Journal of Research in Science and Engineering (JRSE)
ISSN: 1656-1996 Volume-7, Issue-8, August 2025
DOI: 10.53469/jrse.2025.07(08).07

A Hybrid Spiral-Biased Particle Swarm Optimization with Black-winged Kite Algorithm for Mobile Robot Path Planning

Lanyue Yang[1], Wei Fan[2]

[1,2]Chongqing University of Technology, Chongqing 400054, China
[1]yanglanyue0128@outlook.com, [2]19910039@cqut.edu.cn

Abstract: ***To enhance the global search and local optimization capabilities of swarm intelligence algorithms in robot path planning, this paper proposes an improved particle swarm optimization algorithm (HSB-PSO) that incorporates a dynamic strategy. This algorithm features three key innovations: First, a dynamic adaptive spiral strategy is introduced to adaptively guide particles toward the optimal solution by adjusting spiral parameters at different iteration stages, enhancing the particles' global exploration capability and convergence accuracy. Second, a probability-decayed black kite behavior mechanism is designed to simulate the perturbation behavior of black kites during predation, and a probabilistic control factor is introduced to dynamically adjust its influence, effectively improving search diversity and avoiding local optima. Finally, an elite-guided on-demand reverse learning strategy is combined to selectively perform reverse learning on elite particles based on the current state of the swarm, further enhancing local search and convergence speed. Simulation experiments demonstrate that the HSB-PSO algorithm demonstrates superior optimization capability and path quality in multiple typical path planning test scenarios, validating the effectiveness and practical value of the proposed strategy.***

Keywords: Particle Swarm Optimization Algorithm, Robot Path Planning, Dynamic Spiral Strategy, Reverse Learning.

1. Introduction

In recent years, with the rapid development of artificial intelligence and automation technologies, robots have been increasingly applied in fields such as warehouse logistics, intelligent manufacturing, and service industries. Path planning, as a core component of autonomous navigation systems, directly affects the operational efficiency and task performance of robots. Traditional path planning methods, such as A* and Dijkstra algorithms, exhibit certain limitations when dealing with complex dynamic environments or high-dimensional search spaces. As a result, they are gradually being replaced by swarm intelligence algorithms, with Particle Swarm Optimization (PSO) being a representative example.

PSO has been widely applied to 2D and 3D path planning problems due to its simple structure, strong parallelism, and excellent global search capability. However, standard PSO still suffers from issues such as premature convergence to local optima, slow convergence speed, and insufficient search accuracy in complex environments. To address these shortcomings, many researchers have proposed various improvements and enhancements to the PSO algorithm in recent years.

For instance, Shankar, M et al. (2022), in their paper *"A hybrid path planning approach combining artificial potential field and particle swarm optimization for mobile robot [1]"*, introduced an adaptive mutation mechanism and local search operator, which significantly improved the smoothness and stability of the generated paths. Pengfei Yu et al. (2024), in *"A Dynamic Path Planning Method for Mobile Robot Based on Virtual Potential Field and Virtual Impedance Model [2]"*, incorporated a multi-strategy cooperation mechanism, including crowding guidance and dynamic learning factor adjustment, effectively enhancing the algorithm's robustness and adaptability to complex environments. ChangSheng Huang et al. (2023), in *"APSO: An A*-PSO Hybrid Algorithm for Mobile Robot Path Planning [3]"*, proposed the use of environment-aware dynamic factors, enabling the algorithm to maintain strong optimization capability and fast response in dynamic obstacle environments. In addition, Qijiang Su et al. (2024), in *"Path planning for power inspection robot based on improved PSO algorithm and dynamic window approach [4]"*, introduced a mechanism combining elite memory and opposition-based learning to strengthen local exploitation and preserve historical optima.

These research outcomes demonstrate the strong potential of Particle Swarm Optimization in multi-strategy integration and dynamic optimization, providing valuable insights and solid support for future algorithm design.

To address the limitations of PSO in path planning applications, this paper proposes an improved Particle Swarm Optimization algorithm [5], **HSB-PSO**, which integrates multiple strategies. The algorithm introduces three key innovations from the perspectives of global exploration, local exploitation, and convergence control:

1) A dynamically adaptive spiral strategy guides particles to converge along the optimal trajectory;

2) A probabilistic decay-based Black Kite behavior mechanism enhances population perturbation ability;

3) An elite-guided on-demand opposition-based learning mechanism improves local search efficiency.

