International Journal of Science and Engineering Applications
Volume 15-Issue 04, 71 – 74, 2026, ISSN:- 2319 - 7560
DOI: 10.7753/IJSEA1504.1012

Multi-Objective Path Planning for Intelligent Harvesting Robots Based on PSO and A* Algorithms

Kangqin Hu
School of Electronic Information and Electrical Engineering
Yangtze University
Jingzhou, China

Abstract: Under the background of the rapid development of global intelligent manufacturing, improving the efficiency of intelligent harvesting robots in agricultural operations has become a key research focus. Addressing the complex multi-target traversal problem faced by orchard harvesting robots, this study proposes a multi-target global path planner connecting Particle Swarm Optimization (PSO) and the A* algorithm. First, the A* algorithm is used to calculate the actual collision-free path distance between target points to construct a real cost matrix, overcoming the limitation in traditional Traveling Salesperson Problem (TSP) solutions that rely solely on Euclidean distance and ignore the impact of obstacles. Subsequently, an improved inertia-free discrete PSO algorithm is utilized to solve the multi-target path planning problem and generate the optimal target visit sequence. Finally, the A* algorithm is used to connect the sequence into a complete closed-loop working trajectory. Gazebo simulation tests demonstrate that the proposed algorithm converges efficiently in multi-target scenarios of various scales, significantly improving the global planning quality and computational efficiency of robots in complex orchard environments.

Keywords: Path Planning; Particle Swarm Optimization; A* Algorithm; ROS; Multi-objective Traversal

1. INTRODUCTION

With the rapid development of intelligent manufacturing and the economy, robotics technology has penetrated every aspect of human production and life. In agriculture, harvesting operations are generally characterized by strong seasonality, high labor intensity, and high labor costs. The application of intelligent harvesting robots is key to agricultural modernization, and their unstructured working environments require robots to possess efficient autonomous positioning and path planning capabilities.

In actual orchard operations, harvesting robots face a highly complex "multi-target point access" problem (similar to the Traveling Salesperson Problem, TSP). In existing TSP solving algorithms, Euclidean distance is commonly used to measure the relative positions between target points. However, in actual orchard environments, due to the presence of obstacles such as plants and walls, a simple geometric straight-line distance cannot reflect the true traversal cost. As observed in practice, the actual moving path between two points with the shortest Euclidean distance is often much longer when obstructed by obstacles. This discrepancy causes the operational sequences generated by traditional path planners to be sub-optimal in actual execution, resulting in significant invalid movements and energy consumption.

To address these issues, this paper proposes a global path planner fusing the A* algorithm and Discrete Particle Swarm Optimization (PSO). By replacing the Euclidean distance matrix with a real path cost matrix and combining it with an improved discrete PSO algorithm for sequence optimization, this approach provides a navigation solution for agricultural robots that balances global optimality and computational efficiency.

2. SYSTEM ARCHITECTURE AND MODELING

This system is built on the ROS Noetic operating system on Ubuntu 20.04, utilizing ROS's modular nodes and Topic communication mechanism to achieve real-time data interaction. The overall system architecture mainly includes an environmental perception and positioning module, and a global path planning module.

To enable the robot to efficiently acquire orchard environmental information, the system uses the Gmapping algorithm to construct a 2D grid map. This algorithm implements Simultaneous Localization and Mapping (SLAM) based on the Rao-Blackwellized Particle Filter (RBpf), effectively reducing the complexity of the SLAM problem. During navigation, the Adaptive Monte Carlo Localization (AMCL) algorithm is employed. By dynamically adjusting the number of particles and evaluating the alignment between LiDAR data and map obstacles in real-time, it achieves high-precision robot localization in dynamic environments. Accurate map and positioning data are transmitted to the customized multi-target point planner through the global cost map (global_costmap) under the Move_Base framework.

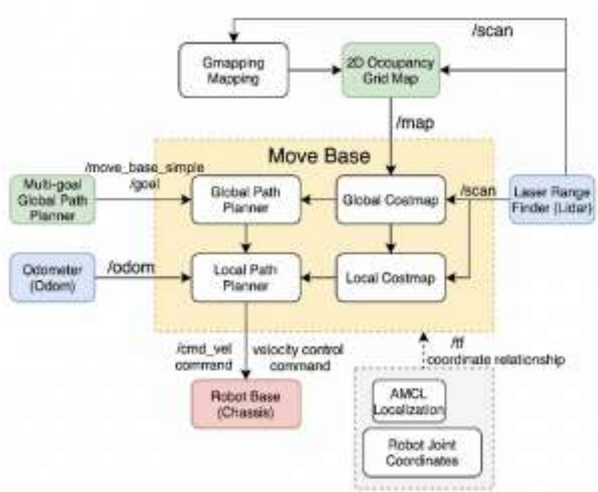

Figure 1. Architecture diagram of the intelligent harvesting robot path planning system

3. GLOBAL PATH PLANNING

Addressing the multi-target traversal problem, this paper designs a fusion planning strategy of "A* algorithm for distance measurement + PSO algorithm for optimization + A* algorithm for connection".

