration phase relies entirely on a single global best solution to guide individual movement, which causes excessive population aggregation and increases the risk of spatiotemporal coordination conflicts among robots. Second, the multi-sentinel cooperation strategy in the exploitation phase randomly selects sentinels with equal weights while ignoring their quality differences, resulting in large fluctuations during exploitation and reduced trajectory smoothness. Third, the original GGO coordinates its five strategies through hard-coded switching rules associated with iteration parity and random thresholds, which leads to a rigid and inflexible strategy selection mechanism that cannot adapt to the changing exploration and exploitation requirements across different optimization stages, ultimately slowing down convergence. To directly address these three fundamental issues, this paper proposes an Evolutionary Game-improved Greylag Goose Optimization algorithm (EG-IGGO) and introduces three innovations: (1) a dual-source adaptive guidance strategy is adopted to replace the elite-guided movement strategy, combining each individual's historical best position and the global leading position through adaptive weighting, thereby weakening the aggregation effect caused by a single source and reducing coordination conflicts; (2) a population alignment strategy is introduced to replace the multi-sentinel cooperation strategy, incorporating distance-weighted neighborhood historical-best focusing to suppress exploitation fluctuations and improve trajectory smoothness; and (3) an evolutionary game-driven strategy selection mechanism is employed to replace hard-coded conditional switching, dynamically adjusting the selection probabilities of all five strategies according to the empirical fitness improvement history of each strategy, thereby ensuring that individuals adopt the most effective strategy at each stage of optimization. Unlike existing GGO variants that address only a single defect in isolation, EG-IGGO integrates these three improvements into a unified framework specifically designed for multi-robot path planning under forest environmental constraints, thereby achieving simultaneous improvements in conflict reduction, trajectory quality, and convergence speed. The main contributions of this study are summarized as follows:

- We propose an improved EG-IGGO algorithm aimed at enhancing the performance of the original GGO algorithm. Through a dynamic adaptive strategy selection mechanism and improved exploration-exploitation strategies, EG-IGGO achieves a better balance during the solution process for complex optimization problems.
- On the basis of the original GGO strategies, EG-IGGO introduces two novel strategies: adaptive guidance and population alignment. The population alignment strategy refines solution quality and improves trajectory smoothness, while the dual-source adaptive guidance strategy balances global exploration and local exploitation to reduce collision risks and coordination conflicts.
- We develop an adaptive dynamic strategy adjustment mechanism that integrates evolutionary game theory with the GGO algorithm. This approach dynamically optimizes the selection probabilities of five strategies based on historical experience, ensuring particles choose the optimal strategy throughout the optimization process.

The remainder of this paper is organized as follows. Section 2 formulates the multi-robot cooperative path planning problem and defines the composite cost function. Section 3 presents the proposed EG-IGGO algorithm in detail, covering the two novel search strategies and the evolutionary game-driven adaptive strategy selection mechanism. Section 4 reports the experimental results, including benchmark evaluation on the CEC2022 test suite, comparative path planning experiments across three forest scenarios, and parameter sensitivity and computational complexity analyses. Section 5 concludes the paper and outlines directions for future work.

2 Model Problem Description

The primary objective of multi-robot cooperative path planning is to achieve coordinated motion toward target destinations under a series of constraints and uncertainties. The set of targets for each robot is denoted as $T = \{T_i \mid i = 1, 2, \ldots, N_T\}$, and the set of start points is represented as $S = \{S_i \mid i = 1, 2, \ldots, N_S\}$. Various environmental threats, including radar detection zones and obstacle regions, are collectively represented by $M = \{M_i \mid i = 1, 2, \ldots, N_M\}$. The set of path points for the robots is given by $P = \{P_i \mid i = 1, 2, \ldots, N_P\}$. In this paper, the path points of each robot refer to a sequence of points constructed between the start and target locations, which form feasible paths for the robot. These points are used to satisfy relevant constraints and adapt to environmental uncertainties. Ultimately, by connecting these path points, the optimal motion trajectory of each robot is obtained.

2.1 Constraint Formulation

To ensure safe and efficient multi-robot cooperative navigation, five categories of constraints must be satisfied: spatial cooperative constraints, path range constraints, velocity constraints, angle constraints, and threat spatial distribution constraints. These constraints collectively capture inter-robot collision avoidance, individual robot physical limitations, and environmental hazard avoidance, and are formally defined in the following subsections.

2.1.1 Spatial Cooperative Constraints

Spatial cooperative constraints among multiple robots, also known as inter-robot collision avoidance constraints, require that during the entire task execution, the distance between any two robots must be greater than a predefined minimum safety distance. This can be expressed as:

$$|d_i - d_k| \geq d_{\text{safe}}, \quad \forall i, k,\ i \neq k, \tag{1}$$

where d_i and d_k denote arbitrary path points of the i-th and k-th robots, respectively, and d_{safe} is the predefined minimum safe distance between any two robots.

2.1.2 Path Range Constraints

During task execution, energy consumption and efficiency of the robot's trajectory should be considered. Hence, the maximum range of each robot's path must be constrained within a certain limit. Suppose the maximum allowable path length for the i-th robot is $L_{\max}$, then:

$$\sum_{j=0}^{N_{P_i}} |P_j P_{j+1}| \leq L_{\max}, \tag{2}$$

where N_{P_i} denotes the number of path points for the i-th robot, thus representing the endpoint index, and $|P_j P_{j+1}|$ is the distance between the j-th and $(j+1)$-th path points along the i-th robot's trajectory. Summing over all path segments yields the total traveled distance for the robot.

Additionally, there is a minimum path segment length constraint to avoid trivial turns. This minimal segment represents the shortest straight-line distance a robot must travel before changing its direction. Formally, it can be expressed as:

$$|P_j P_{j+1}| \geq L_{\min}, \tag{3}$$

where $L_{\min}$ denotes the minimum length of a trajectory segment.

2.1.3 Velocity Constraints

Considering various environmental factors such as road conditions, weather, and restricted zones, the robot's velocity should be constrained within a reasonable range to effectively respond to changing conditions. The velocity constraints for the i-th robot can be formulated as:

$$v_{\min} \leq v_i \leq v_{\max}, \tag{4}$$

where v_i is the actual velocity of the i-th robot, while $v_{\min}$ and $v_{\max}$ are the minimum and maximum allowable velocities, respectively.

2.1.4 Angle Constraints

As this study focuses on mobile multi-robot path planning, the angle constraints mainly refer to the yaw angle constraints. The yaw angle constraint limits the turning angle of the robot's heading between two waypoints. Due to the physical properties of the robot, exceeding these limits during path planning could cause failure in executing movements or even collisions. The angle constraint is expressed as:

$$\theta_{\min} \leq \theta_i \leq \theta_{\max}, \tag{5}$$

where θ_i is the actual heading angle of the i-th robot, and $\theta_{\min}$, $\theta_{\max}$ are the minimum and maximum allowable yaw angles, respectively.

2.1.5 Threat Spatial Distribution Constraints

In the robot operational space, there exist various threats, such as obstacles, radar scanning regions, and inaccessible areas. Considering these threats in multi-robot path planning aligns with practical scenarios. Proactively avoiding these areas improves the task execution performance of the multi-robot system. The threat model can be represented as:

$$M_j = \{M_x, M_y, M_r\}, \tag{6}$$

where $\{M_x, M_y\}$ denotes the coordinates of the threat M_j center, and M_r represents its influence radius.

2.2 Cost Function Method

In Section 2.1, we have discussed various constraints that need to be considered in multi-robot cooperative path planning. The goal of multi-robot cooperative path optimization is to find the optimal feasible path for each robot under these constraints. To address this complex multi-constraint optimization problem, this study adopts a penalty function approach. The specific penalty functions are defined as follows.

First, the cost function associated with the path length constraint can be expressed as:

$$f_L = \sum_{j=0}^{N_{p_i}-1} \sqrt{(x_i^{j+1} - x_i^j)^2 + (y_i^{j+1} - y_i^j)^2}, \tag{7}$$

where $P_i = (x_i, y_i)$ denotes the position of robot i. The total path length is obtained by accumulating the lengths of consecutive path segments. For environmental obstacles, when the distance between the robot and an obstacle is less than the safety distance, a large penalty is imposed based on the proximity; otherwise, the penalty is zero:

$$f_T = \sum_{j=1}^{N_{p_i}-1} \sum_{k=1}^{N_m-1} \begin{cases} (d_s + r_m) - d_{jm}, & \text{if } 0 < d_{jm} - r_m \le d_s \\ 0, & \text{otherwise} \end{cases}, \tag{8}$$

where N_m represents the number of threats or obstacles, d_{jm} is the distance from the j-th path segment point of the robot to the center of the m-th threat, r_m is the radius of the m-th threat, and d_s is the robot's safety distance. Considering the inherent physical characteristics of the robot, the optimized path should maintain smoothness to reduce motion hazards and improve efficiency. The turning angle is defined as the angle between the current and previous path directions, and the corresponding penalty function is:

$$f_\theta = \sum_{j=2}^{N_{p_i}-1} \theta_{ij}, \qquad \theta_{ij} = \arccos\left(\frac{\left(x_{ij} - x_{i(j-1)}, y_{ij} - y_{i(j-1)}\right) \cdot \left(x_{i(j+1)} - x_{ij}, y_{i(j+1)} - y_{ij}\right)^T}{\left\|\left(x_{ij} - x_{i(j-1)}, y_{ij} - y_{i(j-1)}\right)\right\| \cdot \left\|\left(x_{i(j+1)} - x_{ij}, y_{i(j+1)} - y_{ij}\right)\right\|} \right). \tag{9}$$

Here, θ_{ij} denotes the turning angle at the j-th path point of robot i, and f_θ reflects the total bending degree of the robot's path. During multi-robot cooperative motion, a certain safety distance L_{safe} between robots must also be guaranteed to prevent collisions and damage. The collision cost between robots is calculated as:

$$f_C = \sum_{i=1}^{N} \sum_{k=1}^{N} B_{ik}, \quad B_{ik} = \begin{cases} B, & \text{if } L_{ik} \le L_{\text{safe}} \\ 0, & \text{otherwise} \end{cases}, \tag{10}$$

where L_{ik} is the distance between path points P_i and the path points of robot k, L_{safe} is the safety distance, and B is a penalty constant that encourages maintaining a safe distance between robots. Considering the smoothness of actual robot movement, velocity constraints are also imposed. The velocity penalty function is:

$$f_v = \sum_{j=1}^{N_{p_i}-1} C_{ij}, \quad C_{ij} = \begin{cases} C, & \text{otherwise} \\ 0, & \text{if } v_{\min} \le v_i \le v_{\max} \end{cases}, \tag{11}$$

where C_{ij} is the penalty on velocity v_i for robot i on path segment j. Therefore, the overall fitness function is the sum of these penalty terms:

$$F = \min\left(f_L + f_T + f_\theta + f_C + f_v\right). \tag{12}$$

In the path optimization process, the designed optimization algorithm searches the solution space to minimize the cost function F, enabling robots to move smoothly and safely within the feasible domain and reach their target locations. To efficiently solve such complex multi-constraint optimization problems, it is necessary to construct a stable and well-convergent optimization framework. By integrating bio-inspired mechanisms of intelligent optimization algorithms and performance evaluation strategies, the following sections present the proposed optimization algorithm. Its goal is to effectively bridge the gap between theoretical path planning constraints and practical engineering applications, providing smooth, safe, and feasible motion trajectories for robots.