These strategies complement each other, significantly enhancing the algorithm's optimization capability and path feasibility in complex environments, and providing effective support for efficient path planning of robots in real-world scenarios [6].

2. Environment Modeling and Problem Description

In mobile robot path planning, the entire planning process is typically divided into two stages. The first stage is the environment modeling stage, where the robot's motion space model is constructed using sensor data or predefined information to represent obstacles, free space, and boundary information in the environment [7]. The second stage is the path search stage, in which intelligent algorithms or classical pathfinding methods are applied based on the established environment model to generate a feasible path from the start point to the target point while avoiding obstacles.

This paper adopts the Grid Method for environment modeling, based on the following considerations:

1) The grid method can intuitively represent obstacle regions of arbitrary shapes, offering strong adaptability to various environments;

2) Its unified data structure facilitates position representation and path feasibility evaluation within the Particle Swarm Optimization (PSO) algorithm;

3) The grid model supports dynamic updates of obstacle information, making it suitable for extensions in dynamic environments.

By using grid-based modeling, the continuous motion space is discretized, enabling the improved PSO algorithm to efficiently search and generate optimal paths that satisfy practical feasibility constraints. Specifically, the continuous 2D plane is discretized into a number of equally sized rectangular grid cells, forming a two-dimensional grid matrix $G \in R^{M \times N}$, where M and N represent the number of rows and columns [8], respectively. Each grid cell $G_{i,j}$ has a unique coordinate index indicating its position in the overall space. During the modeling process, all obstacle information in the environment is mapped onto the corresponding grid cells.
The following rules are defined:

- Free cell (traversable): Set to 0, indicating that the robot can move freely;

- Obstacle cell (non-traversable): Set to 1, indicating the presence of an obstacle in the cell, which the robot cannot pass through.

In the constructed 2D grid environment, to achieve optimal path planning from the start point to the destination, it is first necessary to establish the conversion relationship between grid indices and their spatial coordinates. Assuming the environment has a size of $N \times N$, each grid cell can be uniquely represented through a mapping between its index mm and its position coordinates (x, y). The mapping relationship is as follows:

$$m = (x - 1) \times N + y \tag{1}$$

$$y = \mathrm{mod}(m, N) \tag{2}$$

$$x = \mathrm{int}\left(\frac{m}{N}\right) + 1 \tag{3}$$

Here, m denotes the index of the current grid cell, while x and y represent the row and column coordinates of the robot in the 2D space. This mapping approach facilitates the linear representation of the grid environment and simplifies path encoding.

During the path planning process, the objective is to search for a path sequence P from the starting grid cell P_1 to the target grid cell P_n , such that the total path length is minimized while avoiding all obstacle cells. The corresponding optimization objective function can be expressed as:

$$minf(P) = \sum_{i=2}^{|P|} d(p_i, p_{i-1}) P \subset W, P \cap O = \emptyset \tag{4}$$

Here, P denotes the set of path points, W represents the set of all traversable grid cells, and O is the set of obstacle points. The term $d(p_i, p_{i-1})$ represents the Euclidean distance between two consecutive points in the path. The path planning must ensure that P and O are disjoint, thereby guaranteeing a collision-free and feasible path.

In this paper, the path planning problem is transformed into the process of optimizing the sequence of path points P. With the help of the grid index mapping, the Particle Swarm Optimization algorithm can directly encode and optimize the path within a one-dimensional search space. By defining an appropriate fitness function and constraint conditions, the algorithm ensures both path feasibility and the global objective of minimizing path length.

Ultimately, the proposed improved PSO algorithm can effectively compute the optimal path point set P_{best}, which satisfies obstacle avoidance constraints. A schematic diagram of the 2D grid map environment and the path planning problem is shown in **Figure 1** below:

