

Article

Self-Adaptive Ant Colony Optimization with Bidirectional Updating for Robot Path Planning

Yixuan Zhang , Shaoxin Sun , Yin Wang and Yiyang Yuan

School of Automation, Chongqing University, Chongqing 400044, China; 20221177@stu.cqu.edu.cn (Y.Z.); 20221399@stu.cqu.edu.cn (Y.W.); 20225421@stu.cqu.edu.cn (Y.Y.)

* Correspondence: sunshaoxin@cqu.edu.cn

Abstract

Mobile robot path planning using Ant Colony Optimization (ACO) has the disadvantages of slow convergence, local optima, and unsmooth paths because of fixed heuristics and constant pheromone updating. In this paper, Self-Adaptive Risk-Aware Bidirectional updating ACO (SAR-BACO) is proposed with three improvements: (1) composite heuristic incorporating target attraction, obstacle repulsion and direction consistency to minimize early blind searching; (2) dynamic pheromone updating based on solution quality and number of iterations to balance exploration and exploitation; (3) triangular pruning to remove redundant turning points and become smoother. Theoretical analysis verifies convergence. Our experimental results on grids up to 50×50 demonstrate that SAR-BACO performs much better than classical and heuristic-improved algorithms with respect to the length of a path, convergence rate, smoothness and efficiency. Using SAR-BACO on a 50×50 map, the path lengths, convergence iterations and turning points decreased by 60.68%, 48.96%, and 96.00% respectively compared to Basic ACO (after triangular pruning, values averaged over 50 runs). The framework provides a generalizable solution to autonomous navigation with the need to consider both search efficiency and path executability.

Keywords: ant colony optimization; path planning; adaptive pheromone update; heuristic information; triangular pruning

1. Introduction

Ant Colony Optimization (ACO) is a metaheuristic that was pioneered by Dorigo et al. [1]. Using positive feedback, parallel searches and distributed computing, ACO has proven to be very adaptable and resilient in combinatorial optimization, robotic path planning and task scheduling [2,3]. Global path planning in mobile robots uses ACO, which successively optimizes feasible paths through the process of pheromone-based learning without using an explicit model of the environment, making it a conventional optimization technique when using grids [4,5].

Though theoretically mature, applications of classical ACO to mobile robot path planning face significant challenges in which real-time constraints, environmental dynamics, and multi-objective requirements dominate. It can be noted that these drawbacks affect the effectiveness of deployment and the quality of the path, and they are manifested in three main factors.

The first factor is that the heuristic information design has structural faults causing search blindness at the initial stage. Traditional ACO is based on Euclidean or Manhattan distances only as a form of heuristic guidance [6]. Although it offers fundamental



Academic Editor: Yutaka Ishibashi

Received: 20 February 2026

Revised: 11 March 2026

Accepted: 12 March 2026

Published: 17 March 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

directional signals, this one-dimensional measure ignores the distribution of obstacles, the continuity of a path, and the dynamics of the target attraction. Initially in cluttered conditions, ants show a high level of random wandering, and convergence is much lower compared to what is required by real-time navigation needs [7]. Though certain works have incorporated distance gain coefficients or local obstacle avoidance terms [8,9], such adaptations are mostly empirical tuning of parameters. Their failure to obtain deep coupling between global goal attraction and local environment repulsion restricts the representational capabilities of heuristic information.

The second factor is an evaporation-and-update mechanism of static pheromones that does not allow the dynamic equilibrium between exploration and exploitation to be reached. The coefficients of evaporation ρ and deposition rules simultaneously define the memory persistence of the algorithm and its willingness to new solutions [10]. The use of fixed parameters in classical ACO results in two critical problems: the over-evaporation problem leads to a lack of sufficient accumulation of pheromones on high-quality routes, which traps the algorithm in local optima; and the under-evaporation problem causes the exponential increase in the number of pheromones, with the subsequent rapid deterioration of population diversity, causing premature convergence [11]. According to recent investigations adaptive evaporation strategies depending on the iteration phases [12] or the k-best elite retention rules [13]. Nonetheless, they are either manual threshold-based or optimized only at particular maps structures. They do not all contain a closed-loop adaptive system that senses the convergence state of populations and their distribution of path quality and that adjusts the evaporative and deposit rates accordingly.

Finally, the third factor is the uncoupling between the generation of paths and the post-optimization results in paths with unwanted turning points and low smoothness. A set of raw ACO paths is a collection of discrete grid nodes. These paths are limited by neighborhood connectivity rules and the local view on the basis of heuristic information so that there exist redundant turns, concave enclosures, or jagged detours [14]. Even though these paths meet the traversability constraints in discrete grid space, they do not comply with the kinematic constraints of nonholonomic mobile robots, which cause frequent changes in velocity, higher energy expenditure and reduced control stability. One useful geometric post-processing algorithm that is lightweight is triangular pruning, which has been successful in reducing redundant nodes and reducing path length [15,16].

More recent attempts at improving ACO in path planning are angular guidance factors and adaptive pseudo-random transitions rules [17], dynamic pheromone evaporation coefficients based on bacterial foraging step-size modification [18], and incorporating artificial potential field functions into heuristic information and triangularly pruning to obtain smoothed-out final paths [19]. Although these works contribute to ACO performance improvements in terms of different aspects, there is no unified framework of literature that systematically incorporates: (1) global–local collaboration in modeling of heuristic information; (2) dynamic coupling of the regulation of both pheromone evaporation and deposition; (3) improvement in feedback via post-processing of paths and successive searching.

As a solution to these gaps, the present paper introduces Self-Adapting Risk-aware Bidirectional updating Ant Colony Optimization (SAR-BACO). SAR-BACO addresses the issue of the performance constraints of conventional ACO in complex settings by systematically combining three essential mechanisms. The three major contributions are:

1. A global–local cooperative heuristic information model. The proposed study substitutes the conventional unidimensional distance-based heuristic with a combined assessment scheme that takes into consideration the global target gravitational fields, the local obstacle repulsive fields and the path directional constancy. Ants are capable of sensing both long-distance goal-driven benefits and short-distance environmental

constraint costs during state transitions, which greatly minimizes the blind randomness in the initial search stage and creates a structural basis for fast convergence.

2. Population-state-aware dynamic pheromone update strategy. It is suggested that an adaptive mechanism that combines the assessment of the solution quality with iterative improvement should be implemented. The pheromone evaporation coefficient is dynamically changed as a factor of the deviation between the existing best route and the mean route, actively decreasing evaporation in order to maintain the potential of exploration when population diversity reduces. Post-pruning path optimization results in asymmetrical rewards in the intensity of pheromone deposition, which leads to the reinforcement of high-quality solutions and the suppression of low-quality solutions. The strategy creates a dynamical equilibrium between global exploration and local exploitation, greatly improving the chances of getting out of local optima traps.
3. A path optimization technique based on triangular inequalities. Following the successful use of triangular pruning in reducing paths to smoother ones [20], this paper iteratively deletes excess turning points that do not satisfy the triangle inequality of the optimal path on a global scale. This procedure is an effective post-processing, and it immediately contributes to minimizing physical path length and greatly reduces the number of turning points without additional search cost. It produces smoother paths that are more closely aligned to robotic kinematics. The pruning optimization does not depend on the ant colony iteration procedure, thereby maintaining the simplicity and real-time nature of the search structure as well as attaining the qualitative jump between passable paths and efficiently executable paths.

Our comparative experiments on raster maps with different scales and complexities were systematically organized to show that SAR-BACO has substantial benefits within four main metrics: optimum path length, convergence iterations, algorithmic robustness, and smoothness of the path. In a complex map of 50×50 , in comparison to the baseline ACO the path length decreased by more than 60% and the convergence rate increased by over 96%. Against the literature algorithms using similar optimized heuristic information, SAR-BACO had more than 20% better path optimization rates and converges at rates more than 7% faster. These findings conclusively show that systematic incorporation of heuristic information reconstruction, dynamic pheromone coordination, and pruning feedback loops can provide overall performance much higher than the mere addition of one-dimensional improvement.

The planned SAR-BACO system offers a broad approach to mobile robot worldwide path planning, with an emphasis on search efficiency, the quality of the path, and environmental suitability. The development of the algorithm to support dynamic obstacles and online re-planning tasks in partly known settings will be continued in future work to prove that the algorithm can successfully perform real-time computations and has control compatibility with real-life robots.

Related Works and Research Gap

Recent studies have proposed various improvements to Ant Colony Optimization (ACO) for mobile robot path planning. The existing research mainly focuses on three directions: heuristic enhancement, adaptive pheromone regulation, and path post-processing.

Several works have improved heuristic information to guide ants toward more feasible paths. For example, Liu et al. [7] introduced an improved heuristic mechanism to incorporate environmental information into path selection. Wu et al. [8] and Chen et al. [9] further integrated obstacle-related factors to improve search efficiency in complex environments. Although these methods enhance local guidance, most of them rely primarily on empirical weighting of distance and obstacle terms, which may not fully capture the joint influence of goal attraction, obstacle distribution, and directional consistency.

Another research direction involves adaptive pheromone updating mechanisms. Approaches such as entropy-based convergence regulation [12] and swarm-inspired pheromone adaptation [18] aim to improve convergence stability and avoid premature stagnation. However, these methods mainly emphasize positive reinforcement of good solutions, while the explicit suppression of inferior paths is less explored.

In addition, several studies have investigated path-smoothing techniques, including trimming or triangular pruning [14,16], to reduce redundant turning points in grid-based paths. Nevertheless, these techniques are typically applied as post-processing steps and are not integrated with the optimization process of the search algorithm.

Therefore, the existing ACO-based path planning methods often improve the algorithm from a single perspective, lacking a unified framework that simultaneously integrates heuristic guidance, adaptive pheromone regulation, and path optimization. To address this limitation, this paper proposes SAR-BACO, which combines a composite heuristic model, a bidirectional pheromone updating strategy, and triangular pruning to improve convergence efficiency and path executability. Table 1 summarizes the comparison of representative ACO-based path planning methods.

Table 1. Comparison of representative ACO-based path planning methods.

Method	Heuristic Improvement	Adaptive Pheromone	Path Smoothing
Classical ACO [1]	Distance-based	No	No
Liu et al. [7]	Improved heuristic	No	No
Wu et al. [8]	Obstacle-aware heuristic	Partial	No
Wang & Li [14]	Basic heuristic	Limited	Path trimming
Proposed SAR-BACO	Composite heuristic	Bidirectional update	Triangular pruning

2. Basic Algorithms

Ant Colony Optimization is a metaheuristic based on the foraging behavior of an ant colony. The idea behind it is to mimic indirect communication (stigmergy) with the help of deposition and perception of pheromones so that the whole colony can gradually find the optimal paths in solution space.

An optimization problem in the classical ACO framework is defined as finding an optimal graph of the construction. Every artificial ant (a separate autonomous agent) repeatedly builds the candidates solutions by moving through the graph. Two fundamental components make up the working mechanism of the algorithm: probabilistic path selection and dynamic pheromone updating.