3.1 Real Cost Matrix Construction Based on the A* Algorithm

To avoid the evaluation distortion caused by Euclidean distance, the system first calls the A* algorithm to calculate the actual collision-free path length between any two target points. The A* algorithm effectively narrows the search scope through a heuristic function, avoiding the inefficiency caused by the blind search of Dijkstra's algorithm. Its evaluation function is expressed as:

$$F(n) = G(n) + H(n)$$

Where G(n) represents the actual path cost from the starting node to the current node n; H(n) is the heuristic function estimating the optimal path cost from the current node to the target node. This study uses Euclidean distance as the heuristic function:

$$H(n) = \sqrt{\left(X_n - X_{target}\right)^2} + \sqrt{\left(Y_n - Y_{target}\right)^2}$$

By calculating the actual path lengths between d target points pairwise, a d × d weight matrix with diagonal elements of 0 is constructed. This matrix converts the physical obstacle information in the environment into distance costs, providing accurate prior data for subsequent sequence optimization.

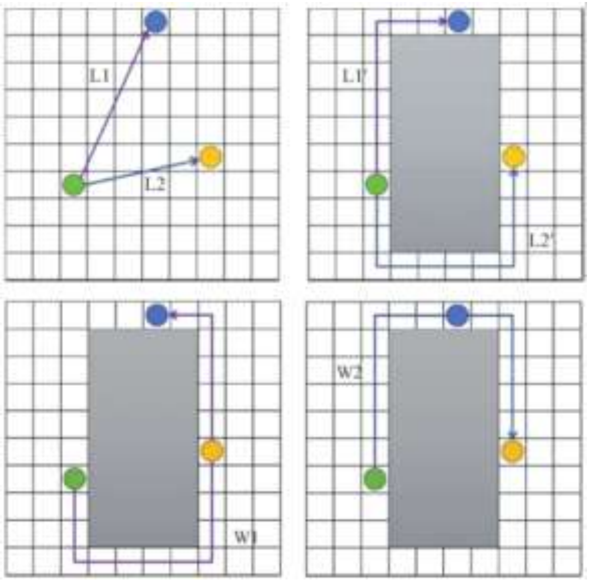

Figure 2. Distance evaluation differences in obstacle environments

3.2 Improved Discrete Particle Swarm Optimization (Discrete PSO)

After obtaining the weight matrix, the traversal order of the target points is treated as a discrete variable, and an improved Discrete Particle Swarm Optimization (Discrete PSO) algorithm is used to solve for the optimal access sequence. In traditional continuous PSO, the velocity update formula relies on inertia weight and continuous spatial step size; however, in multi-target path optimization, the particle's position is defined as an arrangement sequence of target points (e.g., $x_i = \{1,5,2,4,3\}$), and velocity is defined as a set of Swap operators.In a sequence optimization scenario, the superimposition of inertia based on historical swap operations will lead to irreversible path degradation. Therefore, this paper removes the inertia term from traditional PSO and reconstructs the velocity update formula as:

$$v(t+1) = c_1 r[p_{best}(t) - x_i(t)] \oplus c_2 r[g_{best}(t) - x_i(t)]$$

Where ⊕ indicates retaining the swap operation with a set probability. The individual learning factor c_1 and the social learning factor c_2 act as probability thresholds, guiding the particles towards the individual historical best (p_{best}) and the global best (g_{best}).

To compensate for the weakened global exploration ability after removing the inertia term, the algorithm introduces the crossover and mutation mechanisms of genetic algorithms: in each iteration, there is a 15% probability of performing random position crossover exchanges on the sequence, and a 10% probability of reversing sub-path fragments. This hybrid strategy not only retains the fast-convergence characteristic of the PSO algorithm but also effectively overcomes premature convergence and population homogenization by dynamically injecting path diversity. The final optimal sequence obtained is spliced again by the A* algorithm to form a visual and executable closed-loop working trajectory.