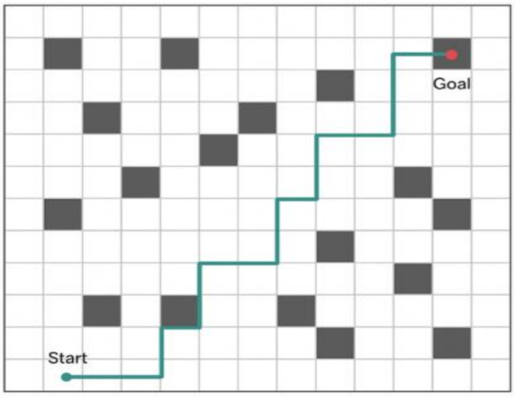

Figure 1: 2D raster map environment

3. Methodology

3.1 Particle Swarm Optimization Algorithm

Particle Swarm Optimization (PSO) is a population-based stochastic optimization algorithm inspired by swarm intelligence, originally proposed by Kennedy and Eberhart. The algorithm simulates the foraging behavior of bird flocks to search for the optimal solution. In PSO, each particle represents a potential solution in the search space, and all particles form a swarm. During the search process, each particle updates its position based on both its individual experience and the collective experience of the swarm, gradually approaching the global optimum.

In this study, the path planning problem is abstracted as an optimization problem that seeks the optimal path solution within a search space. It is assumed that there are N massless and dimensionless particles moving in a D dimensional search space, where each particle represents a candidate solution to the path planning problem. Through the simulation of social cognition and individual learning, each particle updates its position and velocity during the search process, thereby adjusting its search direction and step size to approximate the optimal path [11].

Let the position of the $\mathrm{i}-th$ particle be denoted as $x_i = [xi_1, xi_2, \dots, xi_D]$, where i=1,2, ... ,N, and each component xi_1 corresponds to a key waypoint in the path, represented by its coordinate or index in the map [9].

In standard PSO, each particle has both a position vector and a velocity vector, and its update rules in the search space are as follows:

$$v_i^{k+1} = w \cdot v_i^k + c_1 \cdot r_1 \cdot \left(p_i^k - x_i^k\right) + c_2 \cdot r_2 \cdot \left(g^k - x_i^k\right) \quad (5)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (6)$$

In each generation, particles update their states based on guidance from their personal best position P_i and the global best position g^k, continuously approaching better solutions [15]. Let the fitness function of the algorithm be denoted as Fitness, then the expressions for the personal best position and the global best position are as follows [10]:

$$p_i^{k+1} = \begin{cases} p_i^k, & ifFitness\left(x_i^{k+1}\right) > Fitness\left(p_i^k\right) \\ x_i^{k+1}, & ifFitnness\left(x_i^{k+1}\right) < Fitness\left(p_i^k\right) \end{cases} \quad (7)$$

$$g_i^k = min\{Fitness(p_1^k), Fitness(p_2^k), \cdots, Fitness(p_N^k)\} \quad (8)$$

3.2 Proposed Algorithm

3.2.1 Dynamic Adaptive Spiral Strategy

In standard particle swarm optimization algorithms, particle search behavior is typically controlled by fixed parameters, making it difficult to adapt to the search requirements of different optimization stages. This is particularly prone to problems in path planning, such as insufficient initial search and premature convergence in the later stages. To enhance the particle swarm's global exploration and local exploitation capabilities, this paper proposes a dynamic adaptive spiral strategy that improves both the perturbation factor and the guidance direction.

1) Adaptive spiral perturbation factor

During the particle spiral movement process, the disturbance amplitude is a key factor in controlling search accuracy. To this end, this paper designs a spiral disturbance parameter L and a contraction factor z that adjust dynamically with iterations, Here, L controls the direction and range of the disturbance, with the disturbance decreasing as iterations progress; z governs the nonlinear variation of the disturbance intensity, enabling the particle's movement to have better stability and adaptability. as detailed below:

$$L = (2 \cdot rand() - 1) \cdot \left(1 - \frac{t}{iter}\right) \quad (9)$$

$$z = \exp\left(k \cdot \cos\left(\pi \cdot \left(1 - \left(\frac{t}{iter}\right)^{0.5}\right)\right)\right) \quad (10)$$

Figure 1: Testing data- load current (amperes)