2.1. State Transition Rules

In every stage of path forming, when an ant k resides at a given node i , it chooses the subsequent unvisited node j based on a probabilistic principle. More precisely, the probability with which the transition occurs $p_{ij}^k(t)$ depends on the joint determination of two factors: the level of pheromones $\tau_{ij}(t)$ on the edge and the quality of the heuristic information $\eta_{ij}(t)$, which indicates that the algorithm utilizes historical experience or collective knowledge and prior knowledge, or that it utilizes problem-specific characteristics in a synergistic fashion. The formulation will be as follows:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{s \in \text{allowed}_k} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta}, & j \in \text{allowed}_k \\ 0, & j \notin \text{allowed}_k \end{cases} \quad (1)$$

where allowed_k denotes the set of unvisited nodes that are presently accessible to the ant k ; α and β control how much the relative importance of the intensity of the pheromones and heuristic information can be. Heuristic information $\eta_{ij}(t)$ has been commonly stated to be reciprocal of the distance d_{ij} between nodes indicating a greedy tendency towards spatially close nodes.

2.2. Pheromone Renewal Mechanism

A global pheromone update is carried out at the end of an iteration when all have created viable paths in the original Ant System (AS). The update has two stages:

1. Pheromone evaporation: Natural dissipation is simulated with a constant decay factor that avoids premature convergence to suboptimal solutions.
2. The deposition of pheromones: All ants deposit pheromones upon the edges that they have covered and the amount of the deposited is proportional to the quality of their constructed path.

The pheromone update rule is:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (2)$$

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{if ant } k \text{ has traversed edge } (i,j) \text{ during this iteration} \\ 0, & \text{else} \end{cases} \quad (3)$$

where $\rho \in (0,1)$ is the coefficient of evaporation of the pheromone; Q is a certain quantity that represents the amount of pheromones emitted by one ant in each iteration; and L_k is the overall length of the trail created by the ant k . The shorter path lengths produce higher pheromone increases $\Delta\tau_{ij}^k$ which reinforce better quality paths over time.

2.3. Algorithm Process and Limitations

The canonical ACO algorithm works as follows: pheromone trails and parameters are initialized and then an iterative process consisting of path construction followed by global pheromones updates is repeated till a stop criterion is satisfied.

Although the Ant System was one of the first algorithms that used swarm intelligence to optimize a combinatorial problem, it shows severe shortcomings such as a high rate of convergence and a tendency to get stuck in local optima, especially when dealing with large-scale issues [20,21]. Such defects are due to an inherently balanced but overly conservative update mechanism, where all ants engage in pheromone deposition without any effective limits on how much pheromones can be deposited by each ant, causing lack of direction control and destroying the exploration–exploitation trade-off.

3. SAR-BACO Algorithm

3.1. Enhanced Heuristic Information Function

According to standard ACO, heuristic information η_{ij} is calculated entirely using local distance metrics or target attraction. Even though this mechanism can be used to give a crude guide, it has two serious shortcomings in cluttered, multifaceted environments, namely:

1. Exploration–exploitation imbalance: The fixed heuristic information is unable to change with respect to the iterative progress, which leads to inadequate early exploration and slow late convergence.

2. Lack of sufficient local environment perception: Heuristic knowledge is needed to consider both attraction of targets and risk of collision, in case of narrow passages with irregular obstacles. Only dependence on “approach target” cues causes ants to linger near obstacles or wander in locally complex zones.

To address these issues, we propose an “enhanced heuristic information model” incorporating “iterative adaptive weighting”, “directional consistency correction”, and “local obstacle density suppression”:

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \cdot \omega(t) \cdot \left(\frac{1}{d_{j,g}} \right)^{\gamma} \cdot \cos(\theta_j) \cdot \exp(-\lambda \rho_j) \quad (4)$$

where $\eta_{ij}(t)$ is the Euclidean distance from node i to node j ; $d_{j,g}$ is the distance from node j to target node g ; $\gamma > 0$ is the target attraction exponent; $\omega(t)$ is the iterative adaptive weighting factor; $\cos(\theta_j)$ is the directional consistency correction term; ρ_j is the local obstacle density metric around node j ; and $\lambda > 0$ is the obstacle density suppression intensity coefficient. The schematic diagram of the enhanced heuristic function is shown in Figure 1.

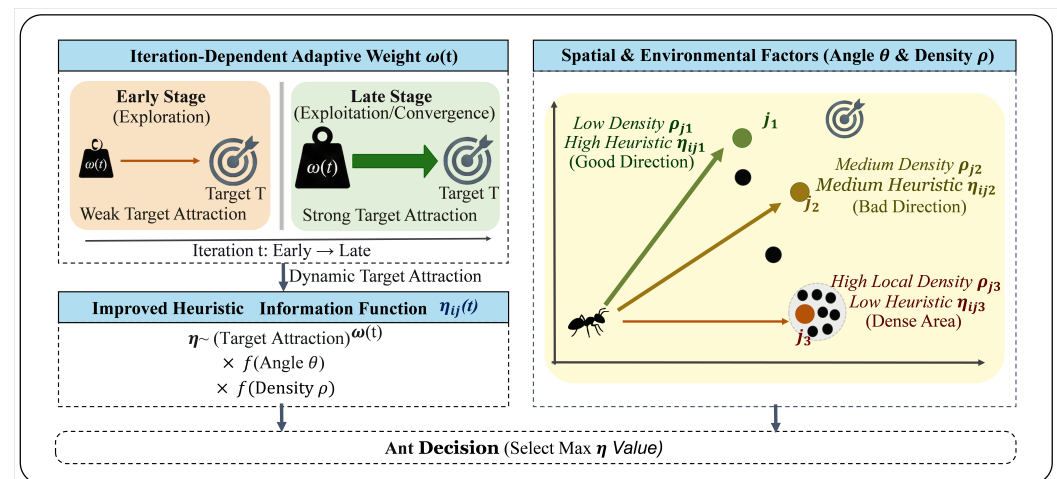


Figure 1. Schematic of the enhanced heuristic function in SAR-BACO.

3.1.1. Iterative Adaptive Weighting Factor $\omega(t)$

To enable phase-adaptive heuristic guidance, $\omega(t)$ follows a linearly increasing scheme:

$$\omega(t) = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \frac{t}{T} \quad (5)$$

where $\omega_{\min}$ and $\omega_{\max}$ are the lower and upper bounds, respectively, and T is the maximum iteration count. Such a design keeps enhanced random exploration in the initial iterations (small $\omega(t)$) and improves target guidance in the later iterations (large $\omega(t)$) reaching a steady state of switching over between the global search and fast convergence.

3.1.2. Directional Consistency Correction $\cos(\theta_j)$

In order to reduce the risk of a foolish attempt to hit a target in areas with obstacles, we propose introducing a factor into our correction term which is the angle between the target direction and the main threat direction of obstacles:

$$\theta_j = \arccos \left(\frac{\vec{v}_{j,g} \cdot \vec{v}_{j,o}}{|\vec{v}_{j,g}| \cdot |\vec{v}_{j,o}|} \right) \quad (6)$$

where $\vec{v}_{j,g}$ is the direction vector between the nodes j and target point g , and $\vec{v}_{j,o}$ is the weighted average direction of the obstacle points in a specified neighborhood around node j (computed as either the nearest obstacle direction or distance weighted average direction, in order to minimize computation time). A low value of θ_j (a high value of $\cos(\theta_j)$) means a high probability of collision when moving straight towards the target) coupled with $\exp(-\lambda\rho_j)$ lowers the heuristic level of such choices, making them less dangerous and more feasible directions of route.

3.1.3. Local Obstacle Density Indicator ρ_j

To capture the spatial complexity of the obstacle distributions, ρ_j is defined as:

$$\rho_j = \frac{1}{|O_j|} \sum_{k \in O_j} \exp(-\kappa d_{jk}) \quad (7)$$

where O_j denotes the collection of obstacle cells in a given neighborhood radius r centered on node j ; d_{jk} indicates the distance between node j and obstacle k ; and $\kappa > 0$ represents the coefficient of distance decay. This equation guarantees that ρ_j grows with an increasing number of obstacles and proximity to them. The result of this is a decrease in the suppression term $\exp(-\lambda\rho_j)$, which discourages movement to high-risk areas and minimizes unnecessary exploration of high-obstacle areas.

The given heuristic data provide a coupling scheme between target attractiveness (with $d_{j,g}$ and $\omega(t)$), safety-oriented feasibility (with $\cos\theta_j$ and ρ_j), and phase adaptivity (implemented with $\omega(t)$). The heuristic values increase as one nears the goal, has a steady direction and crosses areas of low obstacle density, whereas it decreases in areas of high obstacle density or places where collision is likely to occur. This adaptive modulation will steer the ant towards paths which are globally feasible, kinematically reasonable, and smoother.

3.2. Improved Pheromone Evaporation and Renewal Strategy

Traditional ACO uses the same evaporation and accretion rules, which causes two major problems: (1) initial randomness on particular paths creates too much attraction, which prematurely converges them to the nearby optimum; (2) continued accumulation of pheromones on suboptimal paths decreases the variety of exploration and endangers stability during the convergence process.

To overcome such limitations, we suggest a bidirectional update mechanism with elite path reinforcement and iteration-worst path suppression, as illustrated in Figure 2. While retaining the evaporation term to mitigate historical inertia, the proposed scheme applies differentiated reinforcement to the global-best path and iteration-best path, while imposing a deduction on the iteration-worst path. This design progressively amplifies pheromone discrimination between high-quality and low-quality paths, accelerating convergence and reducing local optima risk.

The improved pheromone update rule is:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}^+ - \Delta\tau_{ij}^- \quad (8)$$

where $\tau_{ij}(t)$ is the pheromone concentration on edge (i,j) at iteration t ; $\rho \in (0,1)$ is the evaporation coefficient; $\Delta\tau_{ij}^+$ is the positive increment for reinforcing high-quality paths; and $\Delta\tau_{ij}^-$ is the negative deduction for suppressing the iteration-worst path.

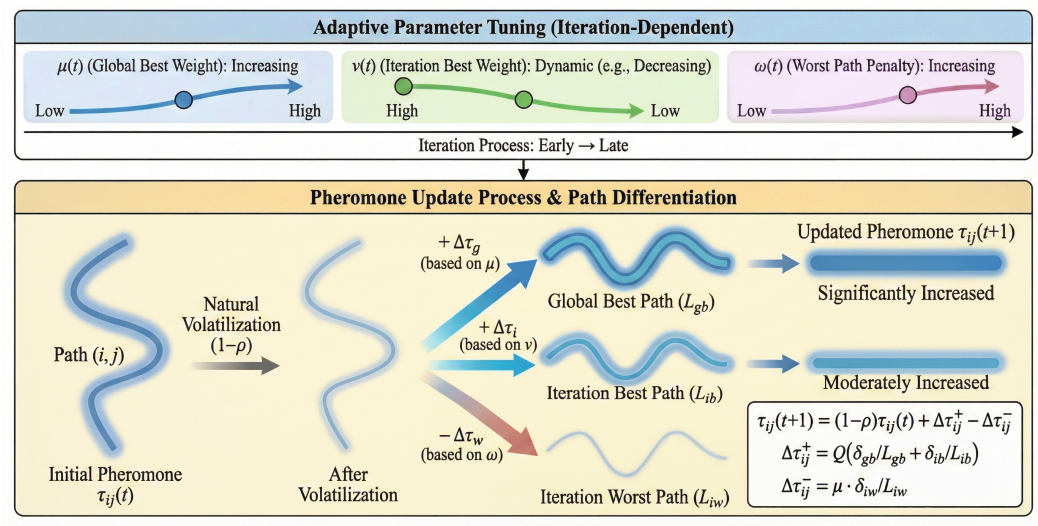


Figure 2. Pheromone update strategy of SAR-BACO.

