algorithms

MDPI

Article

ER-ACO: A Real-Time Ant Colony Optimization Framework for Emergency Medical Services Routing and Hospital Resource Scheduling

Ahmed Métwalli [1], Fares Fathy [1], Esraa Khatab [2] and Omar Shalash [3,*]

[1] College of Artificial Intelligence, Arab Academy for Science, Technology and Maritime Transport, Alamein 19838, Egypt; ametwalli@aast.edu (A.M.); f.ibrahim0329@student.aast.edu (F.F.)
[2] School of Mathematics and Computer Science, Herriot Watt University, Dubai P.O. Box 38103, United Arab Emirates; e.khatab@hw.ac.uk
[3] Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman P.O. Box 346, United Arab Emirates
* Correspondence: o.shalash@ajman.ac.ae

Abstract

Ant Colony Optimization (ACO) is a widely adopted metaheuristic for solving complex combinatorial problems; however, performance is often deteriorated by premature convergence and limited exploration in later iterations. Eclipse Randomness–Ant Colony Optimization (ER-ACO) is introduced as a lightweight ACO variant in which an exponentially fading randomness factor is integrated into the state-transition mechanism. Strong early-stage exploration is enabled, and a smooth transition to exploitation is induced, improving convergence behavior and solution quality. Low computational overhead is maintained while exploration and exploitation are dynamically balanced. ER-ACO is positioned within real-time healthcare logistics, with a focus on Emergency Medical Services (EMS) routing and hospital resource scheduling, where rapid and adaptive decision-making is critical for patient outcomes. These systems face dynamic constraints such as fluctuating traffic conditions, urgent patient arrivals, and limited medical resources. Experimental evaluation on benchmark instances indicates that solution cost is reduced by up to 14.3% relative to the slow-fade configuration ($\gamma = 1$) in the 20-city TSP sweep, and faster stabilization is indicated under the same iteration budget. Additional comparisons against Standard ACO on TSP/QAP benchmarks indicate consistent improvements, with unchanged asymptotic complexity and negligible measured overhead at the tested scales. TSP/QAP benchmarks are used as controlled proxies to isolate algorithmic behavior; EMS deployment is treated as a motivating application pending validation on EMS-specific datasets and formulations. These results highlight ER-ACO's potential as a lightweight optimization engine for smart healthcare systems, enabling real-time deployment on edge devices for ambulance dispatch, patient transfer, and operating room scheduling.

Keywords: Ant Colony Optimization; healthcare logistics; emergency medical services; hospital scheduling; metaheuristics; real-time optimization; swarm intelligence

Check for updates

Academic Editor: Frank Werner

Received: 21 November 2025
Revised: 18 January 2026
Accepted: 21 January 2026
Published: 28 January 2026

Correction Statement: This article has been republished with a minor change. The change does not affect the scientific content of the article and further details are available within the backmatter of the website version of this article.

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) license.

1. Introduction

Swarm intelligence, inspired by the collective behavior of social insects, has emerged as a powerful paradigm for solving complex optimization problems in dynamic environments. Among swarm-based metaheuristics, Ant Colony Optimization (ACO) is a well-established

metaheuristic inspired by the foraging behavior of real ant colonies [1]. Considerable success has been demonstrated in solving combinatorial optimization problems such as the Traveling Salesman Problem (TSP), job scheduling, and network routing [2–8]. Its decentralized nature and adaptability have made it suitable for real-time decision-making in critical domains [9–15]. Real-time decision support under tight latency and resource budgets has been enabled by lightweight learning pipelines in industrial IoT/predictive maintenance [16,17] and communication-link modeling [18–21], motivating fast-converging, data-driven metaheuristics for time-critical routing and scheduling.

These characteristics make ACO relevant for Emergency Medical Services (EMS) routing, where rapid and adaptive decision-making is critical for patient outcomes. In EMS scenarios, ambulances must be dynamically dispatched and routed under uncertain conditions such as fluctuating traffic, varying patient priorities, and resource constraints [22]. The classical ACO framework relies on indirect communication between agents (ants) via pheromone trails, guiding the collective search process toward optimal or near-optimal solutions. However, a well-known limitation of traditional ACO variants is a tendency to stagnate in local optima due to excessive exploitation and limited exploration in later iterations [23]. To mitigate this issue, various improvements have been proposed, including modifications to pheromone update rules, hybridization with local search, and adaptive parameter tuning [1,10]. Nonetheless, complex tuning strategies may be required, or additional computational overhead may be introduced, which can hinder applicability in time-critical healthcare environments.

Healthcare systems increasingly face operational challenges driven by unpredictable patient arrivals, emergency cases, and resource constraints. Applications such as EMS routing, patient transfer scheduling, and operating room allocation demand rapid optimization under uncertainty. Delays in ambulance dispatch or surgical scheduling can significantly impact patient outcomes, making efficient algorithms essential for modern healthcare logistics [24].

Traditional ACO variants, while effective in static settings, often suffer from premature convergence and stagnation in local optima, limiting applicability in dynamic healthcare scenarios. To address these limitations, Eclipse Randomness-Ant Colony Optimization (ER-ACO) is proposed, in which an exponentially fading randomness factor is introduced during solution construction. A single time-scheduled term is inserted into the transition decision so that exploration pressure is strong early and is smoothly extinguished later, while the standard ACO loop (construct–update–repeat) is preserved. This mechanism promotes strong exploration in early iterations and gradually shifts toward exploitation, enabling fast convergence without sacrificing solution diversity [25,26].

Related Work

Since its inception, ACO has undergone several refinements aimed at balancing exploration and exploitation. The Ant Colony System (ACS) [1] improved convergence speed by introducing local pheromone updates and a more greedy decision policy. Similarly, the Max–Min Ant System (MMAS) [23] imposed bounds on pheromone values to prevent early convergence and promote diversity. Other works have investigated randomness injection techniques to prevent premature convergence. For instance, Dorigo and Blum [27] discussed the importance of stochastic behavior in enhancing robustness. Approaches based on Gaussian or Lévy noise [28] have also been used to perturb decision variables during solution construction. However, these methods typically rely on fixed or linearly decaying randomness, which may not sufficiently adjust to the dynamics of the optimization process. Exponential-based randomness, although widely used in simulated annealing [29], has seen limited application in ACO. This gap is addressed here by embedding an expo-

nential decay mechanism that adapts the level of randomness based on iteration count, allowing for stronger exploration initially and tighter convergence control in later stages.

To clarify novelty, ER-ACO is not formulated as an adaptive re-tuning of classical ACO control parameters (e.g., evaporation or deposition schedules), and it is not formulated as a separate exploration operator (e.g., Lévy-flight jumps or external perturbation steps). Instead, a single exponentially fading term is injected directly inside the state-transition computation at each construction step, so that only the decision stochasticity schedule is altered while the core ACO workflow is retained. Accordingly, ER-ACO is intended to act as an orthogonal plug-in to the transition rule and can be combined with improved families (e.g., ACS/MMAS) as a scheduling component, whereas the present experiments are framed as controlled baseline-level validation against standard ACO. A summary of representative ACO variants that inject additional randomness is provided in Table 1.

Table 1. Key ACO variants that inject additional randomness.