2) Hybrid directional guidance mechanism

To overcome the limitation of standard PSO relying solely on the global best $gBest$ for guidance, this paper proposes a hybrid directional guidance strategy that integrates local elite information from the Black Kite Algorithm (BKA). This strategy constructs a hybrid search direction $hybrid_{dir}$, enabling each particle to update its position by referring to both the global best position and a locally guiding individual BKA_{best}. The update formula is as follows:

$$\alpha = 0.3 \cdot \left(\frac{t}{iter}\right) \quad (11)$$

$$hybrid_{dir} = (1 - \alpha) \cdot (gBest - pos_i) + \alpha \cdot (BKA_{best} - pos_i) \quad (12)$$

$$pos_i^{new} = gBest + z \cdot L \cdot hybrid_{dir} \quad (13)$$

3.2.2 Black kite behavior mechanism with probability decay

Standard particle swarm optimization algorithms are prone to falling into local optima in the later stages. They lack the ability to mutate and adapt to perturbations, making it difficult to effectively escape from low-quality solutions. To enhance the algorithm's search perturbation and adaptability, this paper introduces the Black Kite Algorithm (BKA) and proposes an adaptive behavioral perturbation strategy with probability decay control to simulate the dynamic behavioral transitions of a predator during observation and attack.

1) Dynamic behavior selection probability

In traditional BKA behavior, the behavior selection probability is fixed, which makes it difficult to match the different stages of the optimization process. Therefore, this paper adopts a linearly decreasing behavior probability to simulate the biological behavior pattern of "early exploration, late development". The specific expression is as follows:

$$p = 0.9 \cdot \left(1 - \frac{t}{iter}\right) \quad (14)$$

2) Elite-guided Cauchy mutation

To further enhance the diversity of the search space and its ability to jump out of local search paths, this paper introduces a Cauchy distribution mutation mechanism into the Black Kite Algorithm (BKA) that incorporates the historical best information (pBest) of individual particle swarms. The key idea behind this mechanism is to leverage the search experience of individual particle swarms to guide Black Kite individuals toward more promising areas. Furthermore, the Cauchy distribution is introduced for asymmetric perturbation, thereby achieving stronger local jump capabilities.

A variant of the original Black Kite that approaches the leader based on random perturbations

$$X_i^{new} = X_i + \mathrm{cauchy} \cdot (X_i - X_{leader}) \quad (15)$$

This strategy can easily lead to variations that deviate from the true optimization direction. To this end, this paper designs the following improved model:

$$X_i^{new} = pbest + \text{cauchy} \cdot (X_{leader} - m \cdot X_i) \qquad (16)$$

3.2.3 Elite-oriented on-demand reverse learning

In standard reverse learning strategies, reverse computation is typically performed on all particles. While this can enhance local search capabilities, it introduces a large amount of redundant computation in high-dimensional or large population environments, reducing convergence efficiency. Therefore, this paper proposes an on-demand reverse learning strategy based on elite screening and triggering mechanisms to reduce computational complexity while improving search effectiveness.

In traditional reverse learning, reverse calculations are performed on all particles in each generation, resulting in severe computational redundancy. This paper introduces a periodic trigger mechanism and an individual screening strategy to enable reverse learning to be performed only on a subset of individuals in key generations. The specific trigger conditions are:

$$\text{if } \mathrm{mod}(t, 5) = 0 \qquad (17)$$

That is, a reverse learning operation is performed every 5 iterations; at the same time, only the 30% individuals with the lowest fitness ranking in the population are selected to perform the reverse operation:

$$i \in \mathrm{idx}[\mathrm{end} - \mathrm{round}(0.7 \cdot n) : \mathrm{end}] \qquad (18)$$

Where idx is the individual index sorted in ascending order of fitness. The reverse position is calculated as follows:

$$X_i^{new} = \frac{LB+UB}{2} + \frac{UB-LB}{2k} - \frac{X_i}{k} \qquad (19)$$

4. Numerical Experiments

To comprehensively evaluate the performance of the proposed HSB-PSO algorithm in path planning tasks, this paper conducted simulation experiments to assess its optimization capabilities and verify its feasibility in actual path planning. The experimental platform was MATLAB R2024a, the hardware environment was Windows 10, an Intel(R) Core (TM) i5-9300H 2.40GHz processor, and 8GB of RAM. The parameter settings are as follows:

Parameter name	Explanation	Value
N	Population size	60
PSO subpopulation ratio	Global Boot	60%(36 individuals)
BKA subpopulation ratio	Local disturbance	40%(24 individuals)
iter_{max}	Maximum number of iterations	100
w	Inertia Weight	0.7
c1, c2	Learning Factor	c1=c2=2.0
c3	Speed update factor	2.0
p	Initial probability of disturbance	0.9(Linear Decrease)
k	Dynamic perturbation step coefficient	1+(t/iter)2