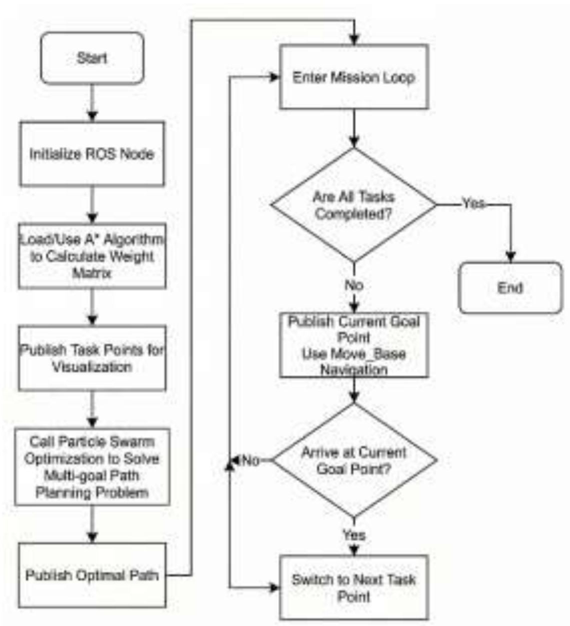

Figure 3. Execution flowchart of the multi-target global path planner

4. SIMULATION AND EXPERIMENTAL RESULTS

To verify the effectiveness of the algorithm, this study is based on the Gazebo 3D simulation platform carrying a TurtleBot3 Waffle robot model, and a complex orchard environment measuring 42m × 46m was independently constructed using Blender. Walls and orchard arrays were added to test the algorithm's optimization capability under complex obstacle conditions.

4.1 Global Path Planning Performance Evaluation

Targeting operational demands of different scales in the orchard, test scenarios with 10, 30, and 50 target points (excluding the starting point) were set up. Each set of scenarios

International Journal of Science and Engineering Applications
Volume 15-Issue 04, 71 – 74, 2026, ISSN:- 2319 - 7560
DOI: 10.7753/IJSEA1504.1012

underwent 10 independent experiments, and the average values were taken. Using 30 target points as an example, we compared the proposed "PSO+A*" fusion algorithm with the conventional Greedy Algorithm (which selects the point with the closest local Euclidean distance each time):

Table 1. Comparison of PSO algorithm and Greedy algorithm under 30 target points

Algorithm Name	Number of Targets	Average Computation Time (s)	Average Path Length (m)
PSO Algorithm (PSO+A*)	30	3.509	189.87
Greedy Algorithm	30	0.001	207.24

As shown by the data in Table 1 and Figures 4 and 5, although the Greedy Algorithm has an extremely short computation time, it is highly prone to falling into local optima due to a lack of global coordination, resulting in paths with numerous backtracks and redundancies. The proposed PSO fusion algorithm, within an allowable computation time overhead (approx. 3.5 seconds), successfully avoided obstacle traps and shortened the overall path length by approximately 8.4% (nearly 17.3 meters), demonstrating extremely high global optimization quality.

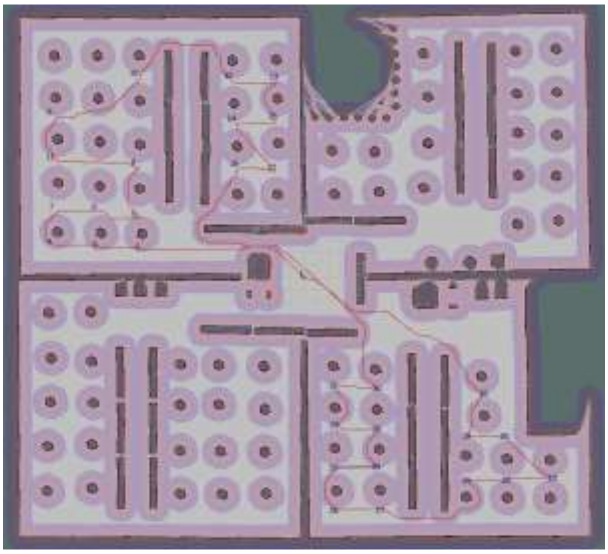

Figure 4. Global optimal path planned based on the PSO algorithm under 30 target points