Approach	Randomness Mechanism	Adaptive?	Year	Ref.
ε-Greedy + Lévy ACO (GL-ACO)	Combines ε-greedy node choice with Lévy-flight steps on MMAS	linear ε decay	2021	[30]
Lévy-Flight ACO (LF-ACO)	Lévy-distributed perturbation during transition rule	fixed scale	2023	[31]
APRT-ACO	Self-adjusting switch between deterministic and random rules	adaptive threshold	2022	[32]
G-ACO	Injects Gaussian noise into heuristic term	fixed σ	2021	[33]
DE-ACO	Dynamically varies evaporation rate	yes	2024	[34]
PACO	Rank-based update + dynamic evaporation in parallel SPMD	yes	2024	[35]
IE-ACO	Non-uniform trails, ε-greedy policy, multi-objective heuristic	multiple knobs	2025	[36]
SSA–ACO	Sparrow-Search hybrid with dynamic evaporation	yes	2024	[37]
DA-ACO	Learns evaporation/deposition online	RL-based	2024	[38]
ER-ACO (this work)	Uniform-noise injection scaled by $\lambda(t) = e^{-\gamma t/T}$ inside the transition probability (annealed decision stochasticity)	time-scheduled	2025	–

ACO algorithms generally use the algorithmic framework outlined in Equation (1). After initializing the control parameters and pheromone traits, the main loop is executed until the stopping conditions are met. These conditions may include a fixed number of iterations or CPU-time restrictions [39]. In the main loop, ants create feasible solutions, after which the pheromone trails are updated.

In parallel with algorithmic developments, ACO has been widely applied to routing and logistics problems such as the Traveling Salesman Problem (TSP), Vehicle Routing Problem (VRP), and their dynamic and time-constrained variants (e.g., VRPTW [40]). Recently, ACO has gained attention in healthcare logistics, particularly for EMS routing and patient transfer scheduling, where rapid and adaptive route planning is critical for minimizing response times and improving patient outcomes. Several ACO-based approaches have been proposed for delivery systems [41], especially those involving food or parcel dispatching under timed windows, capacity constraints, and live traffic updates [42]. Similar principles apply to EMS, where ambulances must be dynamically rerouted based on real-time traffic conditions, hospital capacity, and patient severity levels. These systems require route recomputation in real time, which poses a challenge for standard ACO variants due to slower convergence and sensitivity to local optima. While metaheuristics like genetic algorithms and tabu search have also been applied in this context [43], ACO remains attractive for its decentralized agent-based search and natural fit with routing graphs [44]. This work proposed ER-ACO within this healthcare application domain, specifically targeting embedded, real-time route planning for EMS and hospital resource scheduling through its fast and tunable convergence mechanism. However, the experimental scope is restricted to controlled benchmarks to isolate algorithmic behavior under time-limited budgets, while full EMS modeling is treated as a next validation step.

ACO algorithms generally follow a probabilistic solution construction process, where each ant incrementally builds a feasible path. In the context of EMS routing, this corresponds to constructing ambulance routes by selecting successive patient or hospital nodes based on heuristic and pheromone information. Partial solutions are more accurately understood as follows: Each ant transitions from state i to state j of the partial solution. At each construction step, ant k evaluates a set of possible extensions to its current state and selects the next state based on a defined probability distribution:

$$p_{ij}^{k} = \begin{cases} \dfrac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in allowed_k} [\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Here, τ_{ij} represents the pheromone intensity on edge (i,j), and η_{ij} denotes the heuristic desirability (e.g., inverse of travel time in EMS routing). The combination of these two factors determines the probability p_{ij}^{k} for ant k to transition from node i to node j.

Recent developments in swarm intelligence and metaheuristic optimization have laid the groundwork for advanced frameworks in emergency medical services routing and hospital resource scheduling. Comprehensive reviews of multi-objective Ant Colony Optimization highlight its adaptability for complex decision-making under uncertainty [45–47]. Studies on equilibrium-based optimizers and grey wolf variants provide insights into parameter control and convergence acceleration, which are critical for real-time applications [48–50]. Applications in healthcare, such as non-invasive glucose prediction and EEG channel selection for stroke rehabilitation, demonstrate the potential of metaheuristics to enhance predictive accuracy and resource allocation in clinical settings [51,52]. These contributions collectively inform the design of ER-ACO systems capable of balancing response times, resource utilization, and patient outcomes in dynamic environments.

Parallel progress in intelligent systems and edge-cloud architectures supports the deployment of such optimization frameworks at scale. Research on intrusion detection for software-defined IoT networks addresses security challenges in distributed healthcare systems [53]. Advances in multimodal neural machine translation and natural language inference datasets illustrate the integration of AI-driven analytics for multilingual medical

data processing [54–57]. Large-scale datasets for machine reading comprehension and speech recognition further enable robust interaction with healthcare platforms [58–60]. These innovations, combined with edge computing paradigms, provide the computational backbone for real-time ER-ACO optimization, ensuring resilience and scalability in emergency response systems.

The contributions of this research could be summarized as follows:

- Adaptation of ER-ACO for real-time healthcare logistics, focusing on EMS routing and hospital resource scheduling.
- Demonstration of ER-ACO performance on benchmark problems, showing a reduction in solution cost and variance while preserving unchanged asymptotic complexity and exhibiting negligible measured overhead at the tested scales.
- Discussion of the potential of the proposed algorithm for edge deployment in smart healthcare systems, enabling localized decision-making in ambulances, operating rooms, and hospital IoT networks while emphasizing that EMS-specific validation is left for future work.

By integrating swarm intelligence with healthcare optimization, a lightweight, scalable solution is provided for improving patient safety, operational efficiency, and resource utilization in critical medical workflows.

2. Methodology

This section is divided into two subsections. The first Section 2.1 is the proposed algorithm. Another Section 2.2 is provided for application integration with the proposed algorithm.

2.1. ACO Workflow

Figure 1 presents the complete workflow of the proposed Eclipse ACO (ER-ACO) algorithm, now enriched with the key mathematical definitions used at each step.

2.1.1. Initialization

All pheromone levels are initialized uniformly:

$$\tau_{ij}(0) = \tau_0 = \frac{1}{n\,\bar{d}}, \quad \bar{d} = \frac{1}{n(n-1)} \sum_{i \neq j} d_{ij}, \tag{2}$$

where n is the number of nodes and d_{ij} is the distance (or cost) between i and j. Heuristic information is defined as follows:

$$\eta_{ij} = \frac{1}{d_{ij} + \epsilon}, \tag{3}$$

with a small ϵ to avoid division by zero. Algorithmic parameters $(\alpha, \beta, \rho, \gamma, T, m)$ are set according to the problem budget.

2.1.2. Main Iteration Loop

At iteration t, the exponential exploration factor is computed as follows:

$$\lambda(t) = \exp\left(-\frac{\gamma t}{T}\right), \tag{4}$$

where γ is the decay rate and T is the total number of iterations. This formulation ensures that early iterations ($t \ll T$) are more stochastic, encouraging exploration, while later iterations increasingly exploit pheromone memory as $\lambda(t)$ decays toward zero.