3 Improved Greylag Goose Optimization Algorithm Based on Evolutionary Game Theory

3.1 Overall Framework of the Algorithm

The original GGO algorithm achieves optimization through a cooperative mechanism between the exploration group and the exploitation group [23]. In the exploration phase, the algorithm includes three strategies. The first is the *elite-guided movement strategy*, in which each individual updates its position using the global best solution $\mathbf{X}^*$ as the sole guiding target (Eq. (1)). The second is the *random exploration strategy*, which performs updates based on the differential guidance of three randomly selected individuals (Eq. (2)). The third is the *spiral search strategy*, which constructs a spiral search pattern through an exponential function and a cosine function (Eq. (4)). In the exploitation phase, the algorithm contains two strategies. The first is the *multi-sentinel cooperation strategy*, in which three sentinel individuals are randomly selected, and their equally weighted average position is used to guide the updates of other individuals (Eqs. (5) and (6)). The second is the *sentinel vigilance strategy*, which guides the search through periodic perturbations between the current position of an individual and the global best solution (Eq. (7)). In the original GGO, these five strategies are coordinated and switched through complex conditional mechanisms such as iteration parity judgment and random parameter thresholds, resulting in a high degree of coupling.

However, when the original GGO is applied to the multi-robot path planning problem, the above strategy system exhibits two evident limitations. First, the *elite-guided movement strategy* in the exploration phase relies only on a single global best solution to guide individual movement, which causes the population to exhibit an excessively strong aggregation tendency. As a result, the algorithm is more likely to fall into local optima, while also increasing the probability of coordination conflicts in multi-robot path planning. Second, the *multi-sentinel cooperation strategy* in the exploitation phase randomly selects sentinel individuals and uses an equally weighted average, without distinguishing the quality of the sentinels. This leads to large fluctuations during exploitation and negatively affects path smoothness.

To address the above two shortcomings, this paper proposes two new strategies as replacements and improvements. First, To fully utilize the complementary information of individual local search history and population global guidance information, a *Dual-source Adaptive Guidance Strategy* is proposed. This strategy replaces the original *elite-guided movement strategy* with an adaptively weighted linear combination of the individual historical best position and the global leader position, thereby weakening the excessive dependence on a single global optimum, enhancing global exploration capability, and reducing the probability of coordination conflicts. Second, inspired by the collective alignment behavior observed in bird migration and fish schooling, a *Population Alignment Strategy* is proposed. This strategy replaces the original *multi-sentinel cooperation strategy* with a dynamically weighted neighborhood historical-best focusing mechanism based on Euclidean distance. Through dynamic neighborhood selection and distance-weighted focusing, it reduces fluctuations in the exploitation phase and improves path smoothness.

In terms of integration, EG-IGGO combines the three retained strategies from the original GGO, namely the *random exploration strategy*, the *spiral search strategy*, and the *sentinel vigilance strategy*, together with the two newly introduced strategies, namely the *Dual-source Adaptive Guidance Strategy* and the *Population Alignment Strategy*, to form a strategy pool consisting of five candidate strategies. Unlike the original GGO, which relies on hard-coded conditional rules for strategy switching, EG-IGGO abandons this coupled coordination mechanism and instead adopts the evolutionary game-driven strategy selection mechanism described in Section 3.3. According to the fitness improvement performance of each strategy in historical iterations, the selection probability of each strategy is dynamically adjusted, enabling each individual to adaptively choose the currently most suitable strategy for position updating at each iteration. This integration scheme not only introduces higher-quality replacement strategies, but also fundamentally changes the strategy coordination mechanism of the original GGO, thereby endowing the algorithm with stronger dynamic adaptability.

The workflow of the EG-IGGO algorithm is illustrated in Fig. 1. First, an initial population of size N is randomly generated, and the fitness of each individual is evaluated. The initial selection probabilities of the five candidate strategies, namely the *Dual-source Adaptive Guidance Strategy*, *Population Alignment Strategy*, *Spiral Search Strategy*, *Random Exploration Strategy*, and *Sentinel Vigilance Strategy*, are all set to $SP_n = 1/5$, and the cumulative fitness improvement variable S_{flag} is initialized to zero. The algorithm then enters an iterative optimization process with the learning period as the basic unit. At the beginning of each iteration, a strategy is assigned to each individual through a roulette-wheel selection mechanism according to the current probability distribution. Among them, the *Dual-source Adaptive Guidance Strategy*, *Random Exploration Strategy*, and *Spiral Search Strategy* are responsible for global exploration to expand the search space, while the *Population Alignment Strategy* and *Sentinel Vigilance Strategy* are responsible for local exploitation to refine the current promising region. After each individual executes the selected strategy and completes position updating, the new fitness value is calculated. If the fitness is improved, the fitness increment is recorded into the corresponding strategy's S_{flag}, and the individual position as well as the global best solution are updated accordingly. Otherwise, the original position is retained and the failure count is accumulated. When the number of iterations reaches the length of the learning period, the cumulative fitness improvement of each strategy is summarized and the normalized reward u_n is calculated. Then, the selection probability of each strategy is updated through the replicator dynamic equation (Eq. (20)). After resetting S_{flag}, the algorithm enters the next learning period. This iterative process continues until the maximum number of iterations $t_{\max}$ is reached, and the global best solution is finally output as the path planning result.

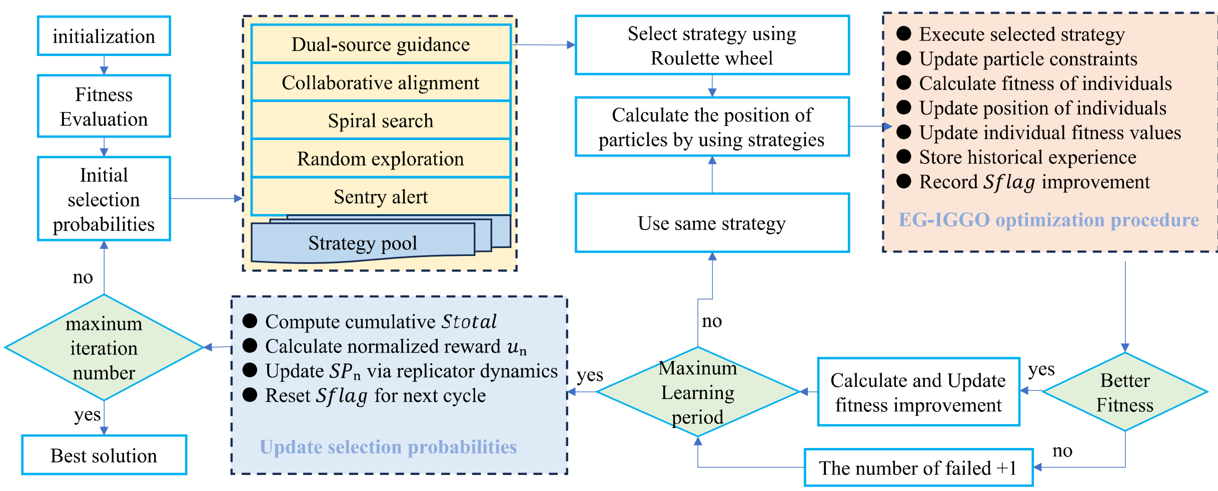


Figure 1: EG-IGGO algorithm framework diagram.

3.2 Dual-Source Adaptive Guidance and Population Alignment Strategies

To overcome the premature convergence problem caused by single-source guidance in the *elite-guided movement strategy*, this paper proposes a *Dual-source Adaptive Guidance Strategy*. This strategy dynamically balances the guidance effects of two information sources on individual position updates through adaptive weights: the individual historical best position $\mathbf{X}^p(t)$ and the global leader position $\mathbf{X}^*(t)$. Its mathematical model is given as follows:

$$\mathbf{X}(t+1) = \mathbf{X}(t) + c_1 \cdot r_1 \cdot \left(\mathbf{X}^p(t) - \mathbf{X}(t)\right) - c_2 \cdot r_2 \cdot \left(\mathbf{X}^*(t) - \mathbf{X}(t)\right) \tag{13}$$

where $r_1, r_2 \in [0,1]$ are random numbers, and c_1 and c_2 are the learning factors for individual historical experience and the global leader, respectively. In Eq. (14), the term $c_1 \cdot r_1$ provides local memory guidance through the individual historical best position $\mathbf{X}^p(t)$, enabling the individual to explore based on its own past experience; the term $c_2 \cdot r_2$ provides global trend guidance through the global best position $\mathbf{X}^*(t)$. The two information sources are weighted by adaptive learning factors, which maintain high exploration diversity in the early stage of the algorithm and narrow the search range in the later stage to meet the requirements of fine exploitation. Compared with the original *elite-guided movement strategy*, this strategy introduces the individual historical best position as a second guiding source, effectively weakening the tendency of the population to gather around a single global optimum and reducing the probability of coordination conflicts in multi-robot path planning.