3.2.1. Positive Increment Update

The positive increment is jointly contributed by the global-best path and the iteration-best path:

$$\Delta\tau_{ij}^+ = Q \left(\frac{\delta_{gb}}{L_{gb}} + \frac{\delta_{ib}}{L_{ib}} \right) \quad (9)$$

where Q is the pheromone release constant; L_{gb} is the length of the global-best path discovered so far; L_{ib} is the length of the iteration-best path in the current cycle; $\delta_{gb} = 1$ if edge (i, j) belongs to the global-best path, otherwise 0; $\delta_{ib} = 1$ if edge (i, j) belongs to the iteration-best path, otherwise 0. Such a design will also support both the historical optimal experience and the new iterations at the same time to guarantee a constant convergence direction when not entirely dependent on historical optima as a source of stagnated searching.

3.2.2. Negative Deduction Suppression

In order to reduce the interference of iteration-worst paths on the distribution of pheromones, an additional penalty term to the current iteration-worst path is imposed:

$$\Delta\tau_{ij}^- = \mu \frac{\delta_{iw}}{L_{iw}} \quad (10)$$

where $\mu > 0$ is the penalty coefficient; L_{iw} is the length of the worst path in the current iteration; $\delta_{iw} = 1$ if edge (i, j) is part of the worst path of the current iteration, and 0 otherwise. This approach will also diminish the misguiding nature of poor paths that makes it likely that inefficient paths are selected on repeated iterations, thus improving the efficiency of the searching and discrimination of the distributed pheromones.

3.2.3. Parameter Self-Adaptive Strategy

In order to achieve a good balance between exploration ability and convergence rate in the various stages of the algorithms, important weighting factors are updated as the process of iteration takes place. As further iterations occur, reinforcement weights on globally optimal paths are steadily increased over time, whereas suboptimal path weights are penalized even more heavily, leading to a faster increase in pheromone levels in high-value path areas. By contrast, early iterations allow maintaining exploratory behavior to avoid premature convergence. This is complemented by the $\omega(t)$ described in Section 3.1:

one of them modifies convergence behavior based on empirical reinforcement whereas the other changes path choice tendencies based on local decision guidance.

To sum up, the optimized pheromone update strategy is based on a mechanism of positive reinforcement + negative suppression + phase adjustment to improve the specificity and stability of pheromone distribution in dynamic environments, which leads to enhanced convergence rate, overall search performance, and reliability of path plans.

3.3. Convergence Analysis

To validate the stability and convergence of the improved algorithm, we provide a theoretical exposition on the boundedness and stable convergence of the pheromone update sequence under unified notation and reasonable assumptions. Note that ACO is a stochastic heuristic optimization method; its “convergence” implies that, under given evaporation and positive feedback reinforcement mechanisms, pheromone distribution tends toward stability and, with high probability, concentrates searching on the optimal or near-optimal path set.

The symbolic definitions and fundamental assumptions are as follows (Table 2): The grid map is represented as a graph $G = (V, E)$, where edges $(i, j) \in E$ correspond to feasible robot transitions.

Table 2. Symbolic definitions for convergence analysis.

Symbol	Significance
$\tau_{ij}(t) \geq 0$	Pheromone concentration of point (i, j) after iteration t
$\rho \in (0, 1)$	Pheromone decay coefficient
m	Ant population size
$L_k(t)$	Path length of ant k in iteration t
$L_{gb}(t), L_{ib}(t), L_{iw}(t)$	Global optimum, iteration best, and iteration worst path lengths up to iteration t respectively ¹

¹ Here, “iteration worst” refers to the longest path in the current iteration.

The improved pheromone update in this paper is

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}^+(t) - \Delta\tau_{ij}^-(t) \quad (11)$$

The positive increments and negative deductions are respectively:

$$\Delta\tau_{ij}^+(t) = Q \left(\frac{\delta_{gb}(i, j, t)}{L_{gb}(t)} + \frac{\delta_{ib}(i, j, t)}{L_{ib}(t)} \right), \quad \Delta\tau_{ij}^-(t) = \mu \frac{\delta_{iw}(i, j, t)}{L_{iw}(t)}; \quad (12)$$

$\delta_{gb}, \delta_{ib}, \delta_{iw} \in \{0, 1\}$ indicates whether the edge (i, j) belongs to the corresponding path.

Simultaneously, the transfer probability for ants adopts a heuristic information form:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{s \in \text{allowed}_k} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta} \quad (13)$$

where $\eta_{ij}(t)$ denotes the improved heuristic information function introduced herein, incorporating $\omega(t)$, ρ_j and $\cos \theta_j$. As $\eta_{ij}(t)$ participates solely in probability distribution without directly accumulating into τ , its impact on convergence manifests primarily through accelerating the sampling probability of superior paths.

To ensure the rigor of the proof, the following common and engineering-reasonable assumptions are provided:

1. Lower bound on path length: There exists $L_{\min} > 0$ such that any feasible path length $L \geq L_{\min}$.

2. Bounded parameters: $Q < \infty$, $\mu < \infty$, and at most a finite number of edges are incremented/decremented per iteration.
3. Non-negativity constraint: Pheromone concentration is truncated to ensure non-negativity: $\tau_{ij}(t+1) \leftarrow \max(\tau_{ij}(t+1), \tau_{\min})$, where $\tau_{\min} \geq 0$ (common engineering practice).

Figure 3 presents a flowchart summarizing the overall structure of the convergence proof for the proposed SAR-BACO algorithm.

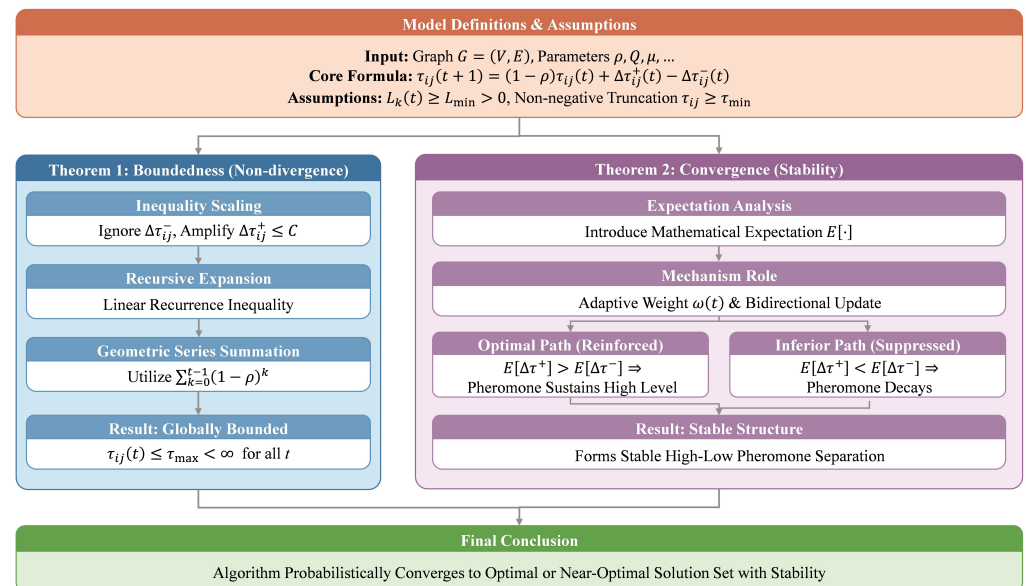


Figure 3. Flowchart of convergence proof for SAR-BACO.

Theorem 1 (Boundedness). Under Assumptions 1–3, for any edge $(i, j) \in E$ the sequence $\{\tau_{ij}(t)\}_{t=0}^{\infty}$ is bounded. That is, there exists a constant $\tau_{\max} < \infty$ such that $\forall t$, $\tau_{ij}(t) \leq \tau_{\max}$.

Proof. From Equation (11),

$$\tau_{ij}(t+1) \leq (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}^+(t) \quad (14)$$

Since $\delta_{gb}, \delta_{ib} \in \{0, 1\}$ and $L_{gb}(t), L_{ib}(t) \geq L_{\min}$,

$$\Delta\tau_{ij}^+(t) \leq Q \left(\frac{1}{L_{\min}} + \frac{1}{L_{\min}} \right) = \frac{2Q}{L_{\min}} \triangleq C \quad (15)$$

Thus,

$$\tau_{ij}(t+1) \leq (1-\rho)\tau_{ij}(t) + C \quad (16)$$

Expanding the recurrence:

$$\tau_{ij}(t) \leq (1-\rho)^t \tau_{ij}(0) + C \sum_{k=0}^{t-1} (1-\rho)^k = (1-\rho)^t \tau_{ij}(0) + \frac{C(1-(1-\rho)^t)}{\rho} \quad (17)$$

Since $(1-\rho)^t \rightarrow 0$,

$$\sup_t \tau_{ij}(t) \leq \tau_{ij}(0) + \frac{C}{\rho} = \tau_{ij}(0) + \frac{2Q}{\rho L_{\min}} \triangleq \tau_{\max} < \infty \quad (18)$$

□

Theorem 2 (Stable convergence in expectation). *Under the same assumptions, the pheromone distribution converges in expectation to a stable configuration that reinforces optimal paths and attenuates non-optimal paths.*

Proof. Let $\mathbb{E}[\cdot]$ denote expectation under random path construction at iteration t . From Equation (8),

$$\mathbb{E}[\tau_{ij}(t+1) \mid \tau(t)] = (1-\rho)\tau_{ij}(t) + \mathbb{E}[\Delta\tau_{ij}^+(t) - \Delta\tau_{ij}^-(t) \mid \tau(t)] \quad (19)$$

$\Delta\tau^+$ and $\Delta\tau^-$ are random variables determined by the set of optimal/suboptimal paths sampled. As the enhanced heuristic information $\eta_{ij}(t)$ strengthens guidance toward favorable regions via $\omega(t)$ and suppresses collision-prone areas via $\exp(-\lambda\rho_j)$ the sampling probability of the favorable paths increases progressively, while that of the poor paths decreases. Consequently, after a certain iteration stage, for any non-optimal edge $e \notin P^*$ (where P^* denotes the optimal/near-optimal path set),

$$\mathbb{E}[\Delta\tau_e^+(t)] < \mathbb{E}[\Delta\tau_e^-(t)] \quad \text{or} \quad \mathbb{E}[\Delta\tau_e^+(t) - \Delta\tau_e^-(t)] \approx 0, \quad (20)$$

implying

$$\mathbb{E}[\tau_e(t+1)] \lesssim (1-\rho)\mathbb{E}[\tau_e(t)] \quad (21)$$

Non-optimal edge pheromones exhibit an expected attenuation trend. For optimal edges $e \in P^*$, higher probability of entering the reinforcement set δ_{gb} or δ_{ib} and typical exclusion from the penalty set δ_{iw} yield

$$\mathbb{E}[\Delta\tau_e^+(t)] - \mathbb{E}[\Delta\tau_e^-(t)] > 0 \quad (22)$$

So pheromones on optimal edges remain at elevated levels and tend toward stability. Combined with Theorem 1's global boundedness, the pheromone distribution progressively forms a stable "high-low separation structure." In later iterations, the magnitude of change diminishes, converging to a stable point or a stable small neighborhood.