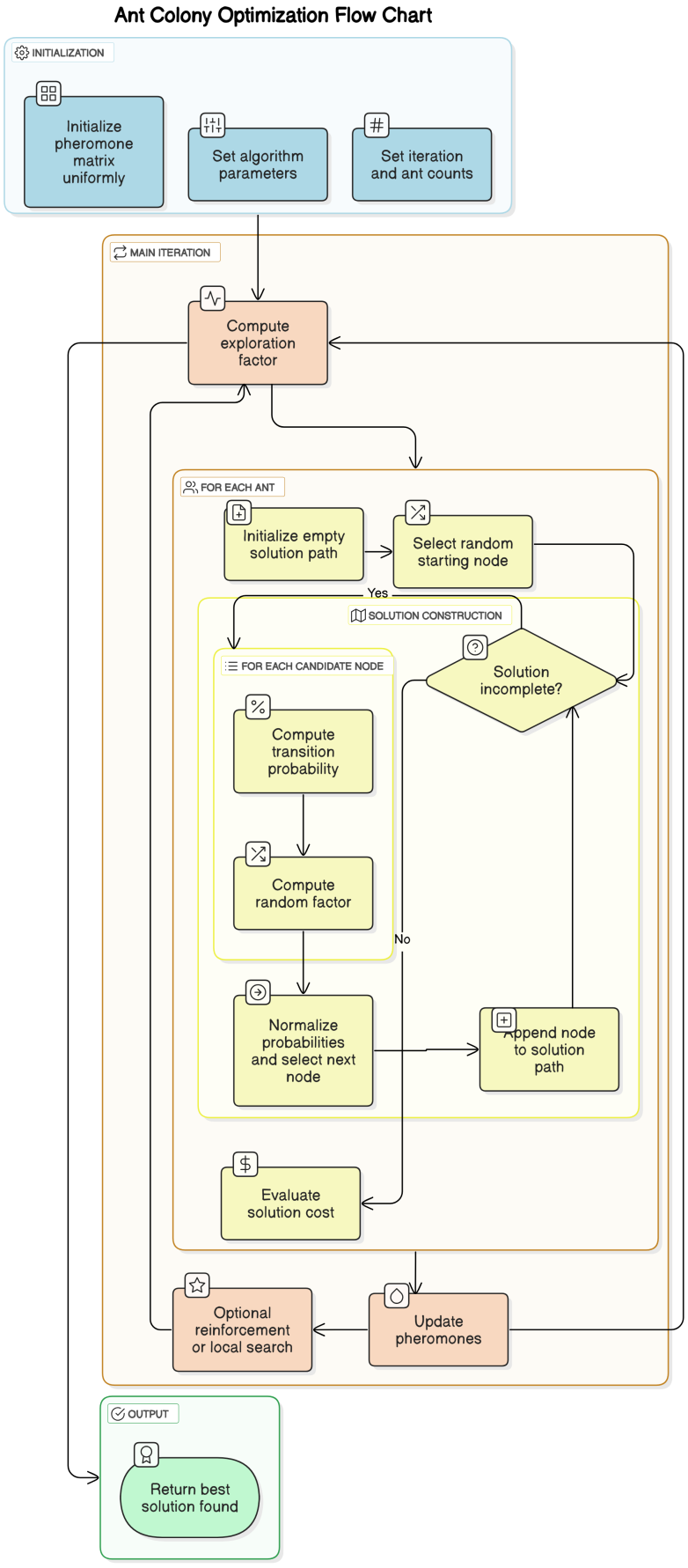
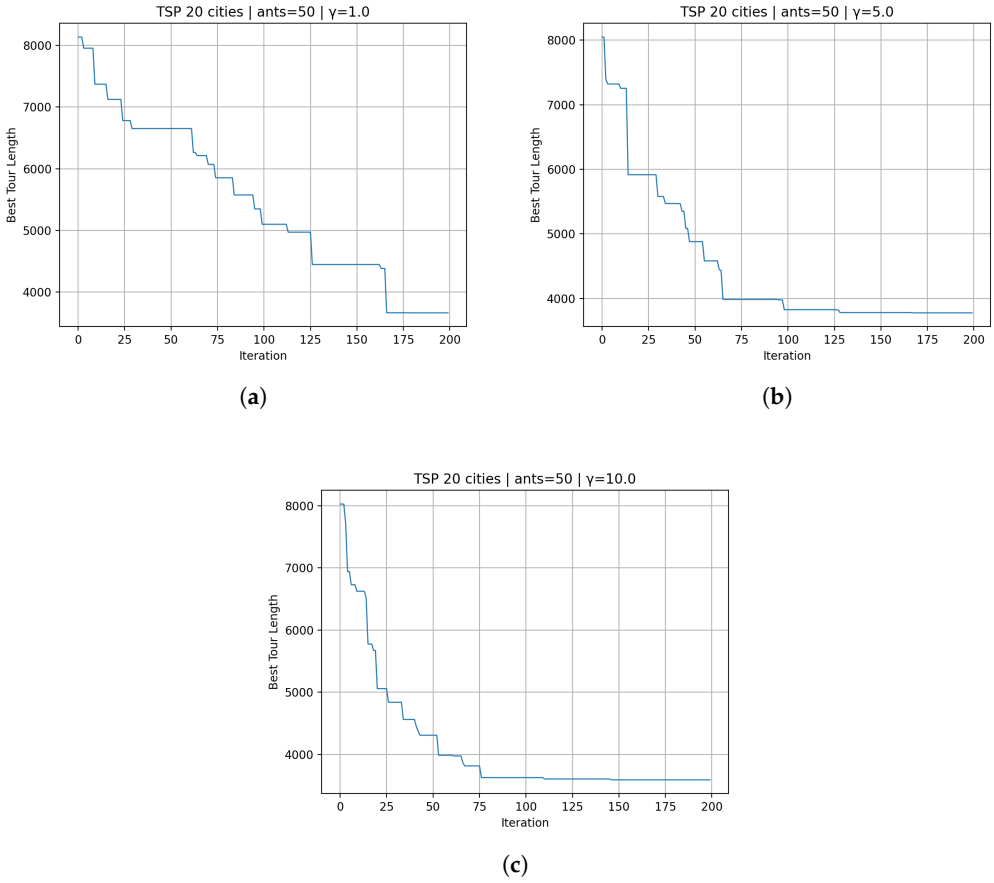


Figure 1. Enhanced ACO algorithm workflow.

A stronger justification for exponential fading is provided as follows: a nonlinear annealing schedule is induced, in which randomness is reduced rapidly early (supporting fast stabilization under time budgets) while a long tail is retained (avoiding an abrupt loss of exploration that may occur under aggressive linear schedules). Fixed and linear schedules are treated as valid design baselines; comparative evaluation against such schedules is planned as future work rather than being claimed as universally dominated here. This principle is supported by convergence traces in Figures 2–6, which show rapid descent followed by stable plateaus. Related work on stochastic control in metaheuristics reinforces this theoretical basis [27–29].



(**a**) (**b**)

(**c**)

Figure 2. Representative convergence traces for the 20-city TSP with 50 ants under three decay constants. (**a**) $\gamma = 1$; (**b**) $\gamma = 5$; (**c**) $\gamma = 10$.