Inspired by the collective alignment behavior in which the velocity directions of individuals in bird migration and fish schooling tend to become consistent, this paper proposes a *Population Alignment Strategy* to enhance the collaborative stability of the exploitation stage of the algorithm. This strategy establishes an information interaction mechanism between each individual and the historical best individuals within its neighborhood, enabling the population to form an orderly cooperative search pattern. Specifically, for the current individual i, the nearest n neighboring individuals are first identified according to the Euclidean distance d_{ij}, and a neighborhood focal point is computed using inverse distance as the adaptive weight:

$$\mathbf{X}_{focus}(t) = \frac{\sum_{j=1}^{n} \alpha_j \cdot \mathbf{X}_j^*(t)}{n}, \quad \alpha_j = \frac{1}{d_{ij} + \epsilon} \tag{14}$$

$$\mathbf{X}(t+1) = \mathbf{X}(t) + r_3\left(\mathbf{X}_{focus}(t) - \mathbf{X}(t)\right) \tag{15}$$

where $\mathbf{X}_j^*(t)$ denotes the historical best position of the n neighboring individuals closest to individual i, and ϵ is a small positive constant introduced to prevent division by zero. In Eq. (16), a neighboring individual closer to the current one has a larger influence weight, thereby avoiding the information ambiguity caused by a simple equal-weight average. As a virtual cooperative target that integrates neighborhood historical best information, $\mathbf{X}_{focus}(t)$ guides the individual to move smoothly toward a more promising region through Eq. (17). Compared with the random equal-weight selection used in the original *multi-sentinel cooperation strategy*, this strategy significantly reduces the fluctuation amplitude in the exploitation stage through dynamic neighborhood selection and distance-adaptive weighting, effectively improving trajectory smoothness in robot path planning.

3.3 Adaptive Strategy Selection Mechanism Inspired by Evolutionary Game Theory

To address the issue of slow convergence, this paper constructs an adaptive strategy selection mechanism by drawing on the core principles of evolutionary game theory (EGT). This mechanism leverages individual fitness values and historical records of strategy usage to enable particles to adaptively select the most effective strategy at each stage of the optimization process, thereby improving search efficiency and convergence speed [28].

3.3.1 Conceptual Mapping from EGT to the Proposed Mechanism

Before presenting the mathematical formulation, we first establish a formal correspondence between EGT concepts and the components of the proposed mechanism. This mapping is summarized as follows:

- **Players → Individual greylag geese (particles):** Each particle in the population is treated as a player participating in the strategic interaction.
- **Strategies → Five search strategies:** The five candidate strategies in the strategy pool (integrated collaborative, collaborative alignment, spiral search, random exploration, and sentry alert) correspond to the pure strategies available to each player.
- **Payoff → Fitness improvement** Δf: The payoff of a strategy is measured by the fitness improvement it produces when adopted by a particle. A strategy that generates a better solution yields a higher payoff.
- **Population state → Strategy selection probability distribution** $x(t) = (x_1(t), \ldots, x_n(t))$: The population state in EGT, which describes the proportion of individuals adopting each strategy, is mapped to the probability distribution over the five candidate strategies.
- **Replicator dynamics → Strategy probability update rule:** The replicator dynamics equation in EGT, which models the growth of strategies with above-average payoffs, is used to inspire the update rule for selection probabilities.
- **Evolutionarily Stable Strategy (ESS) → Strategy diversity maintenance mechanism:** The ESS concept, which characterizes the resistance of a strategy to invasion by mutants, inspires a diversity-preserving mechanism that prevents any single strategy from completely dominating the population.

This mapping provides a rigorous conceptual foundation for the proposed mechanism and situates it within the EGT framework.

3.3.2 Payoff Computation

In each learning cycle, the mechanism maintains a probability distribution over the $n = 5$ candidate strategies for each particle. The selection probabilities remain constant within a cycle. If a strategy generates a solution superior to the current one, its payoff contribution is recorded; otherwise, it is not. Two arrays are maintained to track fitness differences. The array $S_{\mathrm{flag}_n} = \{\Delta f_1^t, \Delta f_2^t, \ldots, \Delta f_n^t\}$ is initialized to zero. During the optimization process, the payoff update rule is:

$$\Delta f^t = \Delta f^t + f_i^t - f_i^x \tag{16}$$

where f_i^t represents the fitness obtained by the i-th particle using the selected strategy. If this fitness is better than the current fitness f_i^x, the improvement is recorded in S_{flag_n}, indicating that the strategy successfully helped the particle achieve an improved solution. After each iteration, the S_{flag_n} arrays are aggregated into:

$$S_{\mathrm{total},m} = \left[S_{\mathrm{flag}_{n1}}, S_{\mathrm{flag}_{n2}}, \ldots, S_{\mathrm{flag}_{nm}}\right]^T \tag{17}$$

which represents the overall performance of all strategies during m iterations within an update cycle. At the beginning of each learning cycle, $S_{\mathrm{total},m}$ is reset to zero, and its values are updated as iterations progress. At the end of each learning cycle, the cumulative fitness performance of each strategy is computed as:

$$\bar{S}_{\mathrm{total}} = \frac{\sum_{i=1}^{m} S_{\mathrm{total},i}}{f_m} \tag{18}$$

where f_m denotes the total number of fitness evaluations in that learning cycle. The average improvements are then normalized to yield the standardized reward $u_n \in [0,1]$:

$$u_n(t) = \frac{\bar{S}_{\text{total}}}{\sum_{i=1}^{n} \bar{S}_{\text{total},i}} \tag{19}$$

Here $u_n(t)$ is the normalized improvement reward for each strategy during the previous cycle, which serves as the payoff signal for the replicator dynamics-inspired update.

3.3.3 EGT-Inspired Strategy Probability Update

It is important to note the relationship between the following update rule and standard EGT. In classical EGT, the payoff of a strategy is computed from the population state $x(t)$ and a fixed payoff matrix A, i.e., $u(e_h, x) = e_h \cdot Ax$. In the proposed mechanism, however, $u_n(t)$ is computed by directly recording the empirical fitness improvement produced by each strategy during a learning period. Consequently, even for the same population state $x(t)$, $u_n(t)$ may take different values depending on the optimization landscape encountered. This constitutes an empirical payoff estimation approach rather than a matrix-based payoff computation. The proposed mechanism is therefore best understood as an EGT-inspired adaptive framework rather than a strict implementation of standard EGT. Within this framework, the population state of strategy n is updated through the following discrete replicator dynamics-inspired equation:

$$x_n(t+1) = x_n(t) + \beta \times x_n(t) \times \left(u_n(t) - \bar{u}(t)\right) \tag{20}$$

where $\bar{u}(t) = x(t) \cdot u(t)$ denotes the average reward value, β is the learning rate controlling update speed, and $x_n(t)$ is the current strategy selection probability within the cycle. This form is structurally consistent with the commonly used discrete replicator dynamics: if $u_n(t) > \bar{u}(t)$, the sampling probability of strategy n increases; otherwise, it decreases. This manifests an adaptive mechanism whereby more effective strategies gain higher sampling probabilities over time.

3.3.4 ESS-Inspired Strategy Diversity Maintenance

In EGT, an Evolutionarily Stable Strategy (ESS) is one that, once adopted by the majority of the population, cannot be invaded by a small group of mutants adopting a different strategy. This concept characterizes the robustness of a strategy against evolutionary pressures and ensures long-term population stability. Motivated by this concept, we introduce a strategy diversity maintenance mechanism. In practice, the replicator dynamics update in Eq. (20) may cause one strategy to dominate rapidly, driving the probabilities of other strategies toward zero. Once a strategy probability reaches zero, that strategy is permanently excluded from future selection, which reduces search diversity and may lead to premature convergence—analogous to the extinction of mutant strategies in EGT. To prevent this, we impose a lower bound ϵ on all strategy selection probabilities:

$$x_n(t) = \max\left(x_n(t), \epsilon\right), \quad \forall n \tag{21}$$

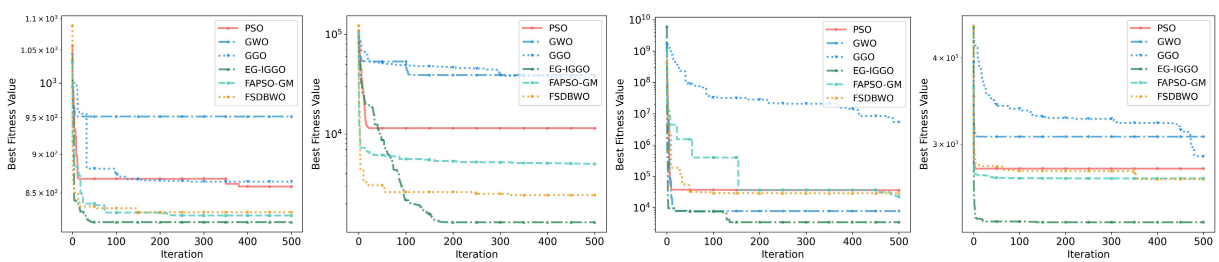
where ϵ is a small positive constant. After applying this lower bound, the probabilities are renormalized to ensure $\sum_n x_n(t) = 1$. This mechanism ensures that all strategies remain active throughout the optimization process, maintaining the strategy diversity necessary for robust global search—consistent with the spirit of ESS in preventing the extinction of viable alternative strategies. The next strategy for each particle is then selected using roulette wheel selection based on the updated probability distribution $x(t)$.

4 Results

To comprehensively evaluate the proposed EG-IGGO algorithm, this section presents experimental results from three perspectives: benchmarking on the CEC2022 test suite to assess general optimization capability, comparative path planning experiments across three forest scenarios of increasing complexity to validate practical effectiveness in multi-robot coordination, and parameter sensitivity and computational complexity analyses to examine robustness and efficiency.

4.1 Test Results of the CEC 2022 Test Function

To fully verify the practical performance of the EG-IGGO algorithm, this paper adopts the CEC2022 benchmark function suite as the evaluation criterion. The suite consists of 12 test functions categorized into four types: unimodal functions, simple multimodal functions, hybrid functions and composition functions. These functions correspond to various engineering challenges in multi-robot path planning, including basic single-path planning scenarios, as well as highly complex multi-robot cooperative path planning scenarios with intricate constraints and local optimum traps. Therefore, the generalization ability of the algorithm can be comprehensively assessed from multiple dimensions. For each benchmark function, 50 Monte Carlo simulation runs are conducted. In the comparative experiments, several optimization algorithms including PSO [29], GWO [30], GGO [23], FAPSO-GM [31] and FSDBWO [32] are selected for performance comparison. Following parameter settings referenced from relevant literature, the dimensionality of all benchmark functions is set to 20, and the population size of each algorithm is fixed at 60. Each algorithm executes 500 iterations, with the final objective function values recorded. The testing results of all algorithms are presented in Fig. 2 and Table 1, where Fig. 2 shows the convergence curves for four different types of CEC2022 test functions, and Table 1 summarizes the average outcomes of the 50 Monte Carlo runs. Since the CEC2022 benchmark functions correspond to minimization problems, lower objective function values indicate superior algorithmic performance.