4.1 Benchmark Function Testing

To evaluate the convergence and robustness of HSB-PSO in complex, high-dimensional search spaces, this paper selected 12 classic mathematical benchmark functions (F1–F12), covering a variety of characteristics such as unimodal, multimodal, noisy, and penalized. These functions are widely used to evaluate the global optimization ability and search stability of swarm intelligence optimization algorithms. See **Table 1** for details:

Table 1: Benchmarking Functions

Number	Name	Search Scope	feature
F1	Sphere	[-100, 100]	SinglePeak
F2	Schwefel 2.22	[-10,10]	SinglePeak
F3	Schwefel 1.2	[-100, 100]	SinglePeak
F4	Maximum	[-100,100]	SinglePeak
F5	Rosenbrock	[-30, 30]	Multi-peak
F6	Shifted Sphere	[-100, 100]	SinglePeak
F7	Quartic	[-1.28, 1.28]	SinglePeak
F8	Schwefel	[-500,500]	Multi-peak
F9	Rastrigin	[-5.12, 5.12]	Multi-peak
F10	Ackley	[-32, 32]	Multi-peak
F11	Griewank	[-600, 600]	Multi-peak
F12	Penalized 1	[-π, π]	Multi-peak

The experimental results are shown in **Figure 2**.

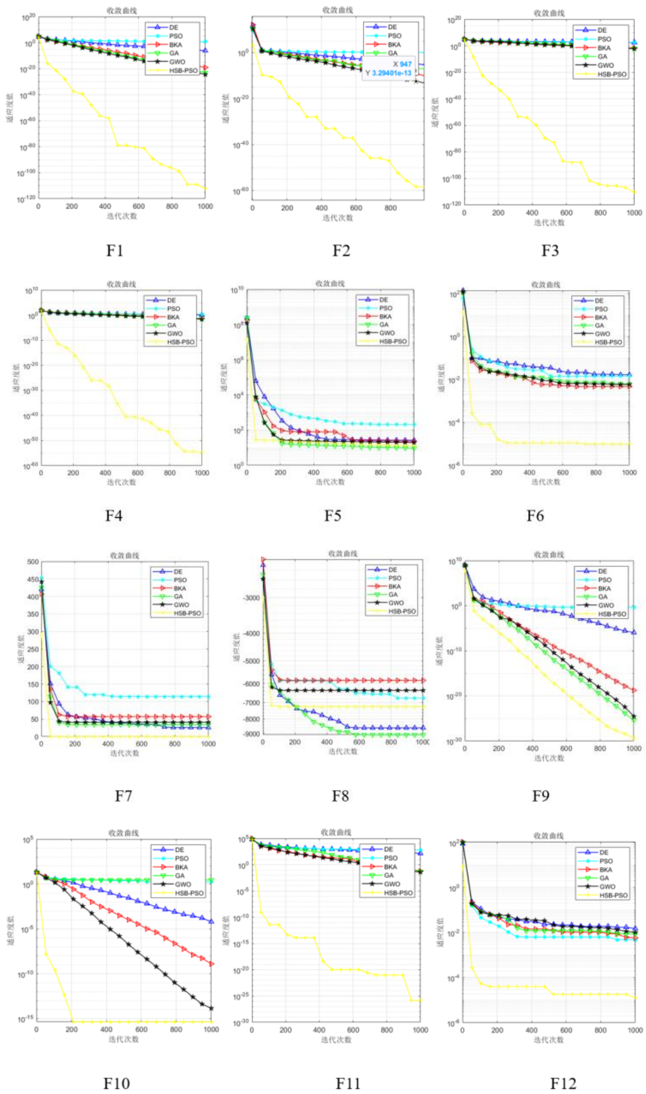

Figure 2: Test function experiment diagram

For each function tested, 30 independent experiments were run, and the average optimal value and standard deviation were calculated. The experimental results show that HSB-PSO can obtain better solutions on most test functions, demonstrating good global search capabilities and result stability. It has a clear advantage in complex multimodal functions such as Ackley and Griewank.

4.2 Path Planning Performance Test

To verify the feasibility and effectiveness of the HSB-PSO algorithm in practical path planning, two typical simulation scenarios are designed:

- **Terrain I (Simple Environment):** Sparse obstacles and a wide path selection space.
- **Terrain II (Complex Environment):** Dense obstacles with multiple feasible paths, posing a higher optimization challenge.