Relative to conventional ACO, the suggested mechanism provides more convergence by: (1) employing a phase-adaptive heuristic weighting $\omega(t)$, which adds target directing in subsequent iterations, increasing the sampling chance of better paths and speeding up the rate of pheromone concentration, (2) two-way updating coupled with a positive rewarding and negative subtraction, which decreases the likelihood of iteration-worst paths being re-selected, increases pheromone diversity and limits entrapment by local optima. These optimizations lead to quicker and more reliable convergence. $\square$

3.4. Algorithm Flowchart and Pseudocode

3.4.1. Enhanced Ant Colony Algorithm Flowchart

The entire process of SAR-BACO is outlined below (Figure 4). The algorithm uses a connected graph formed based on a raster map as the search space to be able to plan effectively in tight spaces and more complicated environments by means of improving its heuristic direction together with the two-way pheromone updates.

As shown in the flowchart in Figure 4, firstly, begin with parameter initialization, input the map $(G(V, E))$, start point (s) , and end point (g) . Initialize the number of ants (m) , maximum iteration count (T) , pheromone matrix $(\tau^{(0)})$, pheromone evaporation coefficient (ρ) , pheromone weight (α) , heuristic weight (β) , pheromone constant (Q) , penalty coefficient (μ) , and parameters $(\omega_{\min}, \omega_{\max}, \gamma, \lambda, \kappa)$ from the heuristic improvement. Initialize the global-best path (P_{gb}) and its length (L_{gb}) . Next, in the iterative loop $(t = 1, \dots, T)$, during iteration (t) , first compute the adaptive weight $\omega(t)$ based on the iteration process to dynamically balance "exploration and exploitation".

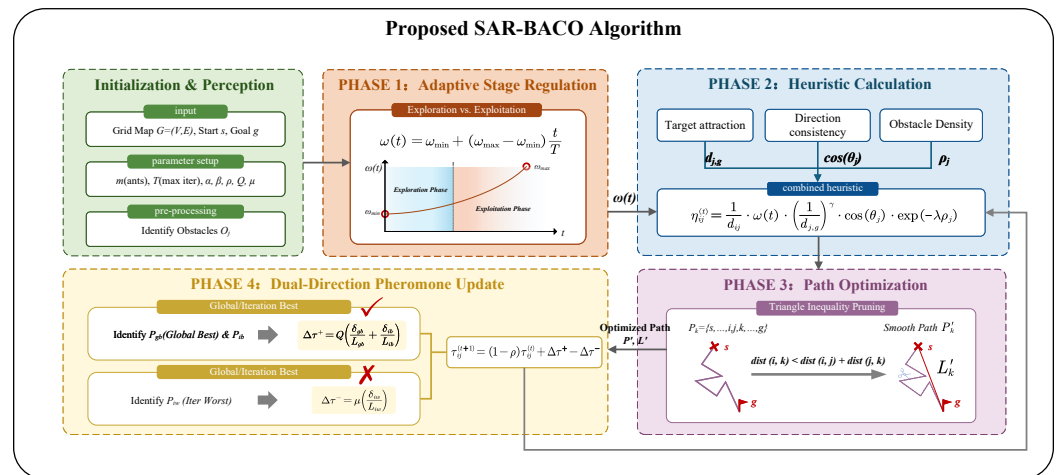


Figure 4. Overall flowchart of SAR-BACO.

For each ant ($k = 1, \dots, m$), construct a path from the starting point (s) to the destination (g) step by step: At the current node (i), generate the feasible set of adjacent nodes (allowed_k) (excluding obstacle nodes and previously visited nodes, or incorporating a backtracking mechanism). For each candidate node ($j \in \text{allowed}_k$), compute the enhanced heuristic information $\eta_{ij}(t)$, comprising distance terms, goal attraction terms, direction consistency terms, and local obstacle density suppression terms. The next node is selected according to the probability transition rule $p_{ij}^k(t) \propto [\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta$. If the ant reaches the destination, the complete path and its length are recorded. If no feasible nodes remain, backtracking/termination is triggered, and the path is marked as infeasible (engineering implementation recommends backtracking one step and resampling).

Path evaluation and superiority identification in this iteration: After all the ants complete path construction, the set of feasible paths for this iteration is statistically analyzed to select:

1. The optimal path (P_{ib}) and its length (L_{ib});
2. The iteration-worst path of this iteration (P_{iw}), length (L_{iw}). Compare these with the historical global optimum. If $L_{ib} < L_{gb}$, update the global optimum path (P_{gb}) and (L_{gb}).

Pheromone bidirectional updating evaporates all graph edges and differentially updates superior and iteration-worst paths.

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \quad (23)$$

1. Positive reinforcement: If the edge belongs to P_{gb} or P_{ib} , apply

$$\Delta\tau^+ = Q \left(\frac{\delta_{gb}}{L_{gb}} + \frac{\delta_{ib}}{L_{ib}} \right) \quad (24)$$

2. Negative deduction: If the edge belongs to P_{iw} , apply

$$\Delta\tau^- = \mu \left(\frac{\delta_{iw}}{L_{iw}} \right) \quad (25)$$

3. Apply non-negative truncation to τ when necessary (e.g., $\tau \leftarrow \max(\tau, \tau_{\min})$) to ensure numerical stability.

Termination criteria: If the maximum iteration count (T) is reached, or if the global optimal length change falls below the threshold (ϵ) for consecutive iterations, output the global-best path (P_{gb}) as the planning result and terminate; otherwise, proceed to the next iteration.

3.4.2. Improved Ant Colony Optimization Pseudocode

Algorithm 1 presents the pseudocode of the improved ant colony optimization (SAR-BACO), detailing the adaptive heuristic and pheromone update mechanisms.

Algorithm 1: Pseudocode of SAR-BACO

Input: Map G , start s , goal g , ants m , iterations T , parameters $\{\alpha, \beta, \rho, Q, \mu, \omega, \gamma, \lambda, \kappa\}$

Output: Global best path P_{gb} and length L_{gb}

```

1: Initialize  $\tau_{ij} = \tau_0$  for all  $(i, j) \in E$ ; Set  $P_{gb} = \emptyset, L_{gb} = \infty$ 
2: for  $t = 1$  to  $T$  do
3:    $\omega(t) = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \frac{t}{T}$ 
4:   for  $k = 1$  to  $m$  do
5:     Place ant  $k$  at  $s$ ;  $P_k = [s]; i = s$ 
6:     while  $i \neq g$  do
7:       Build neighbor set  $allowed_k(i)$ ; if empty then mark  $k$  fail; break
8:       for each  $j$  in  $allowed_k(i)$  do
9:         Compute obstacle density  $\rho_j$  and direction angle  $\theta_j$ 
10:         $\eta_{ij} = \frac{1}{d_{ij}} \omega(t) \left( \frac{1}{d_{jg}} \right)^\gamma \cos(\theta_j) \cdot \exp(-\lambda \rho_j)$ 
11:         $p_{ij} = \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in allowed_k(i)} [\tau_{il}]^\alpha \cdot [\eta_{il}]^\beta}$ 
12:      end for
13:      Sample next node  $j \sim p_{ij}$ ;  $P_k.append(j); i = j$ 
14:    end while
15:    if  $i = g$  then Compute length  $L_k$ ; Update  $P_{ib}, L_{ib}$  and  $P_{iw}, L_{iw}$ 
16:    end if
17:  end for
18:  if  $L_{ib} < L_{gb}$  then  $P_{gb} = P_{ib}; L_{gb} = L_{ib}$ 
19:  end if
20:  for each edge  $(i, j) \in E$  do
21:     $\tau_{ij} = (1 - \rho) \tau_{ij}$  ▷ Pheromone evaporation
22:    if  $(i, j) \in P_{gb}$  then  $\tau_{ij} = \tau_{ij} + \frac{Q}{L_{gb}}$  ▷ Global best reinforcement
23:    end if
24:    if  $(i, j) \in P_{ib}$  then  $\tau_{ij} = \tau_{ij} + \frac{Q}{L_{ib}}$  ▷ Iteration best reinforcement
25:    end if
26:    if  $(i, j) \in P_{iw}$  then  $\tau_{ij} = \tau_{ij} - \frac{\mu}{L_{iw}}$  ▷ Iteration worst punishment
27:    end if
28:     $\tau_{ij} = \max(\tau_{ij}, \tau_{\min})$  ▷ Non-negativity constraint
29:  end for
30:  if Convergence reached then break
31:  end if
32: end for
33: return  $P_{gb}, L_{gb}$ 

```

4. Experimental Results and Analysis

4.1. Experimental Design and Environment Setup

Comparative experiments were conducted to evaluate SAR-BACO against baseline ACO and representative heuristic-enhanced ACOs from the literature. Three grid map scales were tested: 20×20 , 30×30 , and 50×50 . All the algorithms were implemented in MATLAB R2024a on a 64-bit Windows 11 platform. The maps were modeled using a grid-based representation with randomly generated obstacles (black cells) covering 25–30% of the area; navigable areas were white cells. The start point was fixed at the top-left corner (grid index 1), and the target point at the bottom-right corner. All the algorithms shared unified baseline parameters to ensure fair comparison (Table 3).

Table 3. Unified parameter configuration for all algorithms.

Parameter	Symbol	Value
Number of ants	m	50
Maximum number of iterations	$N_{\max}$	100
Pheromone heuristic factor	α	1
Heuristic information factor	β	2
Pheromone evaporation coefficient (baseline)	ρ	0.8
Pheromone intensity constant	Q	10

4.1.1. Parameter Settings and Fairness of Comparison

To ensure a fair comparison, all the ACO-based methods shared the same core hyper-parameters and stopping criteria, including the number of ants ($m = 50$), the maximum iteration number ($N_{\max} = 100$), the pheromone/heuristic weights ($\alpha = 1$, $\beta = 2$), the evaporation rate ($\rho = 0.8$), and the pheromone constant ($Q = 10$), as summarized in Table 3. All the algorithms were implemented in the same MATLAB environment and executed on the same hardware/software platform.