(a) (b) (c) (d)

Figure 2: This is a convergence curves of the EG-IGGO algorithm on the CEC2022 test functions: (**a**) the convergence curve for function F_4, (**b**) the convergence curve for function F_6, (**c**) the convergence curve for function F_{10}, and (**d**) the convergence curve for function F_{12}.

From the experimental data shown in Table 1, the proposed EG-IGGO algorithm demonstrates overall superior performance compared to the other benchmark algorithms on the test suite. Specifically, for the relatively simple problems (f_1 to f_5), the EG-IGGO algorithm achieves the best results in terms of both the mean and standard deviation, indicating that the algorithm not only finds better solutions but also maintains high stability. For the more complex problems, although the differences in test results are smaller and there may be slight fluctuations in stability, the EG-IGGO algorithm still attains a relative advantage and finds comparatively better solutions. This fully validates the effectiveness of the proposed algorithm in handling complex optimization problems.

Table 1: Results of the CEC 2022.

Func	Metric	PSO	GWO	GGO	EG-IGGO	FAPSO-GM	FSDBWO
F1	mean	3.264e+03	3.636e+03	3.296e+04	3.080e+02	3.657e+02	3.372e+02
	sd	2.497e+02	3.024e+02	5.128e+03	1.487e+01	2.036e+01	1.845e+01
F2	mean	2.823e+03	1.904e+03	2.622e+03	4.066e+02	7.050e+02	1.226e+03
	sd	2.034e+02	1.527e+02	3.156e+02	2.487e+01	4.128e+01	8.243e+01
F3	mean	3.139e+03	2.839e+03	5.312e+03	6.030e+02	2.346e+03	1.449e+03
	sd	2.513e+02	2.236e+02	6.284e+02	3.572e+01	1.845e+02	1.036e+02
F4	mean	1.131e+03	1.172e+03	1.110e+03	8.860e+02	9.300e+02	9.890e+02
	sd	5.127e+01	6.284e+01	7.362e+01	2.036e+01	2.513e+01	3.156e+01
F5	mean	2.847e+03	1.924e+03	5.238e+03	9.329e+02	1.122e+03	8.009e+02
	sd	2.236e+02	1.527e+02	6.542e+02	5.128e+01	6.284e+01	4.572e+01
F6	mean	2.333e+09	9.865e+06	5.313e+06	1.964e+03	1.457e+05	1.194e+04
	sd	3.156e+08	1.274e+06	8.243e+05	8.362e+02	1.527e+04	1.236e+04
F7	mean	6.756e+04	4.337e+04	3.831e+04	1.131e+04	2.024e+03	1.407e+04
	sd	6.284e+03	4.572e+03	4.128e+03	1.074e+03	2.036e+02	1.274e+03
F8	mean	5.227e+08	5.139e+05	2.895e+09	2.227e+03	1.263e+04	2.368e+04
	sd	7.542e+07	6.284e+04	4.128e+08	2.513e+02	1.274e+03	2.034e+03
F9	mean	4.791e+04	3.972e+04	3.333e+04	3.635e+03	2.560e+03	2.458e+03
	sd	4.572e+03	3.845e+03	3.627e+03	3.627e+02	2.236e+02	1.274e+02
F10	mean	5.733e+03	3.282e+03	5.220e+03	2.420e+03	2.805e+03	2.600e+03
	sd	4.128e+02	2.513e+02	4.572e+02	1.845e+02	2.034e+02	1.745e+02
F11	mean	3.220e+03	3.667e+03	3.857e+03	2.696e+03	2.741e+03	2.947e+03
	sd	2.236e+02	5.845e+02	3.627e+02	1.845e+02	2.745e+02	1.927e+02
F12	mean	6.601e+03	6.331e+03	5.903e+03	2.901e+03	4.678e+03	5.762e+03
	sd	4.572e+02	3.243e+02	4.128e+02	2.845e+02	4.056e+02	3.845e+02

Fig. 2 depicts the convergence curves of the simple multimodal function f_4, the hybrid function f_6, and the composite functions f_{10} and f_{12} from the CEC2022 test suite. It is observed that the EG-IGGO algorithm exhibits rapid convergence and strong global exploration capability across these test functions. For simple multimodal problems, although the convergence speed of EG-IGGO is slightly slower, it ultimately locates relatively better solutions. On moderately complex problems, the EG-IGGO algorithm converges more slowly but with smoother and more stable behavior near the optimal solutions. Regarding composite functions, the convergence trajectories of the algorithms differ to some extent; however, EG-IGGO consistently attains relatively superior objective values. In contrast, PSO, GGO, and GWO are more prone to premature convergence to local optima and demonstrate less stability on complex problems compared to EG-IGGO.

4.2 Comparison Results of EG-IGGO with Other Methods in Various Scenarios

To validate the performance of the proposed EG-IGGO algorithm under various scenarios, three different verification scenarios were constructed. The robots have different starting positions, target positions and threat area positions. while the related parameters are shown in Table 2. In Scenario 1, the number of robots is relatively small, and the environment is simple. Scenario 2 increases the number of robots based on Scenario 1 to test the adaptability of the algorithm. Scenario 3 further increases the number of threats on top of Scenario 2, thereby enhancing the environmental complexity and increasing the difficulty of the problem. This gradually increasing scenario design allows a comprehensive evaluation of the robustness and effectiveness of the EG-IGGO algorithm under different levels of environmental complexity.

Table 2: Parameter settings.

Parameter	Value	Parameter	Value	Parameter	Value
Number of Agent	60	β	0.1	c_2	0.075
Max iterations	1000	Learn update period	50	c_1	0.1

The multi-robot path planning results for Scenario 1 are illustrated in Fig. 3. It can be clearly observed that there are significant differences in path quality among the different algorithms, among which the EG-IGGO algorithm generates the smoothest paths and achieves the best obstacle-avoidance performance. Specifically, although both the PSO and FAPSO-GM algorithms enable all robots to successfully reach their target positions, the planned paths contain many redundant turns, resulting in relatively long path lengths. In addition, the obstacle-avoidance margins near certain threat regions are insufficient, which may lead to potential safety risks. Due to premature convergence to a local optimum, the GWO algorithm fails to produce optimal paths for some robots and even results in unnecessary detours. The performance of the GGO algorithm is slightly better than that of the PSO-based algorithms; however, there is still room for improvement in terms of path smoothness and multi-robot coordination. Although the FSDBWO algorithm is also capable of finding feasible paths, its stability is slightly inferior to that of EG-IGGO. In contrast, the proposed EG-IGGO algorithm successfully plans three collision-free paths and achieves the minimum overall cost while satisfying all constraint conditions, which fully demonstrates the superiority of the improved algorithm in terms of global search capability and multi-robot coordination optimization.

The convergence characteristics of different algorithms in Scenario 1 are shown in Fig. 4a. It can be observed from the convergence curves that the proposed EG-IGGO algorithm outperforms the other algorithms overall. Although its convergence speed is slightly slower than that of the other methods, it ultimately converges to a relatively better solution. In contrast, the PSO and GGO algorithms converge more rapidly; however, such fast convergence causes them to become trapped in local optima. Based on the statistical results presented in Fig. 4b and Table 3, a quantitative analysis of the comprehensive performance of each algorithm in Scenario 1 is conducted from the following dimensions. In terms of convergence quality and convergence speed, EG-IGGO achieves the best optimal value of 1226.67 with the shortest average runtime of 0.863 s among all compared algorithms. Although GWO obtains the second-best optimal value of 1235.93, its runtime reaches 1.161 s, which is approximately 34.5% slower than EG-IGGO. PSO and GGO achieve comparable runtimes to EG-IGGO, but their optimal values are approximately 1.4% and 0.8% inferior, respectively. These results demonstrate that EG-IGGO achieves superior solution quality while simultaneously maintaining higher convergence efficiency, confirming that it converges both faster and more effectively. In terms of algorithmic stability and path planning conflicts, EG-IGGO achieves the lowest mean value of 1253.92 among all algorithms, with only 2 coordination conflicts recorded across all runs, substantially fewer than PSO (13), GWO (7), and GGO (9). This indicates that EG-IGGO consistently identifies high-quality, low-conflict paths across repeated runs, demonstrating reliable and stable algorithmic behavior. It is worth noting that although FSDBWO achieves a slightly lower mean value of 1249.30, its conflict count of 6 is considerably higher than that of EG-IGGO, revealing its insufficiency in coordinating multi-robot collision avoidance. In terms of path smoothness and solution consistency, EG-IGGO achieves a curvature value of 0.13, which is approximately 18.75% lower than the second-best FAPSO-GM (0.16) and approximately 45.8% lower than GGO (0.24), demonstrating significantly superior path smoothness over all compared algorithms. Meanwhile, EG-IGGO maintains the lowest standard deviation of 2.85, whereas GGO exhibits the highest standard deviation of 4.71, indicating substantial fluctuation in path quality across runs as well as relatively poor path smoothness. These results confirm that EG-IGGO not only generates smoother trajectories but also exhibits greater consistency across repeated runs.