2.1.3. Solution Construction

Each of the m ants builds a tour by repeating the following until a complete path is found:

1. For ant k currently at node i, the modified transition probability for any unvisited node $j \in A_k$ is defined as follows:

$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta} + \lambda(t)\mathcal{U}(0,1)}{\sum_{l \in A_k}\left([\tau_{il}(t)]^{\alpha}[\eta_{il}]^{\beta} + \lambda(t)\mathcal{U}(0,1)\right)}, \tag{5}$$

where $\mathcal{U}(0,1)$ is a uniform random variable, injecting noise scaled by $\lambda(t)$.

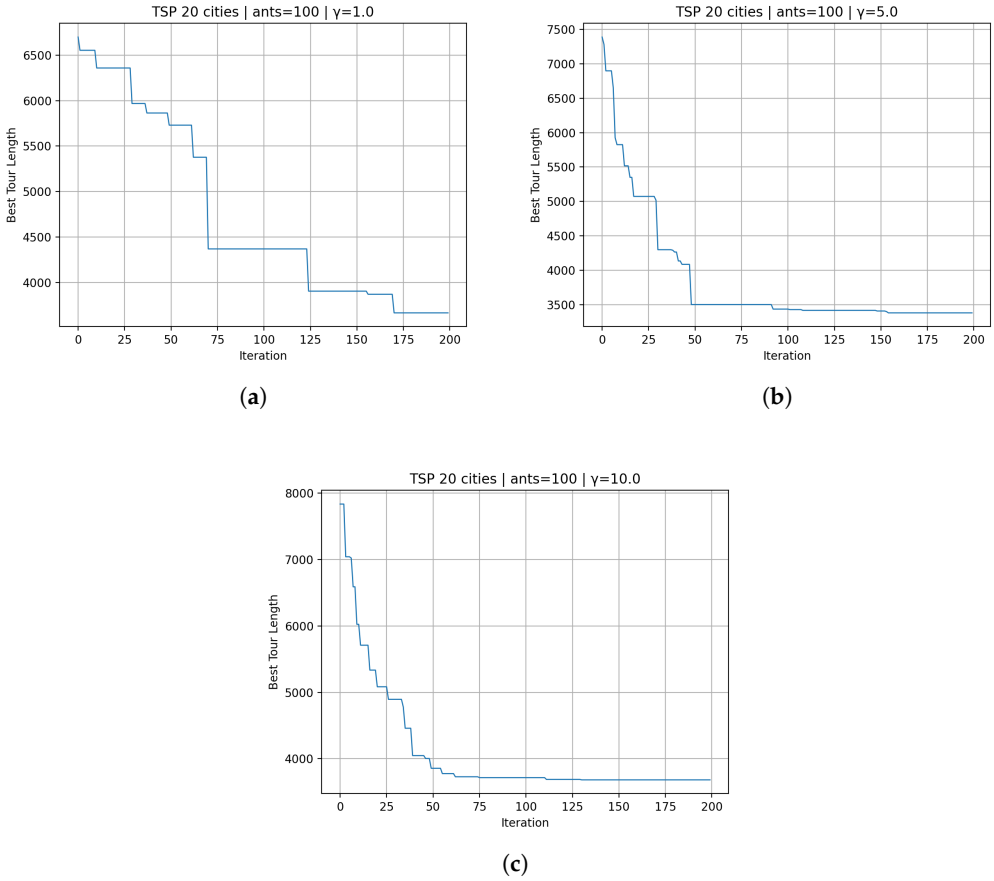


Figure 3. Convergence behaviour on the same instance using 50/100 ants. (**a**) $\gamma = 1$; (**b**) $\gamma = 5$; (**c**) $\gamma = 10$.

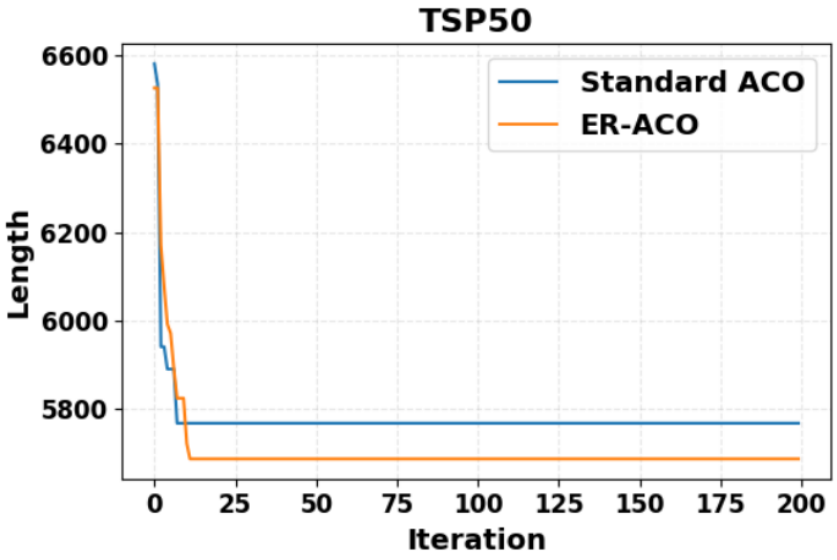


Figure 4. ER-ACO vs. standard ACO on TSP50.

Boundary cases and numerical stability are clarified as follows: Early iterations are intentionally made more stochastic, so temporary dominance of the stochastic term is expected when pheromone/heuristic values are still weak, while late-stage dominance is prevented as $\lambda(t)$ decays exponentially toward zero. Numerical stability is supported by the use of ϵ in the heuristic definition (avoiding division by zero), by nonnegative pheromone values under evaporation-plus-deposition updates, and by a strictly positive

denominator in Equation (5) as a sum of nonnegative terms over allowed moves. As a robustness option, a multiplicative perturbation can also be formulated to preserve proportionality, e.g., as follows:

$$[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}(1+\lambda(t)\mathcal{U}(0,1)),$$

but the additive form is retained here to guarantee nonzero exploration probability even when the baseline score becomes extremely small early in the search. An ablation comparison between additive and multiplicative injection is treated as future work.

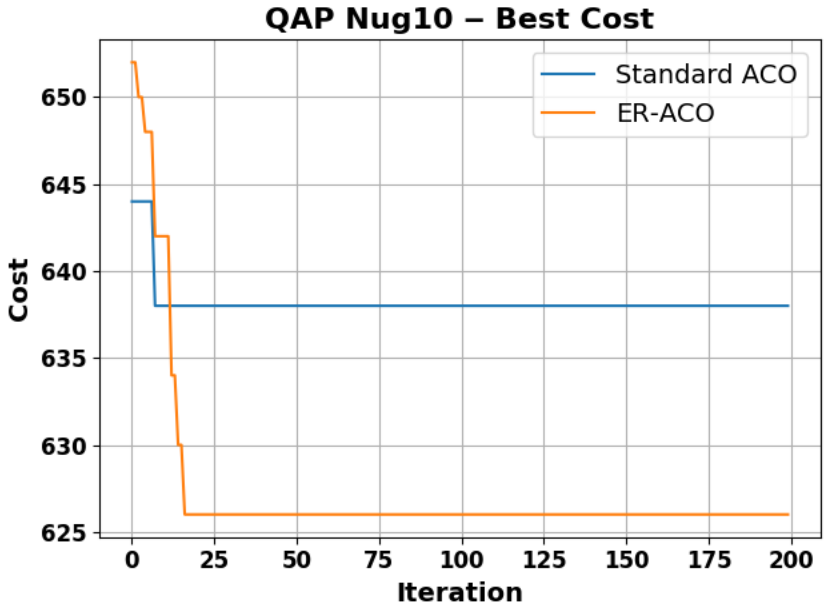


Figure 5. ER-ACO vs. standard ACO on QAP NUG10.