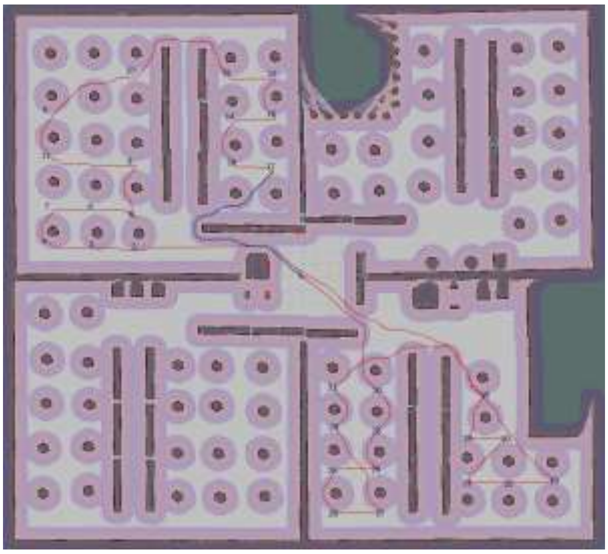

Figure 4. Global optimal path planned based on the PSO algorithm under 30 target points

4.2 Algorithm Convergence Analysis

To further verify the stability of the PSO algorithm, the variation curve of the algorithm's optimal path length with the number of iterations was recorded. In all tests with 10, 30, and 50 target points, the algorithm demonstrated excellent convergence characteristics. Taking 30 target points as an example, in the initial iteration stage, due to the effect of the crossover and mutation mechanism, the search space expanded rapidly, and the path length showed a step-like significant decline; entering the middle stage, the algorithm quickly converged to the optimal solution area; in the later stage, it stabilized and fluctuated near the minimum value without diverging. This fully demonstrates that the discrete PSO algorithm, which removes the inertia term and introduces a mutation mechanism, achieves a good balance between solution quality and computational efficiency.

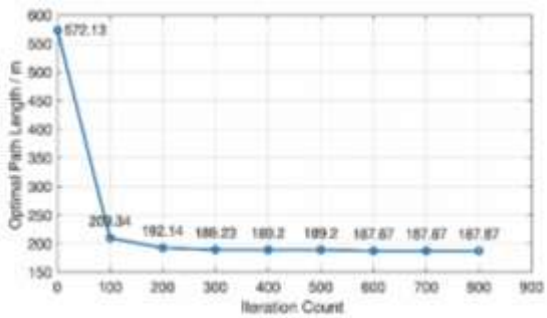

Figure 6. Iteration convergence curve of the PSO algorithm under 30 target points

5. CONCLUSION

Targeting the multi-target path planning problem faced by intelligent harvesting robots in complex orchard environments, this paper proposes a systematic solution fusing the A* algorithm and Discrete Particle Swarm Optimization (PSO). By calculating the actual obstacle avoidance distance between any two points using the A* algorithm to construct a real path cost matrix, the evaluation error brought by Euclidean distance in traditional TSP planning is successfully resolved. Utilizing an improved discrete PSO algorithm without an inertia term combined with a crossover mutation mechanism efficiently yields the optimal sequence for multi-target traversal. Simulation experiments show that the planner can stably converge in operational scenarios of various scales, significantly reducing the robot's invalid movements and improving the overall efficiency of agricultural harvesting operations.

Future research will attempt to combine deep learning technologies to pre-screen critical path points or introduce distributed computing frameworks to further accelerate the algorithm's solving process in ultra-large-scale target point scenarios (e.g., over 100). Meanwhile, advancing the engineering deployment and field verification of this algorithm in physical orchard environments will be the focus of the next step.

6. ACKNOWLEDGMENTS

Heartfelt thanks to the advisor for meticulous guidance on topic selection and research methodology, as well as to the fellow automation students for the technical support provided during the algorithm implementation and experimental validation phases. The successful completion of this design is also inseparable from the help of the open-source ROS ecosystem.

7. REFERENCES

[1] GIRSHICK R. Fast R-CNN[C]//Proceedings of the IEEE International Conference on Computer Vision. Piscataway: IEEE, 2015: 1440-1448.

[2] VIOLA P, JONES M. Rapid object detection using a boosted cascade of simple features[C]//Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2001: 511-518.

[3] WARANUSAST R, BUNDON N, TIMMONG V, et al. Machine vision techniques for motorcycle safety helmet detection[C]//Proceedings of 2013 28th International Conference on Image and Vision Computing New Zealand. Piscataway: IEEE, 2013: 35-40.

[4] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]//Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2015: 91-99.

[5] GIRSHICK R. Fast R-CNN[C]//Proceedings of 2015 IEEE International Conference on Computer Vision. Piscataway: IEEE, 2015: 1440-1448.

[6] Guruji A K, Agarwal H, Parsediya D K. Time-efficient A* Algorithm for Robot Path Planning[J]. Procedia Technology, 2016,23:144-149.