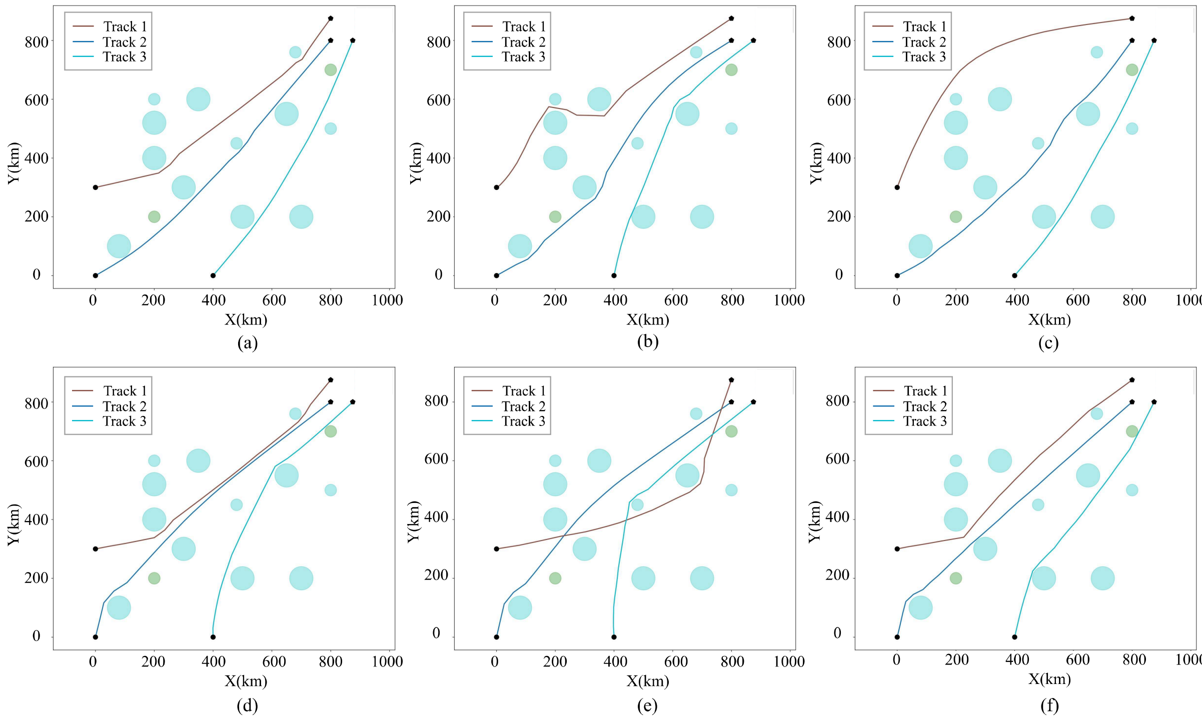


Figure 3: Scene 1: Comparison of path planning results from different algorithms (**a**) EG-IGGO, (**b**) GWO, (**c**) FAPSO-GM, and (**d**) GGO, (**e**) PSO, (**f**) FSDBWO.

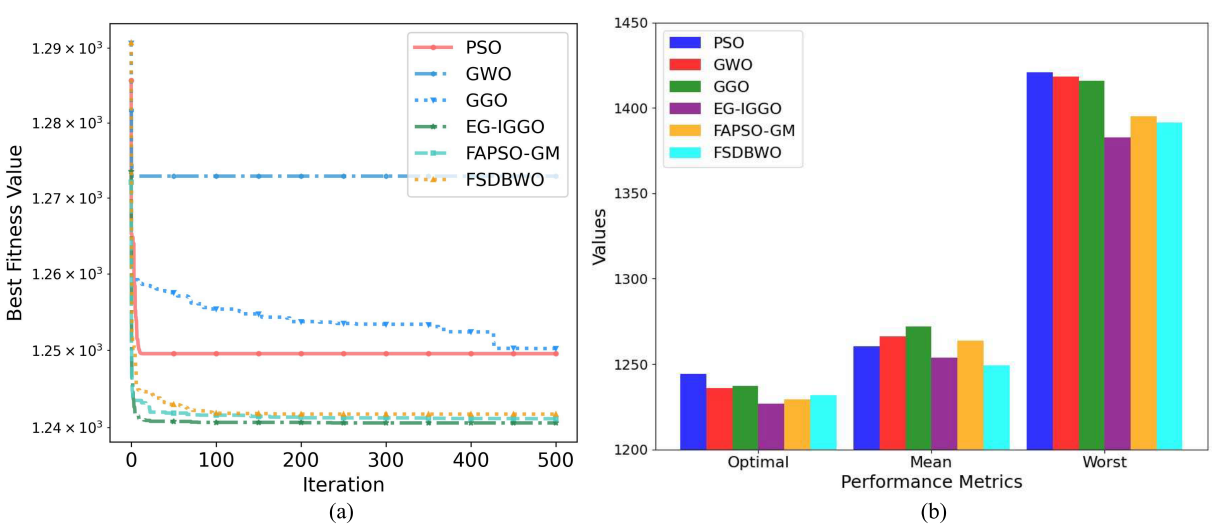


Figure 4: Scene 1: Convergence curve and statistical results (**a**) Convergence curve, (**b**) statistical results.

Table 3: Statistical results of different algorithms in Scenario 1.

	PSO	GWO	GGO	EG-IGGO	FAPSO-GM	FSDBWO
Optima	1244.38	1235.93	1237.29	1226.67	1229.24	1231.93
Mean	1262.27	1266.39	1271.93	1253.92	1263.67	1249.30
Std	4.32	3.28	4.71	2.85	4.15	3.16
Worst	1420.91	1418.26	1415.88	1382.53	1395.26	1391.47
Time (s)	0.896	1.161	0.951	0.863	0.974	0.891
Path Smoothness (Curvature)	0.33	0.41	0.24	0.13	0.16	0.19
Number of conflicts	13	7	9	2	5	6

In the second scenario, the increase in the number of robots significantly raises the problem complexity and the dimensionality of the solution space. A multi-robot system must not only consider path optimization and obstacle avoidance for each individual robot, but also coordinate mutual avoidance among multiple robots to prevent inter-robot collisions while ensuring that the overall path cost is minimized. Fig. 5 shows that, even under more complex conditions, the proposed EG-IGGO algorithm is still able to obtain a multi-robot path with the minimum cost. By examining the path planning results of the different algorithms, it can be found that although the PSO and FAPSO-GM algorithms enable all robots to reach their respective target points, their planned paths are insufficient in terms of robot coordination. In particular, the spatiotemporal coordination at some path intersections is not ideal, which introduces potential collision risks. When handling an increased number of robots, the GWO and GGO algorithms exhibit a noticeable performance degradation, making it difficult for them to converge to the global optimum, and the resulting total path costs are significantly higher. In contrast, the EG-IGGO algorithm demonstrates good scalability and adaptability. It can effectively balance global optimization and local coordination in a multi-robot system and achieves minimization of the overall path cost while satisfying all constraint conditions, including obstacle-avoidance constraints, inter-robot avoidance constraints, and dynamic constraints. In addition, the paths planned by the EG-IGGO algorithm are superior to those of the other algorithms in terms of smoothness and safety, which is crucial for practical robotic task execution.

The convergence characteristics of different algorithms in Scenario 2 are illustrated in Fig. 6a Although the search space becomes larger, all algorithms still exhibit relatively fast convergence. However, the performance differences among the algorithms become more pronounced in terms of the final convergence results. When dealing with a more complex search space, the comparison algorithms are more likely to fall into local optima and have difficulty identifying the global optimum. In contrast, the EG-IGGO algorithm benefits from the adaptive subpopulation size adjustment strategy, which enables it to continuously approach the optimal solution during the convergence process through the improved strategy. Therefore, the EG-IGGO algorithm shows clear advantages in handling the increased complexity caused by a larger number of robots. According to the statistical results of Scenario 2 shown in Fig. 6b and Table 4, a quantitative analysis of the comprehensive performance of each algorithm is conducted from the following dimensions. In terms of convergence quality and convergence speed, EG-IGGO achieves the best optimal value of 1288.86 with the shortest runtime of 1.155 s. Although GWO obtains the second-best optimal value of 1289.83, its runtime is approximately 66.5% slower than EG-IGGO. PSO exhibits both the longest runtime and the worst optimal value, approximately 1.55% inferior to EG-IGGO. These results demonstrate that as problem complexity grows with an increasing number of robots, EG-IGGO maintains its advantage of converging both faster and more effectively. In terms of algorithmic stability and path planning conflicts, EG-IGGO achieves the lowest mean value of 1296.70 with only 5 coordination conflicts, substantially fewer than PSO (27), GWO (19), and

GGO (16). Notably, although FAPSO-GM achieves a slightly lower mean value of 1294.38, its conflict count of 8 remains higher than EG-IGGO, suggesting that its coordination mechanism becomes less reliable as the number of robots increases. In terms of path smoothness and solution consistency, EG-IGGO achieves a curvature value of 0.24, approximately 42.9% lower than PSO and 13.8% lower than GGO, while maintaining the lowest standard deviation of 4.85. By contrast, GWO and GGO exhibit notably higher standard deviations of 7.28 and 7.15, indicating increasing instability as search space dimensionality grows. These results confirm that EG-IGGO generates smoother trajectories with greater consistency under increased problem complexity.

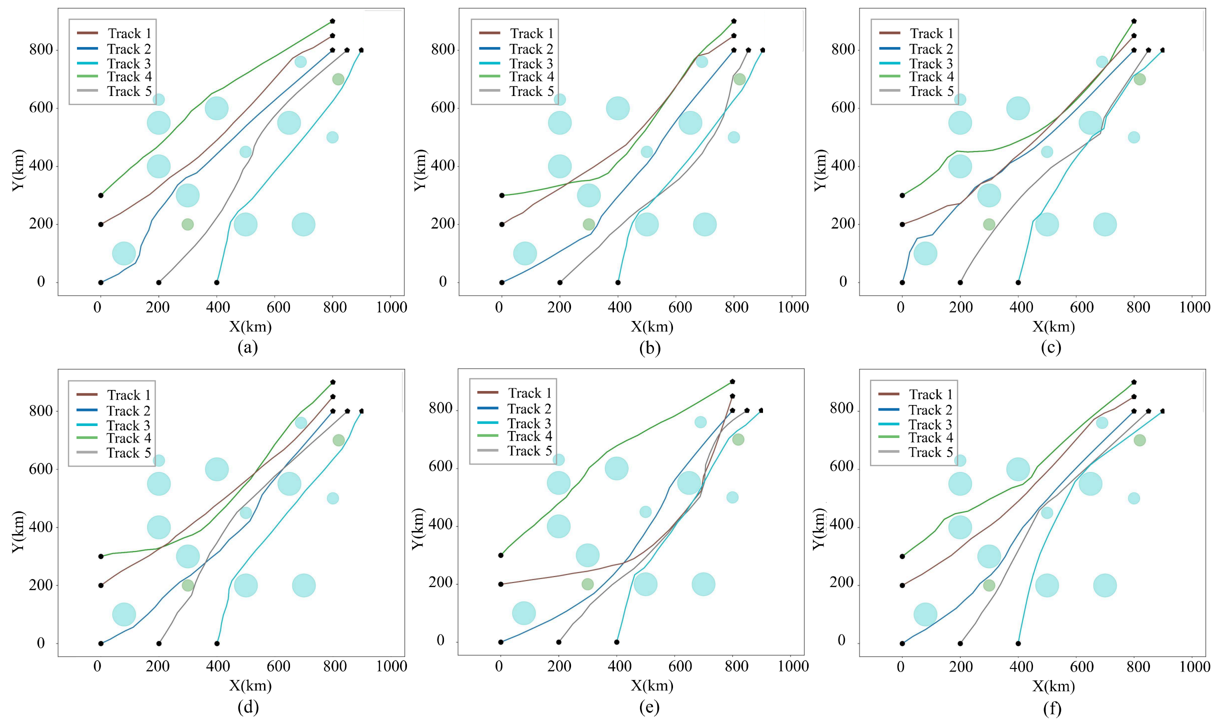


Figure 5: Scene 2: Comparison of path planning results from different algorithms (**a**) EG-IGGO, (**b**) GWO, (**c**) FAPSO-GM, and (**d**) GGO, (**e**) PSO, (**f**) FSDBWO.