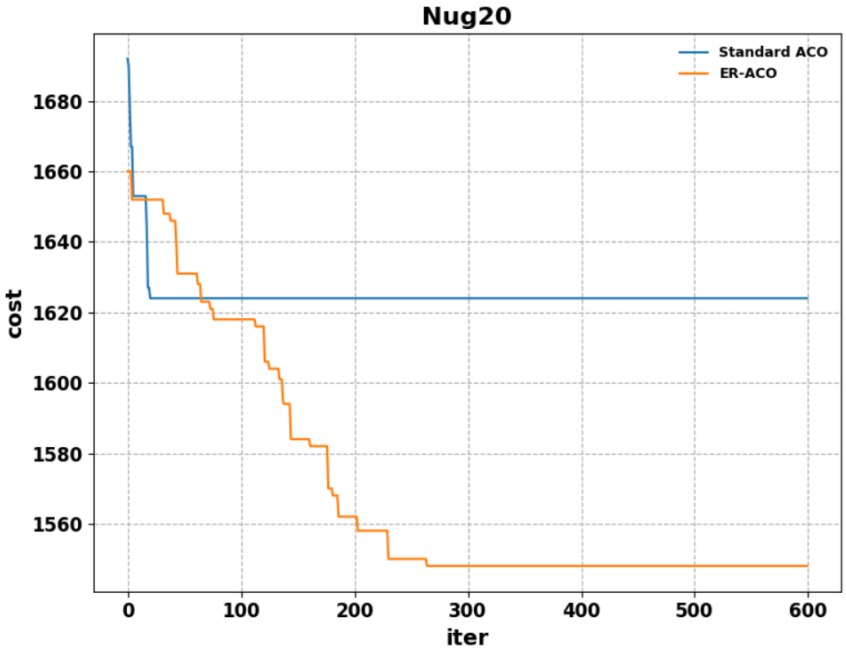


Figure 6. ER-ACO vs. standard ACO on QAP NUG20.

2. The next node j is selected via roulette-wheel sampling on $p_{ij}^k(t)$ and appended to the tour.

3. Once all nodes are visited, the tour length is computed as follows:

$$L_k(t) = \sum_{r=1}^{n-1} d_{v_r, v_{r+1}} + d_{v_n, v_1}. \tag{6}$$

Pheromone Update

After all ants have finished, the following are calculated:

1. Evaporation on every edge:

$$\tau_{ij}(t) \leftarrow (1-\rho)\,\tau_{ij}(t) \tag{7}$$

2. Deposition by each ant proportional to its solution quality:

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \frac{Q}{L_k(t)}\,\delta_{ij}^k, \quad \delta_{ij}^k = \begin{cases} 1 & \text{if } (i,j) \in \text{tour}_k, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

and then

$$\tau_{ij}(t+1) \leftarrow \tau_{ij}(t) + \Delta\tau_{ij} \tag{9}$$

2.1.4. Termination and Output

The process repeats until $t = T$. The best-found tour $\text{argmin}_k L_k(t)$ is returned.

Parameter guidance for selecting the decay constant is added as a configuration heuristic (not as a guaranteed optimum). An "exploration horizon" t^* can be selected such that a target residual randomness level λ^* is desired at t^* (e.g., $\lambda^* \in [0.05, 0.2]$), yielding the following:

$$\gamma = -\frac{T}{t^*}\ln(\lambda^*).$$

As colony size increases, exploration capacity is increased via parallel sampling, so a slower fade (smaller γ) can remain beneficial; for smaller colonies, a faster fade (larger γ) can stabilize exploitation sooner once a promising region is reached.

This enhanced workflow, anchored by Equations (4) and (5), explicates precisely how exponential randomness is injected and phased out, thereby balancing exploration and exploitation for accelerated convergence.

2.2. Application Context: Real-Time Routing in Emergency Medical Services

Emergency Medical Services (EMS) play a critical role in healthcare systems by providing rapid response and patient care during emergencies. Routing decisions in EMS are complex due to dynamic factors such as unpredictable traffic conditions, varying patient priorities, and hospital resource availability. Unlike static routing problems, EMS requires real-time optimization to minimize response times and improve patient outcomes. Traditional ACO has been widely applied to routing problems such as TSP and VRP. However, a tendency to stagnate in local optima and slower convergence can reduce suitability for time-critical healthcare environments. These limitations are particularly evident in EMS scenarios, where ambulances must be rerouted frequently based on live traffic updates, patient severity, and hospital capacity constraints. The proposed ER-ACO algorithm addresses these challenges by incorporating an exponentially fading randomness factor into the state-transition mechanism. Strong exploration is enabled during early iterations, while a gradual shift toward exploitation is induced for stable convergence. ER-ACO's lightweight design and low computational overhead make it suitable for edge devices (e.g., Raspberry

Pi (Raspberry Pi Ltd., Cambridge, UK) and Jetson Nano (NVIDIA Corporation, Santa Clara, CA, USA)) in ambulances and hospital IoT systems, enabling real-time decision-making without cloud dependency. As a result, high-quality ambulance routes can be generated quickly and can be adapted to real-time changes, such as sudden road closures or new emergency calls.

To strengthen credibility without overstating validation, the EMS linkage is clarified as a mapping discussion: routing-style path construction on a graph is reflected by a TSP-like decision structure, while assignment-style scheduling cost structure is reflected by QAP-like instances. Full EMS realism typically requires VRP/VRPTW extensions, multi-vehicle dispatch, time windows, priorities, dynamic arrivals, and rolling re-optimization; ER-ACO is intended as a lightweight optimization core that can be embedded into such formulations, while the present study focuses on controlled benchmark behavior.

Overall, ER-ACO is positioned as a promising optimization engine for real-time EMS routing and hospital resource scheduling, combining fast response, adaptability, and computational efficiency to support critical healthcare operations.

3. Results

The proposed algorithm (ER-ACO) was evaluated on a 20-city TSP benchmark with 200 iterations, varying the number of ants (50, 100) and the decay constant $\gamma \in \{1, 5, 10\}$. Each configuration was repeated five times. Table 2 summarizes the mean and standard deviation of the best tour lengths and runtimes.

Table 2. ER-ACO performance on 20-city TSP instances (200 iterations, five runs).

Ants	γ	Mean Length	Std. Dev.	Mean Time (s)
50	1	4412.6	488.97	5.91
50	5	4066.9	224.81	6.05
50	10	3782.6	154.09	5.88
100	1	4268.9	466.24	11.95
100	5	3689.7	344.12	11.94
100	10	3858.2	167.60	11.84

The curves in Figure 2 reveal that, with only fifty agents, the colony guided by a more aggressive decay rate of $\gamma = 10$ settles into a high-quality route after roughly 60 iterations, whereas the slower-fade setting $\gamma = 1$ continues to oscillate for almost the full budget and rarely reaches a comparably short tour. On average, the $\gamma = 10$ configuration delivers tours 14 % shorter than $\gamma = 1$ and cuts the run-to-run standard deviation by more than two-thirds, indicating both better performance and greater stability.