The comparative algorithms include the classical Genetic Algorithm (GA)[13], Grey Wolf Optimizer (GWO) [14], standard Particle Swarm Optimization (PSO) [16], and Black Kite Algorithm (BKA) [12]. All algorithms are run under identical conditions, each executed independently 30 times, and the average shortest path length is recorded as the performance metric.

The path planning results for Terrain I are shown in **Table 2**, and the corresponding planned paths are illustrated in **Figure 3**.

Table 2: Comparison of Terrain I results

algorithm	Shortest Path	Average Path
GA	29.2335	30.2554
GWO	28.3623	29.2364
PSO	29.4117	29.9658
BKA	28.5317	28.8598
HSB-PSO	**25.0192**	**25.3832**

The path planning diagram in a simple environment is as follows:

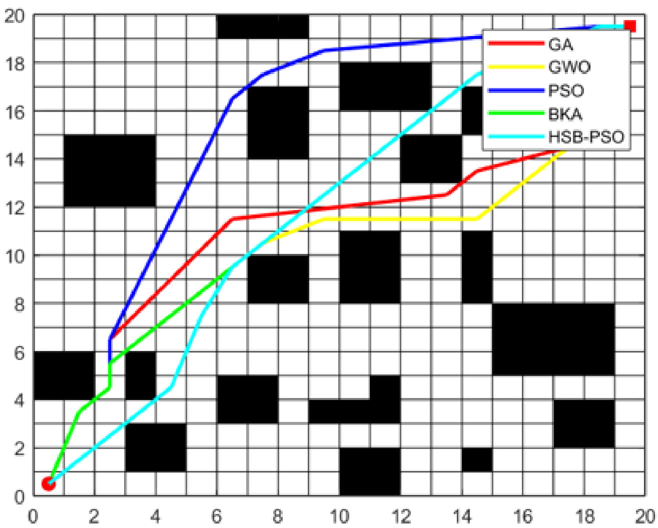

Figure 3: Terrain I Experiment Results

The path planning results for Terrain II are shown in **Table 3**, and the corresponding planned paths are illustrated in **Figure 4**.

Table 3: Comparison of Terrain II results

algorithm	Shortest Path	Average Path
GA	65.0071	66.2235
GWO	68.5753	68.9324
PSO	60.1931	61.3695
BKA	61.7016	62.5571
HSB-PSO	**54.1246**	**55.0326**

The path planning diagram in a complex environment is as follows:

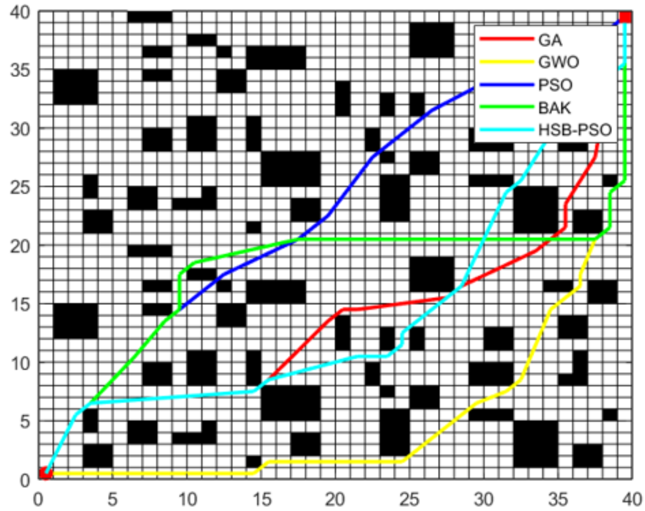

Figure 4: Terrain II Experiment Results

Experiments show that the proposed HSB-PSO algorithm achieves the shortest path length in both path planning scenarios, significantly outperforming other compared algorithms. This result validates the comprehensive optimization effect of HSB-PSO in global path search and local obstacle avoidance.

5. Conclusion

To address the challenges of traditional single-agent optimization algorithms in path planning, such as falling into local optimality and insufficient convergence accuracy, this paper proposes an improved particle swarm optimization algorithm, HSB-PSO, that integrates multiple strategies. By introducing an environment selection mechanism, this algorithm effectively leverages the complementary strengths of the particle swarm optimization algorithm and the Black Kite algorithm, achieving a parallel fusion of multi-agent algorithms and striking a good balance between global search and local exploitation.