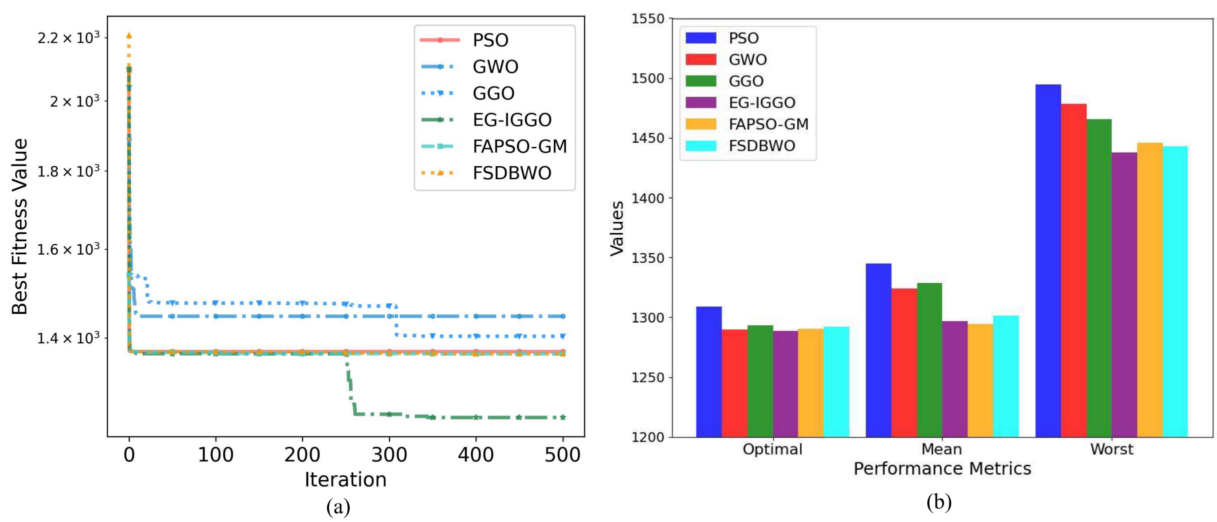


Figure 6: Scene 2: Convergence curve and statistical results (**a**) Convergence curve, (**b**) statistical results.

Table 4: Statistical results of different algorithms in Scenario 2.

	PSO	GWO	GGO	EG-IGGO	FAPSO-GM	FSDBWO
Optima	1308.85	1289.83	1293.06	1288.86	1290.49	1292.01
Mean	1345.04	1324.29	1328.56	1296.70	1294.38	1301.53
Std	6.54	7.28	7.15	4.85	5.01	4.99
Worst	1494.62	1478.16	1465.68	1437.46	1445.82	1442.66
Time (s)	1.569	1.924	1.717	1.155	1.291	1.254
Path Smoothness (Curvature)	0.42	0.35	0.29	0.24	0.31	0.27
Number of conflicts	27	19	16	5	8	10

In Scenario 3, the complexity and density of the environment are further increased by significantly increasing the number of threats and adjusting their spatial distribution. Such a dense-obstacle environment imposes higher requirements on the global search capability, local obstacle-avoidance capability, and constraint-handling ability of path planning algorithms. As shown in Fig. 7, under this more challenging and complex environment, the performance differences among the algorithms become even more significant. Due to their relatively weak global search capability, the PSO and GWO algorithms are prone to falling into local optima in dense-obstacle environments, which results in planned paths that either involve excessively long detours or fail to find feasible solutions in some narrow passages. Although the GGO algorithm shows some improvement over the standard PSO and GWO algorithms, its path quality is still unsatisfactory when dealing with dense threats, with many unnecessary turning points. The FAPSO-GM algorithm improves the ability to escape from local optima to a certain extent, and its performance in dense environments is better than that of standard PSO, but it still cannot reach the level of the EG-IGGO algorithm. The FSDBWO algorithm also exhibits a certain degree of adaptability in Scenario 3, but it is slightly inferior to EG-IGGO in terms of path optimization. In contrast, the proposed EG-IGGO algorithm demonstrates strong performance in dense environments. It effectively balances global exploration and local exploitation and finds a near-optimal path planning solution while satisfying all safety constraints. Specifically, the paths planned by the EG-IGGO algorithm not only have lower total cost, but also perform better in terms of obstacle avoidance, path smoothness, and safety margin. These results fully verify the robustness and effectiveness of the improved algorithm in complex environments.

The convergence characteristics of different algorithms in Scenario 3 are shown in Fig. 8a. Due to the significant increase in environmental complexity, all algorithms face more local optima and constraints during the search process, making convergence more challenging. It can be observed from the convergence curves that the EG-IGGO algorithm is significantly superior to the other comparison algorithms in both convergence performance and solution accuracy. In contrast, traditional optimization algorithms such as PSO and GWO tend to fall into local optima at an early stage. Although improved algorithms such as FAPSO-GM and FSDBWO show some enhancement, their convergence processes are less smooth and relatively slower. The smooth convergence curve of the EG-IGGO algorithm further demonstrates its good stability. According to the statistical results of Scenario 3 shown in Fig. 8b and Table 5, a quantitative analysis of the comprehensive performance of each algorithm is conducted from the following dimensions. In terms of convergence quality and convergence speed, EG-IGGO achieves the best optimal value of 1339.59 with the shortest runtime of 2.483 s. PSO and GGO are approximately 55.9% and 55.2% slower than EG-IGGO, with optimal values approximately 3.0% and 1.2% inferior, respectively. Compared to Scenarios 1 and 2, the performance gap between EG-IGGO and the compared algorithms becomes more pronounced in this most complex environment, demonstrating that EG-IGGO's convergence advantage is further amplified under higher obstacle density. In terms of algorithmic stability and path planning conflicts, EG-IGGO achieves the lowest mean value of 1400.96 with only 9 coordination conflicts, substantially fewer than PSO (47), GWO (38), and GGO (31). Notably, the worst-case performance gap between EG-IGGO (1485.77) and PSO (1780.19) widens substantially compared to Scenario 1, confirming that EG-IGGO's stability advantage becomes increasingly prominent as environmental complexity increases. Although FAPSO-GM and FSDBWO achieve relatively competitive mean values, their conflict counts of 15 and 20 remain considerably higher than EG-IGGO, revealing insufficient coordination capability under denser obstacle distributions. In terms of path smoothness and solution consistency, EG-IGGO achieves a curvature value of 0.32, approximately 28.9% lower than PSO and 21.9% lower than FAPSO-GM, while maintaining the lowest standard deviation of 8.49. GGO exhibits the highest standard deviation of 19.77, more than twice that of EG-IGGO, indicating severe instability in dense-obstacle environments. These results confirm that as environmental complexity increases, EG-IGGO does not suffer performance degradation; instead, its advantages in path smoothness and solution consistency become even more prominent.

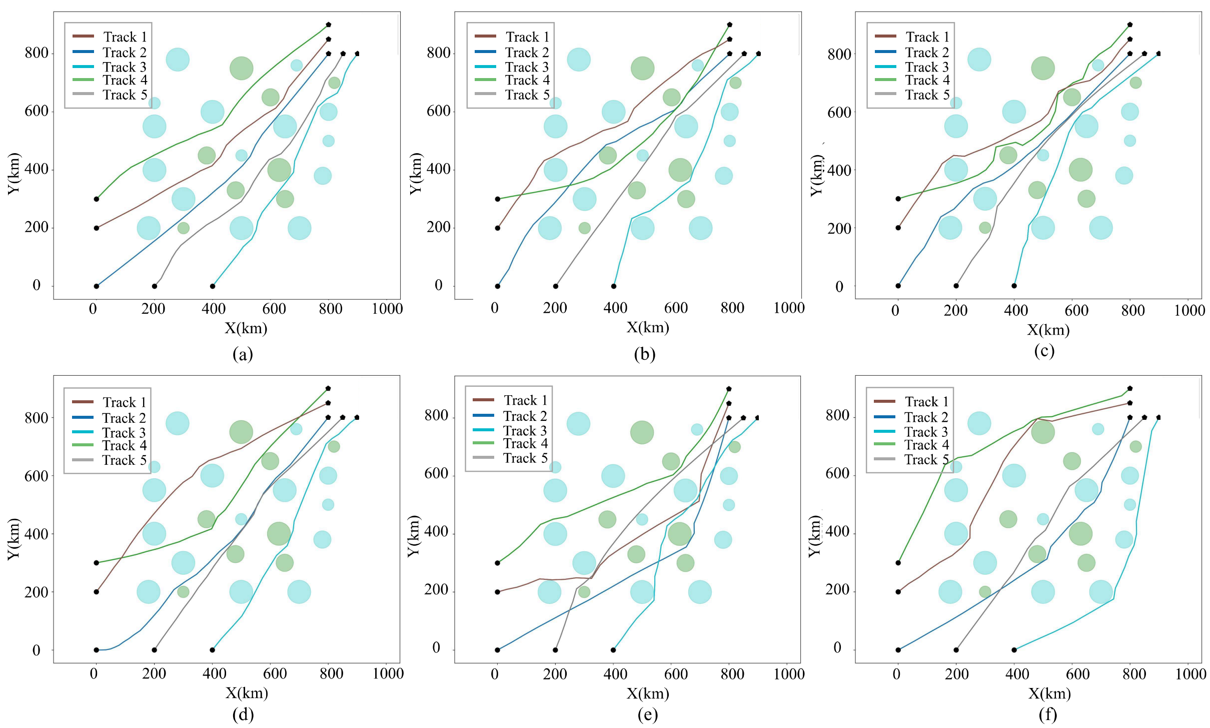


Figure 7: Scene 3: Comparison of path planning results from different algorithms (**a**) EG-IGGO, (**b**) GWO, (**c**) FAPSO-GM, and (**d**) GGO, (**e**) PSO, (**f**) FSDBWO.