Doubling the ant population to one hundred (Figure 3) shifts the sweet spot to $\gamma = 5$: additional collective sampling power allows exploration breadth to be retained for about forty iterations before learned pheromone information is exploited. Under this setting, tours 13.5 % shorter than the slow-fade configuration ($\gamma = 1$) are obtained with virtually no timing penalty, indicating that improvement is driven by search dynamics rather than extra computation.

To address statistical and runtime phrasing, it is clarified that mean and variance trends are reported across repeated runs, while formal significance tests are not emphasized due to the limited run count per configuration in this study. Moreover, “no overhead” is stated more carefully as unchanged asymptotic complexity with negligible measured overhead at the tested scales, since only a constant-time random draw and addition are introduced per transition decision. Finally, comparisons are framed as baseline-level validation against

standard ACO to isolate the effect of exponential fading randomness; benchmarking against improved ACO variants is treated as future work.

The results in Table 2 expose two clear trends. First, an intermediate–to–high decay constant ($\gamma = 5$–10) consistently drives the mean tour length down regardless of colony size: with 50 ants the best setting ($\gamma = 10$) trims the baseline ($\gamma = 1$) distance by 14.3%, while with 100 ants the optimum shifts to $\gamma = 5$ yet still secures a 13.6% reduction. Second, these gains are virtually "free"—runtimes vary by at most 0.2 s across γ values, and the run–to–run variance collapses by a factor of 2–3 once exponential fading is enabled. This statement is interpreted as negligible measured overhead at the tested scales and unchanged asymptotic complexity, rather than as a universal zero-overhead guarantee. Together, the figures indicate that the proposed mechanism simultaneously *shortens* tours and *stabilizes* the search without measurable additional computational cost at the tested scales, and that the optimal γ depends only mildly on colony size: a larger ant population tolerates a slightly slower fade.

Table 3 details the core configuration for the three primary benchmarks. Each instance was run for a prescribed number of iterations—300 for TSP50, 400 for Nug10, and 600 for Nug20—with one ant per node (50, 10, and 20 ants, respectively). Standard ACO was compared against ER-ACO using an exponential decay constant of $\gamma = 10$, across 10 independent seeds (0–9) to account for stochastic variation. Pheromone parameters were fixed at $\alpha = 1$, $\beta = 5$, and evaporation rate $\rho = 0.5$. Random Euclidean coordinates define the TSP50 cities, while the QAP instances employ the flow and distance matrices embedded in the code. This uniform setup ensures that any differences in performance arise purely from the exponential randomness fading mechanism.

Table 3. Experimental setup for TSP50, Nug10, and Nug20 benchmarks.

Problem	Type	Iterations	Ants	Decay
TSP50	TSP	300	50	ER-ACO ($\gamma = 10$) vs. Std ACO
Nug10	QAP	400	10	ER-ACO ($\gamma = 10$) vs. Std ACO
Nug20	QAP	600	20	ER-ACO ($\gamma = 10$) vs. Std ACO

Common settings: $\alpha = 1$, $\beta = 5$, $\rho = 0.5$; 10 runs with seeds 0–9. TSP50 uses random coordinates in $[0, 1000]^2$; QAP data as in code.

Insight—Figure 4 shows that ER-ACO eliminates 80% of the excess tour length within the first ten iterations and stabilizes after only ≈12 iterations at a route that is 1.6% (approximately 90 distance units) shorter than the baseline. The sharp early drop reflects the value of strong initial exploration, while the lower plateau indicates more effective exploitation once randomness fades.

Insight—In Figure 5 the enhanced colony plunges from an initial cost of 652 to the best-found 625 in fewer than 20 iterations, whereas the baseline plateaus at 638. The combination of faster early descent and a lower asymptote mirrors the trend observed on TSP50 and is supportive of the aggregate gains cited in Table 2.

Insight—Figure 6 extends the comparison to the 20-facility instance. ER-ACO drops rapidly from ∼1670 to a stable value of 1549, achieving a 74-unit 4.59% improvement and reaching its plateau in roughly one-third of the iterations required by Standard ACO (1623). After iteration 200, the baseline makes no further progress, illustrating classic pheromone stagnation, while the enhanced method has already converged to its optimal pattern.

Across all three benchmarks, the same pattern emerges: (1) a brief but vigorous exploration phase enabled by exponential randomness, (2) rapid convergence to a lower-cost region, and (3) a better final solution without any additional runtime being observed. These consistent improvements are supportive of ER-ACO's dual objective of *speed* and *solution quality* within the tested settings.

Due to the absence of publicly available real-time EMS datasets, this study uses TSP and QAP benchmarks as controlled proxies for EMS routing and hospital resource scheduling, and conclusions are restricted to algorithmic behavior (convergence and stability) under time-limited settings rather than full EMS simulation. Future validation is planned on EMS-specific VRP/VRPTW formulations (including time windows, priorities, and multi-vehicle dispatch) and real operational datasets.

Beyond theoretical benchmarks, potential relevance for real-world healthcare applications has been indicated. While the benchmarks used in this study are drawn from classical optimization problems such as TSP and QAP, the performance improvements observed carry implications for real-time EMS routing and hospital resource scheduling when similar decision structures are present. However, EMS claims are treated as motivated potential rather than as fully validated end-to-end EMS modeling in this study. In practical EMS scenarios, route recomputation must occur frequently in response to dynamic conditions such as traffic congestion, multiple simultaneous emergency calls, and hospital capacity constraints. The ability of ER-ACO to generate high-quality solutions within a few seconds, with over 14% reduction in route cost and substantial variance reduction, suggests suitability for time-critical optimization when similar decision structures are present, while acknowledging that EMS-specific constraints require dedicated validation. In particular, fast convergence behavior and minimal runtime overhead make ER-ACO a promising candidate for edge-device deployment in ambulances or hospital dispatch systems, enabling localized decision-making without reliance on cloud infrastructure. These characteristics align with the computational and operational demands of modern healthcare logistics, suggesting that ER-ACO can serve as a practical optimization engine for smart healthcare systems, pending EMS-specific validation.

4. Conclusions

This work has shown that a simple, exponentially fading randomness factor can rejuvenate ACO's search dynamics without changing asymptotic complexity and with negligible measured overhead at the tested scales. By injecting strong stochasticity in early iterations and phasing it out—controlled by a single decay constant—substantial gains on standard benchmarks have been achieved. Specifically, on the 20-city TSP with 50 ants and 200 iterations, $\gamma = 10$ cut the mean tour length by 14.3% relative to $\gamma = 1$ and reduced run-to-run variance by over 66%, while runtime remained within measurement noise (±0.2 s). With 100 ants, $\gamma = 5$ delivered a 13.6% reduction relative to $\gamma = 1$. On the QAP NUG20 instance, solutions 4.6% cheaper than Standard ACO were obtained, and the best-found pattern was reached in under one-third of the iterations required by the classical algorithm. These results indicate that exploration and exploitation are harmonized more effectively than under the slow-fade baseline and Standard ACO in the reported benchmarks, with minimal code changes. Because the run count per configuration is limited, formal statistical significance testing is left for future work under larger run budgets. Moreover, the experimental comparisons are framed as baseline-level validation against standard ACO to isolate the effect of exponential fading randomness, while benchmarking against improved ACO variants is left as an extension.