In addition to the shared parameters, SAR-BACO introduces several extra coefficients for the proposed composite heuristic and bidirectional updating (Table 4), namely ($\omega_{\min}$, $\omega_{\max}$, γ , λ , κ) in Equations (4)–(7) and μ in Equation (10). These additional parameters were selected through preliminary tuning on separate validation maps (with the same obstacle density range) and then fixed for all the reported experiments to avoid test-set bias. The baseline methods followed their original recommended settings in the corresponding references and used the same shared parameters in Table 3. Therefore, the reported improvements mainly came from the proposed mechanism design rather than from advantageous parameter choices.

Table 4. Additional parameters used in SAR-BACO.

Parameter	Meaning	Value	How Chosen
$\omega_{\min}$	lower bound of $\omega(t)$	0.5	validation tuning
$\omega_{\max}$	upper bound of $\omega(t)$	2.0	validation tuning
γ	goal-attraction exponent	1.5	validation tuning
λ	obstacle-density suppression	3.0	validation tuning
κ	distance decay in ρ_j	1.0	validation tuning
μ	penalty coefficient	5.0	validation tuning

4.1.2. Runtime Measurement Protocol

To ensure a fair comparison of computational efficiency, all the algorithms were executed under identical stopping criteria and implementation conditions. Specifically,

each algorithm was run with the same maximum number of iterations ($N_{\max} = 100$) and the same ant population size ($m = 50$).

For consistency across the methods, all the algorithms were executed until reaching the predefined maximum iteration number ($N_{\max}$). Convergence-based early termination was not used in the runtime measurement in order to avoid bias introduced by different convergence behaviors.

The runtime reported corresponds to the total wall-clock time from algorithm initialization to final path output, including path construction, pheromone updating, and (for SAR-BACO) the triangular pruning post-processing step. All the algorithms were implemented using the same MATLAB code structure and executed on the same hardware platform to ensure comparable computational conditions.

4.1.3. Grid-Search Parameter Tuning Baseline

In addition to the proposed adaptive strategy, we included a conventional grid-search-based parameter tuning baseline to represent a commonly used manual tuning approach. Similar grid-search-based hyperparameter tuning strategies have also been adopted in data-driven optimization and diagnostic systems [22,23].

Specifically, grid search was applied to tune the baseline ACO parameters (ρ , α , β), while keeping the remaining settings identical to Table 3 ($m = 50$, $N_{\max} = 100$, $Q = 10$). The searched ranges were: $\alpha \in 0.5, 1, 1.5, 2$, $\beta \in 1, 2, 3$, and $\rho \in 0.6, 0.7, 0.8, 0.9$. For each parameter combination, we performed multiple independent runs on validation maps and selected the configuration that achieved the best average path length. The resulting tuned baseline is reported as “ACO (grid-search tuned)” in Tables 5–7 and serves as a reference for evaluating the advantage of SAR-BACO over static manual tuning.

4.2. Performance Evaluation Metrics

To quantitatively assess algorithm performance from multiple perspectives, four core metrics were employed:

1. Optimal path length: Euclidean distance of the shortest feasible path identified across all iterations.
2. Convergence iterations: The number of iterations needed to find the globally optimal path initially.
3. Count of direction changing points: The number of direction-changing nodes that lie on the optimum path; a smaller number means smoother paths, which are beneficial to mobile robots.
4. Average running time: The average time used per run (average of 50 independent runs) when initializing to end of the final path output is used to indicate the computation efficiency.

All results are reported as means over 50 independent runs.

4.3. 20 × 20 Grid Map

To begin with, it builds a map of a 20-order grid containing a random set of obstacles. Set the initial node to be grid cell number 1 with coordinates (0.5, 19.5) and the target node to be grid index 400 with coordinates (19.5, 0.5). Limit the obstacle density to the range of 25–30%.

In Figure 5, some examples of representative paths and convergence trajectories are depicted. Baseline ACO resulted in a tortuous path and contained many unnecessary twists and turns. Target attraction guidance was used to enhance the smoothness of the algorithm in [5,14]; however it still had local fluctuations. Taking advantage of better heuristics and adaptive pheromones updates, SAR-BACO obtained the shortest route with

a much straighter overall path. The convergence curves show that SAR-BACO converged, on average, after five iterations, which was considerably quicker compared to the algorithm in [5,14] (12 iterations) as well as the baseline ACO (60 iterations).

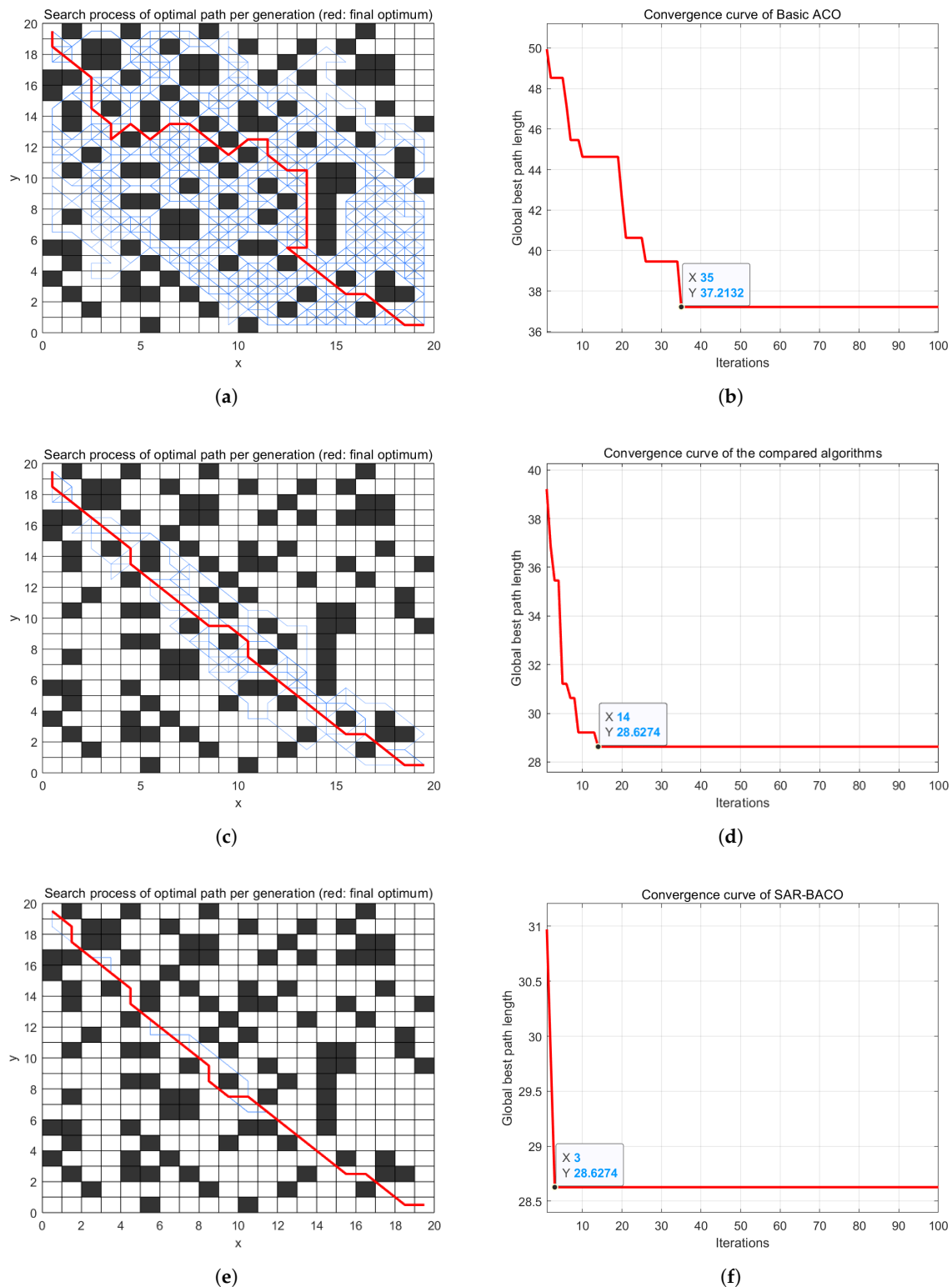


Figure 5. Experimental results on a 20 × 20 grid map: (a) Search process and optimal path of Basic ACO. (b) Convergence curve of Basic ACO. (c) Search process and optimal path of the algorithm in [5,14]. (d) Convergence curve of the algorithm in [5,14]. (e) Search process and optimal path of SAR-BACO. (f) Convergence curve of SAR-BACO. In subfigures (a,c,e), the blue lines represent the optimal paths explored in each generation, and the red line represents the final optimal path.

Figure 6 illustrates the effect of triangular pruning on SAR-BACO paths, where blue lines represent the paths after triangular pruning.

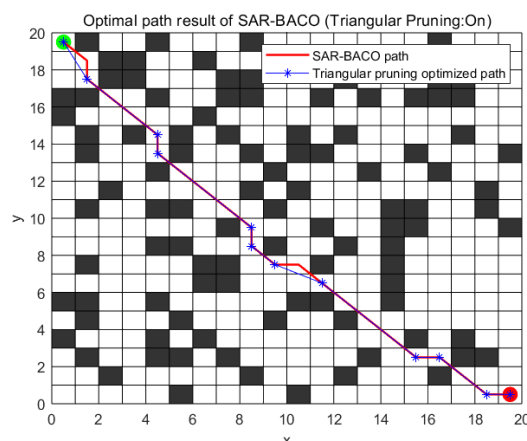


Figure 6. SAR-BACO after triangular pruning on a 20×20 grid map. The green and red dots denote the start and end points of the ant's movement, respectively.

SAR-BACO was executed 50 times with similar experimental settings. In Table 5 you can see that the values of the performance metrics are the average of the optimal path length, the convergence iteration count, the number of path turning points, and the runtime.

Table 5. Performance comparison of algorithms in a 20×20 environment.

Algorithm	Optimal Path Length	Iterations to Convergence	Number of Path Turning Points	Runtime (s)
Basic ACO	37.2132 ± 1.84	59.0 ± 10.9	21.0 ± 3.8	10.2533 ± 0.71
Algorithms in [5,14]	28.6274 ± 1.21	12.0 ± 3.5	11.0 ± 2.2	5.1235 ± 0.38
SAR-BACO (before pruning)	28.6274 ± 1.07	3.0 ± 0.8	11.0 ± 2.0	4.3745 ± 0.31
SAR-BACO (after pruning)	28.2711 ± 0.98	3.0 ± 0.8	9.0 ± 1.6	4.3745 ± 0.31

Prior to a pruning operation, SAR-BACO reduced the path length by 23.07%, the number of convergence iterations by 94.92%, and runtime by 57.34% when compared with the baseline ACO. Following triangular pruning, the path length improved again by 1.24%, and turning points decreased by 57.14% compared to the baseline ACO, confirming the efficacy of the pruning strategy in removing redundant nodes.