Experiments conducted under three scenarios with different levels of complexity demonstrate that the EG-IGGO algorithm performs well in solving the multi-robot cooperative path planning problem. The algorithm shows stable performance in terms of path quality, producing paths with lower total cost, and all key indicators are superior to those of the comparison algorithms. At the same time, the algorithm has a lower standard deviation, indicating strong robustness and low sensitivity to initial conditions. In addition, as the scenario complexity increases, its performance advantages become more prominent, demonstrating good environmental adaptability.

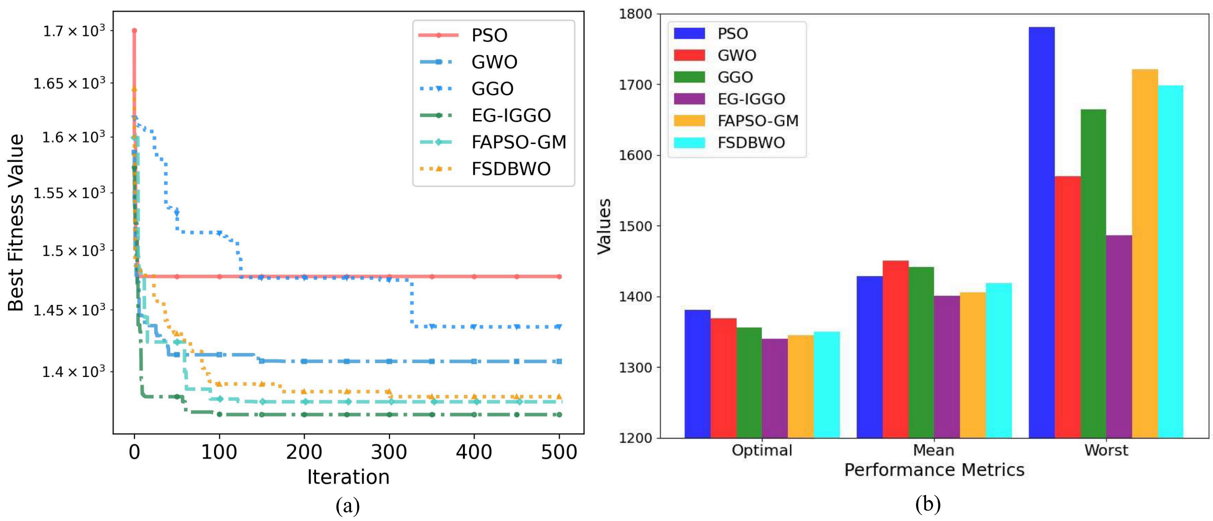


Figure 8: Scene 3: Convergence curve and statistical results (**a**) Convergence curve, (**b**) statistical results.

Table 5: Statistical results of different algorithms in Scenario 3.

	PSO	GWO	GGO	EG-IGGO	FAPSO-GM	FSDBWO
Optima	1380.34	1369.13	1355.65	1339.59	1345.31	1349.74
Mean	1428.63	1450.59	1441.19	1400.96	1405.58	1418.93
Std	17.93	12.58	19.77	8.49	14.22	13.37
Worst	1780.19	1569.46	1664.28	1485.77	1720.49	1698.27
Time (s)	3.874	3.795	3.853	2.483	2.972	3.419
Path Smoothness (Curvature)	0.45	0.36	0.33	0.32	0.41	0.35
Number of conflicts	47	38	31	9	15	20

4.3 Parameter Sensitivity Analysis and Computational Complexity Analysis

4.3.1 Parameter Sensitivity Analysis

In this section, Scenario 2 is selected as the experimental environment to conduct a sensitivity analysis of the core parameters in the proposed algorithm. The evaluation metrics include the average runtime *Time (s)*, path smoothness *Path Smoothness* (PS), and the number of conflicts *Number of Conflicts* (NC). The learning rate β is one of the most critical parameters in EG-IGGO, as it directly controls the update speed of the strategy selection probabilities, thereby affecting the balance between exploration and exploitation. Fig. 9 shows the variation trends of the three evaluation metrics under different β values, where each data point represents the average result over 10 independent runs.

As can be observed from Fig. 9, when β is too small, the update of strategy probabilities is excessively slow, and the algorithm behaves similarly to a uniformly random strategy selection mechanism, resulting in a significant increase in runtime, reduced path smoothness, and more conflicts. As β gradually increases, all three metrics show clear improvement and reach their optimal levels around $\beta = 0.1$ (*Time* = 1.155 s, *Path Smoothness* = 0.24, *Conflicts* = 5). However, when β increases further, the update of strategy probabilities becomes overly aggressive, causing one strategy to dominate rapidly and reducing the diversity of the search process. As a consequence, all three metrics exhibit varying degrees of degradation. In addition to the learning rate β, sensitivity analyses are also conducted on the update period *LP*, the individual experience learning factor c_1, and the global leader learning factor c_2. The detailed experimental results are listed in Table 6. The results indicate that these three parameters achieve the best performance around $LP = 50$, $c_1 = 0.1$, and $c_2 = 0.075$, respectively, and moderate parameter deviations within a reasonable range do not lead to significant performance deterioration.

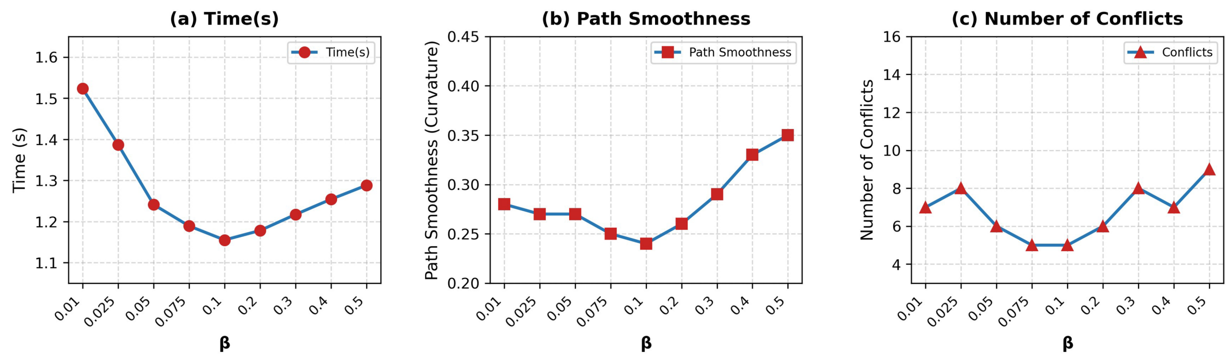


Figure 9: Sensitivity analysis results of parameter β.

Table 6: Sensitivity analysis results of parameters LP, c_1, and c_2.

LP				c_1				c_2			
Value	**Time (s)**	**PS**	**NC**	**Value**	**Time (s)**	**PS**	**NC**	**Value**	**Time (s)**	**PS**	**NC**
5	1.523	0.37	14	0.01	1.489	0.36	13	0.01	1.512	0.39	14
10	1.412	0.33	11	0.025	1.367	0.31	10	0.025	1.389	0.33	11
20	1.298	0.29	8	0.05	1.254	0.27	7	0.05	1.261	0.28	8
30	1.224	0.26	6	0.075	1.192	0.25	6	0.075	1.183	0.24	5
50	1.155	0.24	5	0.1	1.125	0.24	5	0.1	1.192	0.25	5
80	1.187	0.25	6	0.2	1.203	0.26	7	0.2	1.198	0.27	7
100	1.219	0.27	7	0.3	1.278	0.30	9	0.3	1.264	0.31	9
150	1.276	0.30	9	0.4	1.356	0.34	11	0.4	1.342	0.35	12
200	1.341	0.34	11	0.5	1.431	0.38	13	0.5	1.418	0.40	14

4.3.2 Computational Complexity Analysis

From the perspective of computational complexity, the main computational cost of EG-IGGO in each iteration consists of three parts. The complexity of the strategy execution stage is $O(N \cdot D)$, where N denotes the population size and D denotes the problem dimension. The fitness evaluation stage has the same complexity, namely $O(N \cdot D)$. The strategy probability update stage has a complexity of $O(n)$, where $n = 5$ is the number of strategies. Therefore, the overall time complexity of EG-IGGO per iteration is $O(N \cdot D)$, which remains consistent with that of the original GGO algorithm. The additional computational overhead introduced by the strategy probability update, i.e., $O(n)$, is negligible, and this operation is executed only once in each learning period, resulting in a very low amortized cost. From the experimental runtime results, as indicated by the *Time (s)* metric in Tables 3–5, EG-IGGO maintains the shortest or nearly the shortest runtime across all three scenarios. These results demonstrate that, while introducing the evolutionary game-based strategy selection mechanism, EG-IGGO does not impose a significant computational burden. Its computational efficiency fully satisfies the engineering application requirements of practical multi-robot path planning tasks.