By introducing a dynamic spiral guidance strategy, an adaptive probabilistic perturbation mechanism, and an elite-guided on-demand reverse learning method, the proposed HSB-PSO algorithm possesses strong search capabilities and the ability to escape local extremum traps. Experiments have demonstrated that HSB-PSO not only improves the convergence speed and optimization accuracy of path planning, but also exhibits good stability and robustness in complex environments. When applied to the path planning problem of mobile robots, the HSB-PSO algorithm can efficiently complete the path search from the starting point to the target point, avoid obstacles, and generate a smooth, feasible, and near-shortest path. Therefore, the proposed HSB-PSO algorithm provides a practical optimization solution for solving the mobile robot path planning problem and has promising application prospects in intelligent navigation, automated logistics, service robotics, and other fields.

Acknowledgment

This work was supported by the project titled **“Research on Visual AI Recognition Technology for Physical Defects in Microelectronic Components”** under Grant No. **2024ZYD0286.**

References

[1] Shankar, M., & Sushnigdha, G. (2022). “A hybrid path planning approach combining artificial potential field and particle swarm optimization for mobile robot”. IFAC-Papers on Line, 55(22), 242-247.

[2] Pengfei Yu, & Jinzhu Peng, (2024). “A Dynamic Path Planning Method for Mobile Robot Based on Virtual Potential Field and Virtual Impedance Model”. in Proc. 14th Asian Control Conf. (ASCC).

[3] Changsheng Huang, & Yanpu Zhao, (2023). “APSO: An A∗-PSO Hybrid Algorithm for Mobile Robot Path Planning”. IEEE Access.

[4] Qijiang Su, & Yanghua Cai, (2024). “Path planning for power inspection robot based on improved PSO algorithm and dynamic window approach,” in Proc. Conf. Information Science, Parallel and Distributed Systems (ISPDS)

[5] Xiaoyi Zhan, & Zihong Chen, (2023). “Path planning of service robot based on improved particle swarm optimization algorithm,” in Proc. Symp. Computer Engineering and Intelligent Communications (ISCEIC)

[6] MUHAMMADHARIS, & HAEWOON NAM. (2024). “Path Planning Optimization of Smart Vehicle with Fast Converging Distance-Dependent PSO Algorithm,” IEEE Open Journal of Intelligent Transportation Systems

[7] Ambuj, & Harsh Nagar. (2024). “Reinforcement learning and particle swarm optimization-based hybrid approach for mobile robot path planning,” Robotics and Autonomous Systems, vol. 174

[8] Sanli Fu, & Hengyi Huang. (2024). “Research on Robot Path Planning Technology by Integrating Particle Swarm Optimization Algorithm and DWA Algorithm,” in Proc. Conf. Control, Automation and Robotics (ICCAR)

[9] Luoyang Luo. (2024). “Robot path planning based on a hybrid whale optimization algorithm,” in Proc. Conf. Information Systems and Computer Aided Education (ICISCAE)

[10] LU WANG, & LULULIU. (2024). “Robot path planning based on generative learning particle swarm optimization,” IEEE Access, vol. 12

[11] ZHANG Wen-ning, & ZHOU Qing-le. (2023). “A particle swarm optimization algorithm with centroid opposition-based learning and simplex search,” Computer Engineering & Science

[12] Jun Wang,& Wen-chuan Wang. (2024). “Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems,” Artificial Intelligence Review

[13] S. Katoch, S. S. Chauhan and V. Kumar, “A review on genetic algorithm: past, present, and future,” Multimedia Tools and Applications, vol. 80, no. 6, pp. 8091–8126, Oct. 2020

[14] S. Mirjalili, & Zhe-peng Yue, “Grey Wolf Optimizer,” Advances in Engineering Software, vol. 69, pp. 46–61, Mar. 2014

[15] Tai-shan Lou, H. Li, and Y. Wang. (2025) “A hybrid multi-strategy SCSO algorithm for robot path planning,” Evolving Systems, vol. 16, pp. 54–70,

[16] J. Kennedy & R. C. Eberhart, “Particle swarm optimization,” in Proceedings of the 1995 IEEE International Conference on Neural Networks, Perth, Australia, Nov.