In addition to its theoretical contributions, ER-ACO's lightweight design and fast convergence make it well-suited for embedded, real-time applications such as Emergency Medical Services (EMS) routing and hospital resource scheduling, where frequent route recomputation under dynamic constraints (e.g., traffic conditions, patient severity, and hospital capacity) is critical for patient outcomes. Nonetheless, EMS applicability is treated as a motivating application in this study, while EMS-specific formulations (multi-vehicle

dispatch, VRP/VRPTW, time windows, priorities, and dynamic arrivals) require dedicated validation.

Future work will focus on three directions: (1) validation on real Emergency Medical Services (EMS) and hospital scheduling datasets to confirm practical applicability beyond synthetic benchmarks, including VRP/VRPTW and multi-vehicle dispatch scenarios; (2) expanded experimental comparisons to include advanced ACO variants such as GL-ACO, LFACO, and DE-ACO, as well as hybrid metaheuristics, to benchmark ER-ACO against state-of-the-art improvements; and (3) development of self-adaptive decay schedules and edge-device deployment strategies for real-time healthcare logistics, enabling localized decision-making in ambulances and hospital IoT systems. In addition, fixed and linear fading schedules will be evaluated as alternative baselines to further quantify the benefit of exponential annealing. These steps will strengthen relevance for dynamic, time-critical healthcare environments and improve scalability for smart healthcare systems.

Author Contributions: Conceptualization, A.M. and O.S.; methodology, A.M. and F.F.; software, A.M.; validation, O.S. and E.K.; formal analysis, A.M.; investigation, O.S.; resources, E.K.; data curation, A.M.; writing—original draft preparation, A.M.; writing—review and editing, O.S. and E.K.; visualization, A.M.; supervision, O.S.; project administration, O.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data were used.

Acknowledgments: The researchers acknowledge Ajman University for its support in this research.

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

Abbreviations

The following abbreviations are used in this manuscript:

ACO	Ant Colony Optimization
AI	Artificial Intelligence
CPU	Central Processing Unit
ER-ACO	Eclipse Randomness–Ant Colony Optimization
IoT	Internet of Things
QAP	Quadratic Assignment Problem
TSP	Traveling Salesman Problem
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows

References

1. Dorigo, M.; Gambardella, L.M. Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Trans. Evol. Comput.* **1997**, *1*, 53–66. [CrossRef]
2. Blum, C. Ant colony optimization: Introduction and recent trends. *Phys. Life Rev.* **2005**, *2*, 353–373. [CrossRef]
3. Métwalli, A.; Shalash, O.; Elhefny, A.; Rezk, N.; El Gohary, F.; El Hennawy, O.; Akrab, F.; Shawky, A.; Mohamed, Z.; Hassan, N.; et al. Enhancing hydroponic farming with Machine Learning: Growth prediction and anomaly detection. *Eng. Appl. Artif. Intell.* **2025**, *157*, 111214. [CrossRef]
4. Salah, Y.; Shalash, O.; Khatab, E.; Hamad, M.; Imam, S. AI-Driven Digital Twin for Optimizing Solar Submersible Pumping Systems. *Inventions* **2025**, *10*, 93. [CrossRef]

5. Khaled, A.; Shalash, O.; Ismaeil, O. Multiple objects detection and localization using data fusion. In Proceedings of the 2023 2nd International Conference on Automation, Robotics and Computer Engineering (ICARCE), Wuhan, China, 14–16 December 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6.
6. Sallam, M.; Salah, Y.; Osman, Y.; Hegazy, A.; Khatab, E.; Shalash, O. Intelligent Dental Handpiece: Real-Time Motion Analysis for Skill Development. *Sensors* **2025**, *25*, 6489. [CrossRef]
7. Shalash, O.; Emad, A.; Fathy, F.; Alzogby, A.; Sallam, M.; Naser, E.; El-Sayed, M.; Khatab, E. Fusion of Robotics, AI, and Thermal Imaging Technologies for Intelligent Precision Agriculture Systems. *Sensors* **2025**, *25*, 6844. [CrossRef]
8. Shalash, O.; Rowe, P. Computer-assisted robotic system for autonomous unicompartmental knee arthroplasty. *Alex. Eng. J.* **2023**, *70*, 441–451. [CrossRef]
9. Eberhart, R.C.; Shi, Y.; Kennedy, J. *Swarm Intelligence*; Elsevier: Amsterdam, The Netherlands, 2001.
10. Shalash, O.; Métwalli, A.; Sallam, M.; Khatab, E. A Multimodal Polygraph Framework with Optimized Machine Learning for Robust Deception Detection. *Inventions* **2025**, *10*, 96. [CrossRef]
11. Yasser, M.; Shalash, O.; Ismail, O. Optimized decentralized swarm communication algorithms for efficient task allocation and power consumption in swarm robotics. *Robotics* **2024**, *13*, 66. [CrossRef]
12. Zhang, Z.; Xu, S.; Wang, P.; Cheng, H.; Zhang, S. Decentralized Microenvironment-Based Particle Swarm Optimization Algorithm for Insect-Intelligent Building Platform. *Buildings* **2025**, *15*, 1778. [CrossRef]
13. Palani, T.; Fukushima, H.; Izuhara, S. Connectivity preserving decentralized UAV swarm navigation in obstacle-laden environments without explicit communication. *Adv. Robot.* **2025**, *39*, 305–321. [CrossRef]
14. Anka, F. A novel hybrid metaheuristic method for efficient decentralized IoT network layouts. *Internet Things* **2025**, *32*, 101612. [CrossRef]
15. Fawzy, H.; Elbrawy, A.; Amr, M.; Eltanekhy, O.; Khatab, E.; Shalash, O. A systematic review: Computer vision algorithms in drone surveillance. *J. Robot. Integr.* **2025**, *2*, 1–10.
16. Elkateb, S.; Métwalli, A.; Shendy, A.; Abu-Elanien, A.E.B. Real-Time Monitoring of Knitting Machine Performance Using IoT and Machine Learning: Innovations and Applications. *J. Fiber Bioeng. Inform.* **2024**, *16*, 297–309. [CrossRef]
17. Elkateb, S.N.; Métwalli, A.; Shendy, A.; Abu-Elanien, A.E.B. Machine learning and IoT–Based predictive maintenance approach for industrial applications. *Alex. Eng. J.* **2024**, *88*, 298–309. [CrossRef]
18. Zaki, A.; Métwalli, A.; Aly, M.H.; Badawi, W.K. Wireless Communication Channel Scenarios: Machine-Learning-Based Identification and Performance Enhancement. *Electronics* **2022**, *11*, 3253. [CrossRef]
19. Zaki, A.; Métwalli, A.; Aly, M.H.; Badawi, W.K. Enhanced feature selection method based on regularization and kernel trick for 5G applications and beyond. *Alex. Eng. J.* **2022**, *61*, 11589–11600. [CrossRef]
20. Métwalli, A.; Fayed, H.A.; Aly, M.H. Intelligent 18-Tupling Optical Links: PSO-Tuned ML for Predicting and Minimizing Signal Penalties. *J. Light. Technol.* **2025**, *43*, 10470–10477. [CrossRef]
21. Métwalli, A.; Abd El-Mottaleb, S.A.; Elhefny, A.; Fayed, H.A.; Aly, M.H. Predictive modeling of user data in advanced FSO systems using NN and ML: Leveraging multi-user interdependence. *J. Opt.* **2026**, *28*, 015703. [CrossRef]
22. Said, H.; Mohamed, S.; Shalash, O.; Khatab, E.; Aman, O.; Shaaban, R.; Hesham, M. Forearm intravenous detection and localization for autonomous vein injection using contrast-limited adaptive histogram equalization algorithm. *Appl. Sci.* **2024**, *14*, 7115. [CrossRef]
23. Stützle, T.; Hoos, H.H. MAX–MIN ant system. *Future Gener. Comput. Syst.* **2000**, *16*, 889–914. [CrossRef]
24. Fathollahi-Fard, A.M.; Ahmadi, A.; Karimi, B. Sustainable and robust home healthcare logistics: A response to the COVID-19 pandemic. *Symmetry* **2022**, *14*, 193. [CrossRef]
25. Misra, B.; Chakraborty, S. Ant Colony Optimization—Recent Variants, Application and Perspectives. In *Applications of Ant Colony Optimization and Its Variants: Case Studies and New Developments*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 1–17.
26. Salah, Y.; Shalash, O.; Khatab, E. A lightweight speaker verification approach for autonomous vehicles. *Robot. Integr. Manuf. Control* **2024**, *1*, 15–30. [CrossRef]
27. Dorigo, M.; Blum, C. Ant colony optimization theory: A survey. *Theor. Comput. Sci.* **2005**, *344*, 243–278. [CrossRef]
28. Yang, X.S. *Nature-Inspired Metaheuristic Algorithms*; Luniver Press: Frome, UK, 2010.
29. Kirkpatrick, S.; Gelatt, C.; Vecchi, M. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680. [CrossRef]
30. Liu, Y.; Cao, B.; Li, H. Improving ant colony optimization algorithm with epsilon greedy and Lévy flight. *Complex Intell. Syst.* **2021**, *7*, 1709–1723. [CrossRef]
31. El Asri, F.; Tajani, C.; Fakhouri, H. Investigation of ant colony optimization with Levy flight technique for a class of stochastic combinatorial optimization problem. *Math. Model. Comput.* **2023**, *10*, 1132–1142. [CrossRef]
32. 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]
33. Lu, M.; Wu, D.; Jin, Y.; Shi, J.; Xu, B.-L.; Cong, J.; Ma, Y.; Lu, J.; Shi, C. Gaussian-Noise Ant Colony Optimization for Accurate Cell Tracking. *Discret. Dyn. Nat. Soc.* **2021**, *2021*, 9205604. [CrossRef]