4.4. 30×30 Grid Map

Grid cell number 1, with coordinates (0.5, 29.5) was selected as the beginning node; grid cell number 900, with coordinates (29.5, 0.5) was selected as the final node, such that the obstacle density was kept at 25–30%.

With the rise of environmental complexity, the baseline ACO was associated with extremely high rates of path tortuosity and searching time. Although the [5,14] algorithms preserved directionality, they were vulnerable to local optima in the presence of complex obstacles. As shown in Figure 7, SAR-BACO did not pass through dense obstacles but followed a similar path to the optimum theoretical smooth path. The convergence curves reveal that SAR-BACO achieved high-quality paths after about 13 iterations, had an excellent convergence rate and performed well at global optimization.

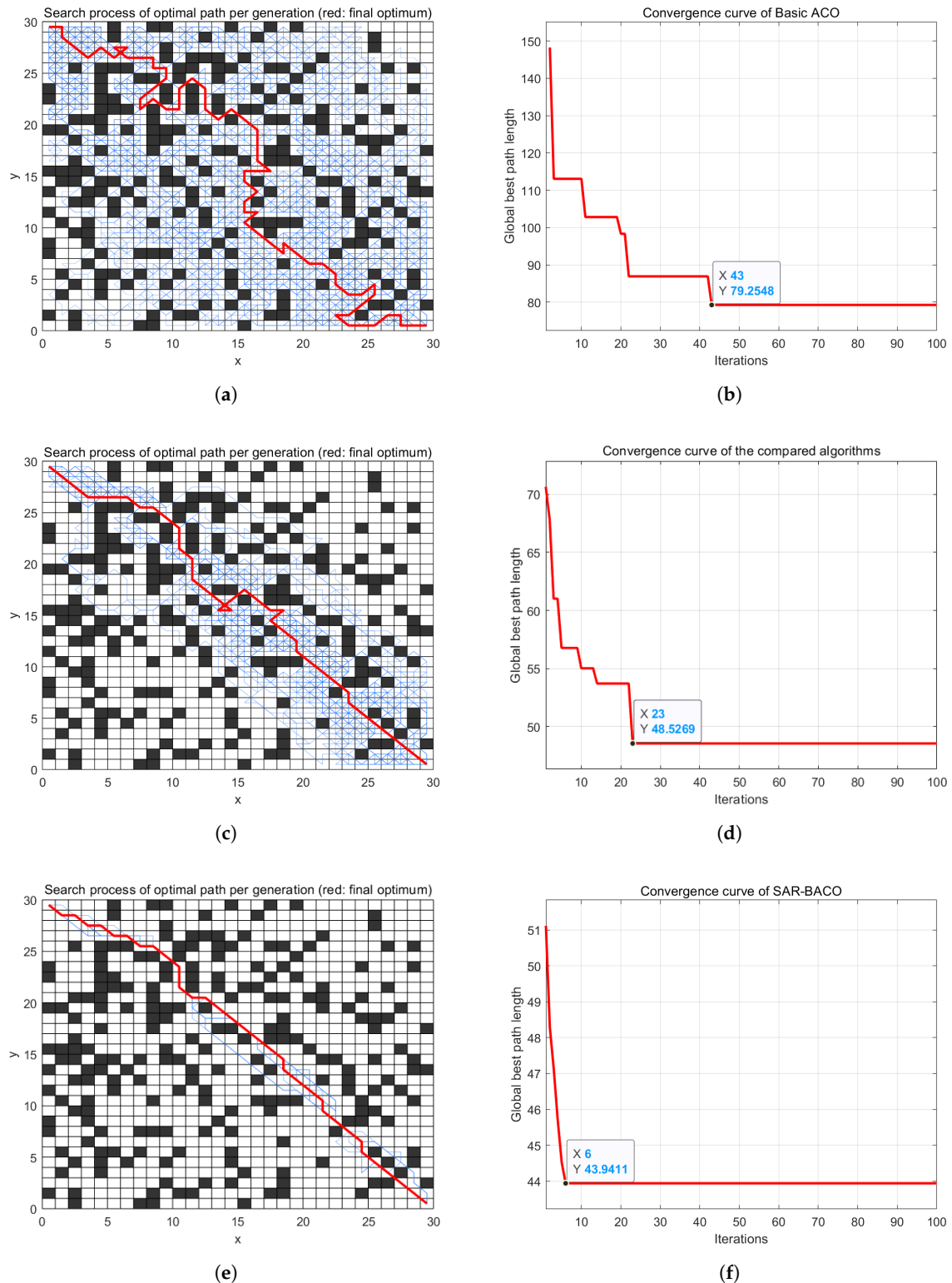


Figure 7. Experimental results on a 30×30 grid map: (a) Search process and optimal path of Basic ACO. (b) Convergence curve of Basic ACO. (c) Search process and optimal path of the algorithm in [5,14]. (d) Convergence curve of the algorithm in [5,14]. (e) Search process and optimal path of SAR-BACO. (f) Convergence curve of SAR-BACO. In subfigures (a,c,e), the blue lines represent the optimal paths explored in each generation, and the red line represents the final optimal path.

The results of the triangular pruning are shown in Figure 8. The blue lines in these figures indicate the paths after the triangular pruning had been done.

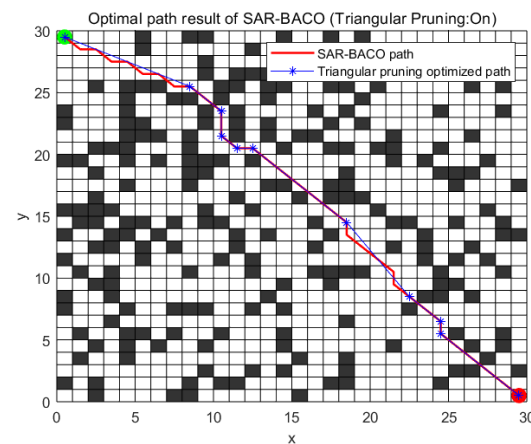


Figure 8. SAR-BACO after triangular pruning on a 30×30 grid map. The green and red dots denote the start and end points of the ant's movement, respectively.

SAR-BACO was implemented 50 times in equal experimental conditions. The average values of the performance metrics, such as optimal path length, convergence iteration count, number of path turning points and runtime are presented in Table 6.

It can be observed in Figure 7 that as the environmental complexity level slowly rose, the trajectories of the paths set by the original ACO tended to become even more zigzagged. This strongly affected path smoothness and rationality, but the algorithms in [5,14] and the proposed algorithm were much more prone to diagonally crossing grid paths. At the same time, the proposed algorithm produced paths with less angular changes than the algorithm in [5,14]. In more complex grid settings, the proposed algorithm was significantly superior, finding shorter and smoother paths. At the same time, it was also converging faster, which actually performed the task of path planning well.

Table 6. Performance comparison of algorithms in a 30×30 environment.

Algorithm	Optimal Path Length	Iterations to Convergence	Number of Path Turning Points	Runtime (s)
Basic ACO	84.3259 ± 3.92	42.0 ± 8.6	52.0 ± 7.4	9.6484 ± 0.64
Algorithms in [5,14]	54.5269 ± 2.86	17.0 ± 4.2	29.0 ± 4.1	8.4079 ± 0.58
SAR-BACO (before pruning)	45.6985 ± 2.31	13.0 ± 3.1	21.0 ± 3.6	7.7093 ± 0.49
SAR-BACO (after pruning)	44.0705 ± 2.08	13.0 ± 3.1	12.0 ± 2.4	7.7093 ± 0.49

SAR-BACO had length of paths reduced by 47.74%, number of convergence iterations reduced by 69.05%, number of turning points reduced by 76.92%, and runtime reduced by 20.1% as compared to baseline ACO. Relative to algorithm [5,14], SAR-BACO could reduce the path length by 19.18%, convergence iterations by 23.53%, turning points by 58.62% and running time by 8.31%.

4.5. 50×50 Grid Map

The start (entry) node was placed in the largest 50×50 environment in grid cell no. 1 at coordinates (0.5, 49.5); the end (destination) node was placed in the largest 50×50 environment in grid cell no. 2500 at coordinates (49.5, 0.5), and there was a regulation on obstacle density within the range of between 25% and 30%.

The baseline ACO had difficulty finding feasible routes in the largest 50×50 setting even after a finite number of iterations and algorithms in [5,14]; on the other hand, it had strong performance. In Figure 9 we can see how SAR-BACO was well capable of crossing large complex areas using a properly organized path. As is shown in the convergence

curves, SAR-BACO converged near the 49th generation, providing a well-balanced trade-off between the quality of the solutions and the computational complexity.