5 Conclusion and Future Directions

To address the issues of poor trajectory smoothness, difficulty in avoiding cooperative conflicts, and slow convergence in multi-robot path planning within complex forest environments, this paper proposes the EG-IGGO algorithm, which integrates two novel strategies—the Dual-source Adaptive Guidance Strategy and the Population Alignment Strategy—and replaces the original rigid strategy switching with an evolutionary game-driven adaptive selection mechanism. Experimental results across three forest scenarios of increasing complexity demonstrate that EG-IGGO consistently achieves superior path smoothness, lower conflict frequency, and faster convergence compared with PSO, GWO, GGO, FAPSO-GM, and FSDBWO, while maintaining computational efficiency. Nevertheless, several limitations should be acknowledged. The strategy payoff model relies on simplified empirical fitness improvement estimates, and the formal convergence properties of the resulting dynamics remain to be theoretically established. The interaction between the replicator dynamics update and the diversity maintenance mechanism may introduce oscillatory behavior in the strategy probability distribution under certain problem landscapes. Furthermore, the scalability of EG-IGGO in extremely large-scale multi-robot scenarios has not been fully evaluated, and the key parameters currently require empirical tuning, which may limit generalizability to new problem settings. Future research will focus on four directions corresponding to the above limitations: establishing formal convergence guarantees for the empirical payoff-based replicator dynamics; developing adaptive stabilization mechanisms to suppress potential probability oscillations; evaluating and enhancing scalability for large-scale multi-robot systems; and designing self-adaptive parameter tuning mechanisms to reduce manual configuration dependence. In addition, the algorithm will be extended to handle complex terrain disturbances such as vegetation occlusion and ground roughness, and validated on real robot hardware platforms.

Acknowledgement: Not applicable.

Funding Statement: This work was supported by the Science and Technology Program of Wuhan, China (Grant No. 2025071104010370); and the Key Research and Development Program of Hubei Province, China (Grant No. 2023BEB031).

Author Contributions: Conceptualization, Ao Nie and Wan Xu; methodology, Ao Nie and Wan Xu; software, Ao Nie and Yi Yu; validation, Ao Nie and Wei Zhou; formal analysis, Ao Nie and Yi Yu; resources, Ao Nie and Wei Zhou; data curation, Ao Nie; writing—original draft preparation, Ao Nie; writing—review and editing, Ao Nie; visualization, Ao Nie; project administration, Wan Xu and Ao Nie; funding acquisition, Wan Xu. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The original contributions presented in the study are included in the article material. Further inquiries can be directed to the corresponding author.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Moosavi SKR, Zafar MH, Sanfilippo F. Collaborative robots (cobots) for disaster risk resilience: a framework for swarm of snake robots in delivering first aid in emergency situations. Front Robot AI. 2024;11:1362294. doi:10.3389/frobt.2024.1362294.
2. Tolomei S, Di Lorenzo G, Angelini F, de Simone L, Fanfarillo E, Fiaschi T, et al. Harnessing robotics for European union forest habitats monitoring: toward a robotic-assisted framework for standardized field surveys. IEEE Robot Autom Mag. 2026;33(2):137–45. doi:10.1109/mra.2025.3584343.

3. Ullah I, Adhikari D, Khan H, Anwar MS, Ahmad S, Bai X. Mobile robot localization: current challenges and future prospective. Comput Sci Rev. 2024;53:100651. doi:10.1016/j.cosrev.2024.100651.
4. Gul OM, Erkmen AM. Energy-efficient cluster-based data collection by a UAV with a limited-capacity battery in robotic wireless sensor networks. Sensors. 2020;20(20):5865. doi:10.3390/s20205865.
5. Ul Abideen SZ, Ullah I, Bai X, Zhang W, Li J, Sam Ge S. Multi-robot task assignment: a survey of algorithmic approaches, challenges, and future directions. Comput Sci Rev. 2026;61:100968. doi:10.1016/j.cosrev.2026.100968.
6. Cui Y, Hu W, Rahmani A. Multi-robot path planning using learning-based artificial bee colony algorithm. Eng Appl Artif Intell. 2024;129:107579. doi:10.1016/j.engappai.2023.107579.
7. Wang CH, Hu K, Wu X. Multi-robot path planning in online dynamic obstacle environments based on parallel cooperative strategy optimization algorithm. Discov Comput. 2025;28:132. doi:10.1007/s10791-025-09664-5.
8. Matos DM, Costa P, Sobreira H, Valente A, Lima J. Efficient multi-robot path planning in real environments: a centralized coordination system. Int J Intell Robot Appl. 2025;9(1):217–44. doi:10.1007/s41315-024-00378-3.
9. Hamidoğlu A, Elghirani A, Gül Ö.M, Kadry S. A survey of hybrid energy-aware and decentralized game-theoretic approaches in intelligent multi-robot task allocation. Comput Mater Contin. 2026;87(3):5. doi:10.32604/cmc.2026.077060.
10. Yang L, Li P, Wang T, Miao J, Tian J, Chen C, et al. Multi-area collision-free path planning and efficient task scheduling optimization for autonomous agricultural robots. Sci Rep. 2024;14:18347. doi:10.1038/s41598-024-69265-y.
11. Khakzad N. A methodology based on Dijkstra's algorithm and mathematical programming for optimal evacuation in process plants in the event of major tank fires. Reliab Eng Syst Saf. 2023;236:109291. doi:10.1016/j.ress.2023.109291.
12. Zhao FJ, Gao YF, Luo H, Sun XM. Robust MPC for obstacle avoidance with tightened dual constraints and reachability-guided planner. IEEE Trans Ind Electron. 2026. doi:10.1109/TIE.2026.3684187.
13. Wang Z, Zhao X, Zhang J, Yang N, Wang P, Tang J, et al. APF-CPP: an artificial potential field based multi-robot online coverage path planning approach. IEEE Robot Autom Lett. 2024;9(11):9199–206. doi:10.1109/lra.2024.3432351.
14. Hu L, Wei C, Yin L. Fuzzy A-star quantum multi-stage Q-learning artificial potential field for path planning of mobile robots. Eng Appl Artif Intell. 2025;141:109866. doi:10.1016/j.engappai.2024.109866.
15. Theurkauf A, Kottinger J, Ahmed N, Lahijanian M. Chance-constrained multi-robot motion planning under Gaussian uncertainties. IEEE Robot Autom Lett. 2024;9(1):835–42. doi:10.1109/LRA.2023.3337700.
16. Li J, Hu Y, Yang SX. A novel knowledge-based genetic algorithm for robot path planning in complex environments. IEEE Trans Evol Comput. 2025;29(2):375–89. doi:10.1109/tevc.2025.3534026.
17. Lin S, Wang J, Huang B, Kong X, Yang H. Bio particle swarm optimization and reinforcement learning algorithm for path planning of automated guided vehicles in dynamic industrial environments. Sci Rep. 2025;15:463. doi:10.1038/s41598-024-84821-2.
18. Li D, Wang L, Cai J, Ma K, Tan T. Research on terminal distance index-based multi-step ant colony optimization for mobile robot path planning. IEEE Trans Autom Sci Eng. 2023;20(4):2321–37. doi:10.1109/TASE.2022.3212428.
19. Lin X, Gao F, Bian W. A high-effective swarm intelligence-based multi-robot cooperation method for target searching in unknown hazardous environments. Expert Syst Appl. 2025;262:125609. doi:10.1016/j.eswa.2024.125609.
20. Abdulbaqi AS, Alsultan QH, Nejrs SM, Abd alla IH, Jabur YK, Jassim SA, et al. Design and fabrication of bio-inspired robotic systems for developed mobility and functionality in unstructured environments. Khwarizmia. 2023;2023:46–62. doi:10.70470/KHWARIZMIA/2023/005.
21. Nguyen TMB, Nguyen THH, Do TNH. Optimized metaheuristic strategies for addressing the multi-picker robot routing problem in 3D warehouse operations. Comput Mater Contin. 2025;84(3):5063–76. doi:10.32604/cmc.2025.064610.
22. Yang J, Yan F, Zhang J, Peng C. Hybrid chaos game and grey wolf optimization algorithms for UAV path planning. Appl Math Model. 2025;142:115979. doi:10.1016/j.apm.2025.115979.
23. El-kenawy EM, Khodadadi N, Mirjalili S, Abdelhamid AA, Eid MM, Ibrahim A. Greylag goose optimization: nature-inspired optimization algorithm. Expert Syst Appl. 2024;238:122147. doi:10.1016/j.eswa.2023.122147.

24. Shi K, Yang L, Wu Z, Jiang B, Gao Q. Multi-robot dynamic path planning with priority based on simulated annealing. J Frankl Inst. 2025;362(1):107396. doi:10.1016/j.jfranklin.2024.107396.
25. Yu Z, Si Z, Li X, Wang D, Song H. A novel hybrid particle swarm optimization algorithm for path planning of UAVs. IEEE Internet Things J. 2022;9(22):22547–58. doi:10.1109/jiot.2022.3182798.
26. Fei H, Du Z, Ma P, Liu R, Liu F, Wang M, et al. A multi-strategy particle swarm optimization algorithm for three-dimensional path planning of amphibious unmanned aerial vehicles. Eng Appl Artif Intell. 2026;167:113606. doi:10.1016/j.engappai.2025.113606.
27. Ji Y, Hu P. Wafer handing robotic arm vibration trajectory planning based on graylag goose optimization. Sensors. 2026;26(3):829. doi:10.3390/s26030829.
28. Kuznetsov AV, Schumann A, Rataj M. Continuous optimisation problem and game theory for multi-agent pathfinding. Int J Game Theory. 2024;53:1–41. doi:10.1007/s00182-023-00851-6.
29. Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95–International Conference on Neural Networks; 1995 Nov 27–Dec 1; Perth, WA, Australia. p. 1942–8. doi:10.1109/ICNN.1995.488968.
30. Chen D, Liu J, Li T, He J, Chen Y, Zhu W. Research on mobile robot path planning based on MSIAR-GWO algorithm. Sensors. 2025;25(3):892. doi:10.3390/s25030892.
31. Pu C, Jia Y, Zhang Z, Zhou H, Liu L, Qian P, et al. A fuzzy adaptive particle swarm optimization algorithm with Gaussian mutation for constrained engineering problems. Appl Soft Comput. 2025;185:113908. doi:10.1016/j.asoc.2025.113908.
32. Li M, Wang J, Deng S, Zhao Y, Li Y. Enhanced black widow optimization algorithm incorporating food sufficiency strategy and differential mutation strategy for feature selection of high-dimensional data. Expert Syst Appl. 2025;290:128506. doi:10.1016/j.eswa.2025.128506.