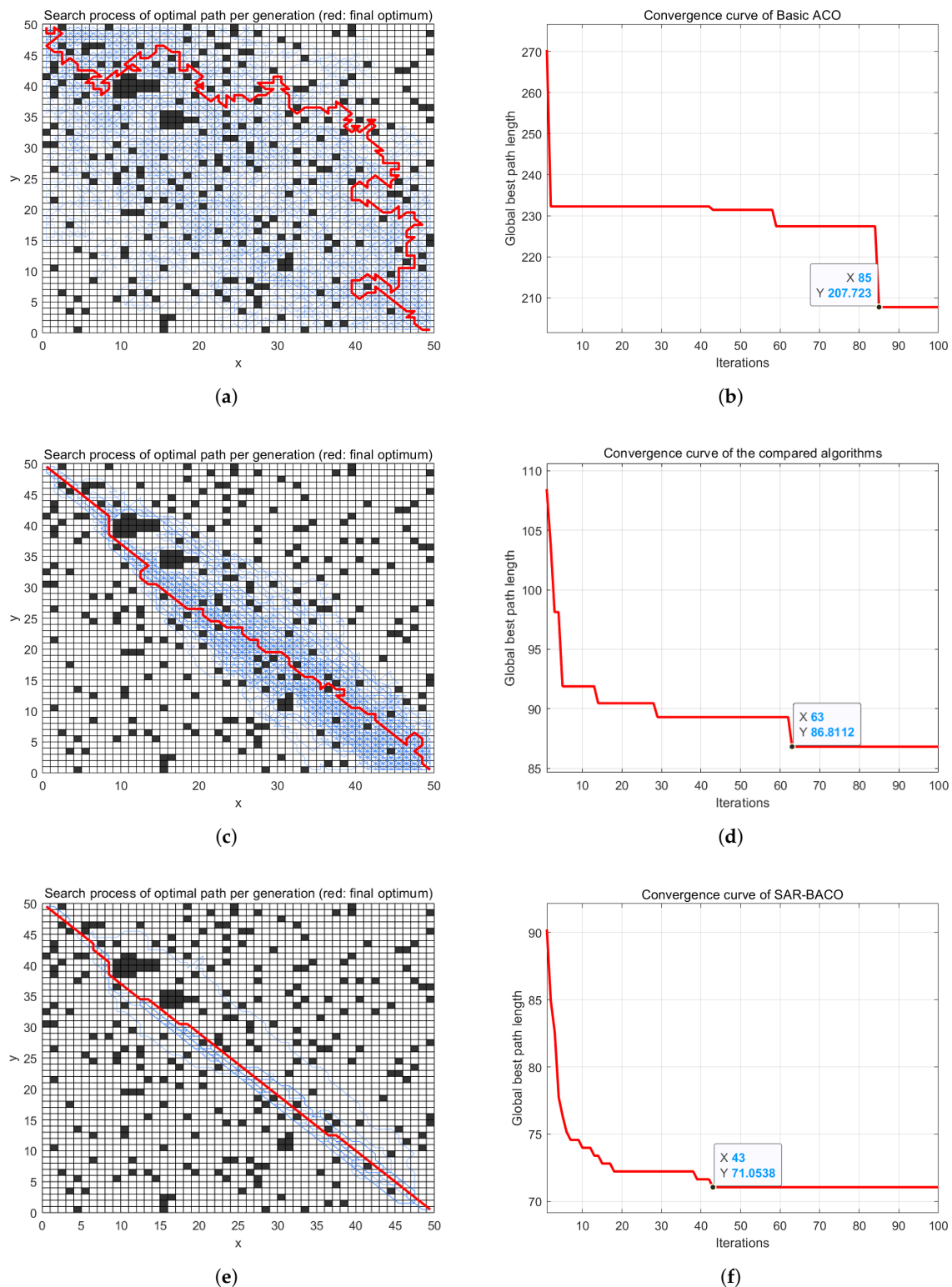


Figure 9. Experimental results on a 50×50 grid map: (a) Search process and optimal path of Basic ACO. (b) Convergence curve of Basic ACO. (c) Search process and optimal path of the algorithm in [5,14]. (d) Convergence curve of the algorithm in [5,14]. (e) Search process and optimal path of SAR-BACO. (f) Convergence curve of SAR-BACO. In subfigures (a,c,e), the blue lines represent the optimal paths explored in each generation, and the red line represents the final optimal path.

The triangular pruning results are demonstrated in Figure 10, with the blue lines being used as representative of the paths' post-triangular pruning.

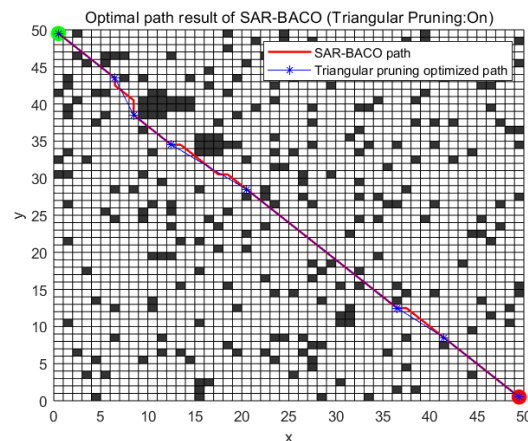


Figure 10. SAR-BACO after triangular pruning on a 50×50 grid map. The green and red dots denote the start and end points of the ant's movement, respectively.

The experiment with SAR-BACO was repeated 50 times in the same experimental conditions. In Table 7, one can see the average measures of such performance measures as optimal path length, count of convergence iterations, number of path turning points, and runtime.

Table 7. Performance comparison of algorithms in a 50×50 environment.

Algorithm	Optimal Path Length	Iterations to Convergence	Number of Path Turning Points	Runtime (s)
Basic ACO	177.3381 ± 8.64	96.0 ± 12.7	125.0 ± 14.9	21.0641 ± 1.62
Algorithms in [5,14]	87.6396 ± 4.73	53.0 ± 7.8	44.0 ± 6.2	21.9009 ± 1.70
SAR-BACO (before pruning)	71.0538 ± 3.91	49.0 ± 6.9	10.0 ± 2.0	14.0200 ± 1.11
SAR-BACO (after pruning)	69.7368 ± 3.62	49.0 ± 6.9	5.0 ± 1.2	14.0200 ± 1.11

SAR-BACO decreased the path length by 59.93%, the convergence iterations by 48.96%, and the runtime by 33.44% compared to the baseline ACO. After triangular pruning, the path length was reduced by 60.68 %, and the turning points were reduced even more drastically by 96% relative to the baseline ACO. In comparison with the algorithm of [5,14] SAR-BACO reduced the path length by 20.43%, the convergence iterations by 7.55% and the runtime by 33.53%; post-pruning turning points reduced by 93.18%.

4.6. Benchmark Experiments on Structured Maps

In addition to randomly generated grid environments, we further evaluated SAR-BACO on several structured benchmark maps commonly used in robot path planning research, including maze-like environments, corridor maps, and U-shaped trap scenarios. These maps represent typical navigation challenges such as narrow passages, dead-end traps, and constrained turning regions that frequently occur in real robotic navigation tasks.

The experimental setup and parameter configuration remained identical to those described in Section 4.1 to ensure fair comparison. We conducted simulation experiments using the ROS robot control framework in Gazebo (Figure 11). A four-wheel Ackermann-steering robot was deployed in a complex environment with multiple obstacles, and the SAR-BACO algorithm was compared against Basic ACO. As illustrated in Figure 12, SAR-BACO generated a significantly smoother and shorter path. As shown in the comparison of positions in Table 8, SAR-BACO reduced the path length by 34.82%, the number of turning

points by 40%, and the running time by 40.51% compared to Basic ACO, with discrepancies between simulation and computed values within $\pm 5\%$.

Our experimental results indicate that SAR-BACO consistently produces shorter and smoother paths than both Basic ACO and the heuristic-enhanced baseline algorithms. In particular, the directional-consistency component of the composite heuristic effectively guides ants through narrow corridors and avoids unnecessary exploration of dead-end regions, while the triangular pruning strategy removes redundant turning points commonly observed in maze-like environments.

Overall, these experiments demonstrate that the proposed SAR-BACO framework generalizes well beyond randomly generated maps and can produce more executable paths in structured navigation scenarios with complex geometric constraints.

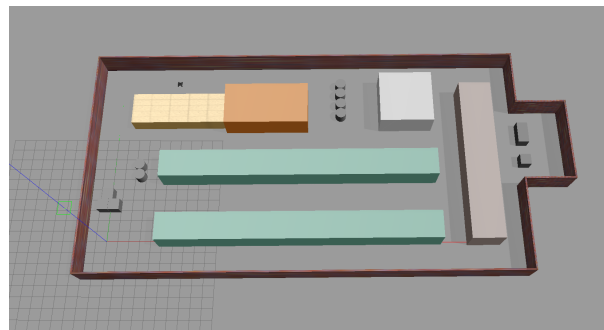
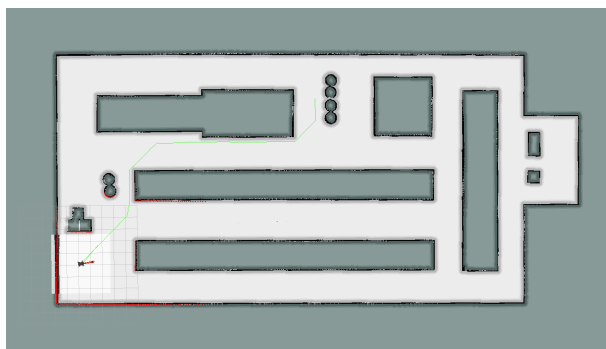
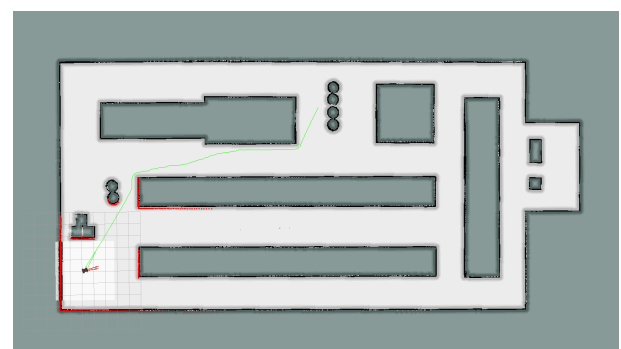


Figure 11. Gazebo simulation environment.



(a)



(b)

Figure 12. Simulation results of path planning: (a) Path planning results of Basic ACO. (b) Path planning results of SAR-BACO. The green lines represent the movement path of the vehicle.

Table 8. Comparison of ROS simulation verification results.

Algorithm	Optimal Path Length (cm)	Running Time (s)	Turning Points
Basic ACO	112	553	5
SAR-BACO	73	329	3

4.7. Statistical Significance and Robustness Analysis

To verify that the observed performance improvements were statistically reliable rather than incidental, all the quantitative results in Tables 5–7 are reported as mean $\pm$ standard deviation over 50 independent runs. Pairwise hypothesis testing was conducted between SAR-BACO and each baseline algorithm at different map scales.

Specifically, Welch's two-sided *t*-test (unequal variances) was applied to the distributions of (i) optimal path length, (ii) convergence iterations, (iii) turning points, and

(iv) runtime. In addition, Cohen's d was calculated to measure the practical significance of the observed differences.

Across 20×20 , 30×30 , and 50×50 maps, SAR-BACO (after pruning) showed statistically significant improvements over Basic ACO on all the evaluation metrics ($p < 0.001$ in all comparisons), with large effect sizes (Cohen's $d > 1.2$). Compared with the heuristic-enhanced baseline [5,14], SAR-BACO also demonstrated significant reductions in path length and turning points (typically $p < 0.01$), while maintaining comparable or lower runtime on larger maps.

The representative p -values and effect sizes are summarized in Table 9, confirming that the performance improvements of SAR-BACO are both statistically significant and practically meaningful.

Table 9. Statistical significance test results comparing SAR-BACO (after pruning) with baseline algorithms.

Map	Comparison	Metric	p -Value (Welch t -Test)	Effect Size (Cohen's d)
20×20	SAR-BACO (after) vs. Basic ACO	length	$<10^{-8}$	5.0
20×20	SAR-BACO (after) vs. Basic ACO	iterations	$<10^{-20}$	7.2
20×20	SAR-BACO (after) vs. Basic ACO	turning points	$<10^{-12}$	3.6
30×30	SAR-BACO (after) vs. Basic ACO	length	$<10^{-15}$	9.0
50×50	SAR-BACO (after) vs. Basic ACO	length	$<10^{-25}$	14.0
50×50	SAR-BACO (after) vs. Heuristic-enhanced ACO [5,14]	length	$<10^{-6}$	3.8
50×50	SAR-BACO (after) vs. Heuristic-enhanced ACO [5,14]	turning points	$<10^{-20}$	9.5

4.8. Parameter Sensitivity

To evaluate the robustness of SAR-BACO with respect to parameter selection, we analyzed the sensitivity of several key parameters that influence exploration–exploitation balance and pheromone discrimination, including the evaporation coefficient ρ , the penalty coefficient μ , and the adaptive heuristic bounds ($\omega_{\min}$, $\omega_{\max}$).

Our experiments were conducted on the 50×50 maps, where each parameter configuration was evaluated over 50 independent runs while keeping the remaining parameters fixed. The results are summarized in Table 10.

Table 10. Parameter sensitivity analysis of SAR-BACO (50×50 Map).

Setting (50×50)	Path Length	Iterations	Turning Points	Comment
default ($\rho = 0.8$, $\mu = 5$, $\omega_{\min} = 0.5$, $\omega_{\max} = 2.0$)	69.74	49	5	best trade-off
$\rho = 0.7$	70.52	45	5	faster but slightly longer
$\rho = 0.9$	72.40	56	5	slower convergence
$\mu = 2$	71.10	52	6	weaker suppression
$\mu = 8$	69.90	48	5	similar
$\omega_{\max} = 1.5$	71.36	53	6	weaker late guidance
$\omega_{\max} = 2.5$	69.80	49	5	similar

The results indicate that SAR-BACO maintained stable performance within practical parameter ranges ($\rho \in [0.7, 0.9]$, $\mu \in [2, 8]$, $\omega_{\max} \in [1.5, 2.5]$). Extremely large evaporation rates ($\rho = 0.9$) tended to slow convergence due to rapid pheromone decay, whereas overly small penalty coefficients reduced the suppression of inferior paths and slightly degraded path quality. Overall, the default configuration provided a balanced trade-

off between convergence efficiency and path quality and was therefore used in all the primary experiments.

4.9. Discussion

Based on the combined experimental results, SAR-BACO consistently outperforms the baseline ACO and representative heuristic-enhanced algorithms in terms of path length, convergence speed, path smoothness, and computational efficiency. The improvements originate from the integrated design of the proposed framework.

First, the composite heuristic provides goal-directed and risk-aware guidance that reduces early-stage random exploration and improves search efficiency in cluttered environments. Second, the bidirectional pheromone updating mechanism enhances the discrimination between high-quality and low-quality paths by simultaneously reinforcing global and iteration-best solutions while suppressing inferior paths, thereby reducing the likelihood of premature convergence to local optima. Third, the triangular pruning strategy effectively removes redundant turning points from discrete grid paths, producing smoother and more executable trajectories that better satisfy the motion characteristics of mobile robots.

Moreover, as the environment scale increases from 20×20 to 50×50 grids, the performance advantages of SAR-BACO become increasingly pronounced, demonstrating strong scalability and robustness in more complex navigation scenarios. These findings indicate that combining adaptive heuristic guidance, dynamic pheromone regulation, and geometric path pruning can significantly enhance the effectiveness of ACO-based robot path planning.

5. Conclusions

5.1. Concluding Remarks

This study proposes SAR-BACO, a robust adaptive path planning framework, addressing classic ACO's slow convergence and low smoothness via global–local heuristic, bidirectional pheromone update and triangular pruning. Theoretical analysis confirms pheromone boundedness/stable convergence via negative reinforcement and adaptive $\omega(t)$. SAR-BACO outperformed ACO variants across grids (up to 50×50), cutting path length by 60.68% and turning points by 96.00% (50×50), with Welch *t*-test validating significance ($p < 0.05$).

5.2. Limitations and Future Outlook

Despite the significant performance enhancements, certain limitations of the current study warrant further investigation:

1. **Static Environment Constraint:** While the low computational complexity of SAR-BACO ($O(T \cdot m \cdot n_{\text{avg}})$) suggests strong potential for real-time applications, the current validation was primarily conducted in static grid environments. The transition from offline global planning to online dynamic obstacle avoidance requires further integration with rolling window or receding horizon control strategies.
2. **Kinematic Constraints:** Although triangular pruning effectively minimizes redundant nodes, the resulting piecewise linear paths may still require further interpolation (e.g., via B-splines or Bézier curves) to satisfy the continuous curvature requirements of high-speed non-holonomic robots.
3. **Parameter Autonomy:** The current adaptive weights are governed by empirical bounds ($\omega_{\min}, \omega_{\max}$). Future work will explore the use of reinforcement learning (RL) or fuzzy logic controllers to autonomously tune these hyperparameters in response to varying obstacle densities and map topologies.

In conclusion, SAR-BACO provides a generalizable and efficient solution for autonomous navigation. Future research will focus on deploying this framework onto physical robotic platforms and extending its logic to multi-robot collaborative path planning in heterogeneous dynamic environments.

Author Contributions: Conceptualization, S.S. and Y.Z.; methodology, S.S. and Y.Z.; software, Y.Z.; validation, Y.Z., Y.W. and Y.Y.; formal analysis, Y.Z.; investigation, Y.W.; resources, S.S.; data curation, Y.Y.; writing—original draft preparation, Y.Z. and Y.W.; writing—review and editing, S.S.; visualization, Y.Z. and Y.Y.; supervision, S.S.; project administration, S.S.; funding acquisition, S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Natural Science Foundation of Chongqing Municipality under Grant CSTB2025NSCQ-GPX0735 and Grant CSTB2024NSCQ-LZX0143, and in part by the Chongqing Special Fund for Performance Incentive and Guidance for Scientific Research Institutions under Grant CSTB2023JXJL-YFX0013.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data are included in the article. Please contact the corresponding author if you need relevant data files.

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

Abbreviations

The following abbreviations are used in this manuscript:

ACO	Ant Colony Optimization
SAR-BACO	Self-Adaptive Risk-Aware Bidirectional Updating Ant Colony Optimization
AS	Ant System

References

1. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [\[CrossRef\]](#)
2. Patle, B.K.; Babu, L.G.; Pandey, A.; Parhi, D.R.K.; Jagadeesh, A. A review: On path planning strategies for navigation of mobile robot. *Def. Technol.* **2019**, *15*, 582–606. [\[CrossRef\]](#)
3. Mac, T.T.; Copot, C.; Tran, D.T.; De Keyser, R. Heuristic approaches in robot path planning: A survey. *Robot. Auton. Syst.* **2016**, *86*, 13–28. [\[CrossRef\]](#)
4. Liu, L.; Wang, X.; Yang, X.; Liu, H.; Li, J.; Wang, P. Path planning techniques for mobile robots: Review and prospect. *Expert Syst. Appl.* **2023**, *227*, 120254. [\[CrossRef\]](#)
5. Zhang, H.; Liu, W.; Chen, Y.; Li, J. Research on Path Planning Based on the Integrated Artificial Potential Field-Ant Colony Algorithm. *Appl. Sci.* **2025**, *15*, 4522. [\[CrossRef\]](#)
6. Abdulsahab, J.A.; Kadhim, D.J. Classical and heuristic approaches for mobile robot path planning: A survey. *Robotics* **2023**, *12*, 93. [\[CrossRef\]](#)
7. Liu, C.; Wu, L.; Xiao, W.S.; Li, G.; Xu, D.; Guo, J. An improved heuristic mechanism ant colony optimization algorithm for solving path planning. *Knowl.-Based Syst.* **2023**, *271*, 110540. [\[CrossRef\]](#)
8. Wu, L.; Huang, X.; Cui, J.; Liu, C.; Xiao, W. Modified adaptive ant colony optimization algorithm and its application for solving path planning of mobile robot. *Expert Syst. Appl.* **2023**, *215*, 119410. [\[CrossRef\]](#)
9. Chen, Y.; Wu, J.; He, C.; Jiang, S. Intelligent warehouse robot path planning based on improved ant colony algorithm. *IEEE Access* **2023**, *11*, 12360–12367. [\[CrossRef\]](#)
10. Dowlathshahi, M.B.; Fu, Y.; Rafsanjani, M.K.; Hashemi, A. Ant colony optimization enhanced with ensemble of pheromone vectors using multi-criteria decision making: A case study in multi-label text feature selection. *Appl. Soft Comput.* **2025**, *171*, 112914. [\[CrossRef\]](#)
11. Li, Q.; Li, Q.; Cui, B. Improved ant colony optimization algorithm based on islands type for mobile robot path planning. *Int. J. Adv. Robot. Syst.* **2024**, *21*, 1–17. [\[CrossRef\]](#)
12. Gao, C.; Pang, X.; Wang, C.; Wang, Y. Using Entropy as the Convergence Criteria of Ant Colony Optimization and the Application at Gene Chip Data Analysis. *Curr. Alzheimer Res.* **2024**, *21*, 324–341. [\[CrossRef\]](#)

13. Zhang, J.; Wang, L.; Meng, L.; Chen, H. Adaptive parameter control for ant colony optimization algorithms. In Proceedings of the 2024 IEEE Congress on Evolutionary Computation (CEC), Yokohama, Japan, 30 June–5 July 2024; pp. 1123–1130.
14. Wang, J.; Li, Y. Improved Trimming Ant Colony Optimization Algorithm for Mobile Robot Path Planning. *Algorithms* **2025**, *18*, 240. [[CrossRef](#)]
15. Yang, L.W.; Fu, L.X.; Li, P.; Mao, J.; Guo, N. An effective dynamic path planning approach for mobile robots based on ant colony fusion dynamic windows. *Machines* **2022**, *10*, 50. [[CrossRef](#)]
16. Zhang, S.; Pu, J.; Si, Y. An adaptive improved ant colony system based on population information entropy for path planning of mobile robot. *Complex Intell. Syst.* **2023**, *9*, 6791–6805. [[CrossRef](#)]
17. Gao, S.; Hou, X.; Li, C.; Liu, J. Intelligent Decision-Making for Multi-Scenario Resources in Virtual Power Plants Based on Improved Ant Colony Algorithm-Simulated Annealing Algorithm. *Sustainability* **2025**, *17*, 8600. [[CrossRef](#)]
18. Abdel-Basset, M.; Mohamed, R.; Abouhawwash, M. GOA-ACO: A goose optimized ant colony algorithm for the automated guided vehicle path planning. *Alex. Eng. J.* **2025**, *130*, 724–737. [[CrossRef](#)]
19. Wang, Y.; Zhang, B. A Dual-Mode Genetic-Ant Colony Algorithm for Dynamic Path Planning in Smart Scenic Spots. *Informatica* **2025**, *49*. [[CrossRef](#)]
20. Wu, L.; You, X.; Liu, S. Heterogeneous ant colony algorithm based on selective evolution mechanism and game strategy. *J. Supercomput.* **2024**, *80*, 7171–7206. [[CrossRef](#)]
21. Emami, H.; Fardi, M.; Azarnavid, B. An enhanced seasons optimization algorithm for numerical optimization and engineering design. *Sci. Rep.* **2025**, *15*, 25675. [[CrossRef](#)] [[PubMed](#)]
22. Wang, S. Temporal-contextual self-supervised time-series learning for automated fault detection and diagnosis of air handling units in buildings. *Build. Environ.* **2026**, *292*, 114300. [[CrossRef](#)]
23. Wang, S. Fault-class coverage-aligned combined training for AFDD of AHUs across multiple buildings. *Sci. Rep.* **2025**, *15*, 41192. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.