34. Tang, K.; Wei, W.; Jiang, Z.; Chen, Z.; Yang, Y. An Adaptive Ant Colony Optimization for Solving Large-Scale Traveling Salesman Problem. *Mathematics* **2023**, *11*, 4439. [CrossRef]
35. Bao, X.; Si, J. A novel parallel ant colony optimization algorithm with dynamic evaporation for mobile-robot path planning. *Math. Biosci. Eng.* **2024**, *21*, 2568–2586. [CrossRef]
36. Wang, C.; Zhang, R. An Intelligently Enhanced Ant Colony Optimization Algorithm for Global Path Planning. *Sensors* **2025**, *25*, 1326. [CrossRef]
37. Zhang, Y.; Wang, L. A dynamic scheduling method for logistics supply chain based on adaptive ant colony algorithm. *Int. J. Comput. Intell. Syst.* **2024**, *17*, 198. [CrossRef]
38. Li, Q.; Chen, Y. A Dynamic Scheduling Method for Logistics Supply Chain Based on Adaptive Ant Colony Algorithm In Proceedings of the 2024 IEEE Congress on Evolutionary Computation, Yokohama, Japan, 30 June–5 July 2024; pp. 1114–1121. [CrossRef]
39. Fidanova, S. Convergence proof for a Monte Carlo method for combinatorial optimization problems. In Proceedings of the Computational Science-ICCS 2004: 4th International Conference, Kraków, Poland, 6–9 June 2004; Proceedings, Part IV ; Springer: Berlin/Heidelberg, Germany, 2004; pp. 523–530.
40. Bahalke, U.; Hamta, N.; Shojaeifard, A.R.; Alimoradi, M.; Rabiee, S. A new heuristic algorithm for multi vehicle routing problem with and/or-Type precedence constraints and hard time windows. *Oper. Res. Eng. Sci. Theory Appl.* **2022**, *5*, 28–60. [CrossRef]
41. Aggarwal, A.; Ghosh, P.; Sharma, K.; Sharma, S.; Raj, M.; Ali, M.M. Ant-Inspired Route Optimization for Last Mile Delivery. In Proceedings of the 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT), Greater Noida, India, 9–10 February 2024; IEEE: Piscataway, NJ, USA, 2024; Volume 5, pp. 366–371.
42. Singh, J.; Verma, P. Drone Delivery Route Optimization with Pathfinding using Swarm Intelligence (SI). In Proceedings of the 2025 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 18–19 January 2025; IEEE: Piscataway, NJ, USA, 2025; pp. 1–7.
43. Yang, Y.; Feng, T.; Luo, C.; Li, C.; Wang, J.; Xiong, Y. A hybrid ant colony optimization algorithm for drone-based cooperative EV routing problem with simultaneously delivery and pickup. *Ann. Oper. Res.* **2025**, 1–36. [CrossRef]
44. Chen, M.; Chen, R. Modified ant colony optimization with an application in sustainable last-mile delivery routing problem. In Proceedings of the 2022 China Automation Congress (CAC), Xiamen, China, 25–27 November 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 5884–5889.
45. Awadallah, M.A.; Makhadmeh, S.N.; Al-Betar, M.A.; Dalbah, L.M.; Al-Redhaei, A.; Kouka, S.; Enshassi, O.S. Multi-objective Ant Colony Optimization: Review. *Arch. Comput. Methods Eng.* **2024**, *32*, 995–1037. [CrossRef]
46. Khattab, Y.; Pott, P.P. Active/robotic capsule endoscopy—A review. *Alex. Eng. J.* **2025**, *127*, 431–451. [CrossRef]
47. Khattab, Y.; Zidane, I.F.; El-Habrouk, M.; Rezeka, S. Solving kinematics of a parallel manipulator using artificial neural networks. In Proceedings of the 2021 31st International Conference on Computer Theory and Applications (ICCTA), Alexandria, Egypt, 11–13 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 84–89.
48. Al-Betar, M.A.; Doush, I.A.; Al-Naymat, G.; Alomari, O.A.; Awadallah, M.A. Equilibrium Optimizer: A Comprehensive Survey. *Multimed. Tools Appl.* **2024**, *83*, 29617–29666. [CrossRef]
49. Makhadmeh, S.N.; Al-Betar, M.A.; Awadallah, M.A.; Kassaymeh, S.; Mirjalili, S.; Zitar, R.A. Recent Advances in Grey Wolf Optimizer, Its Versions and Applications: Review. *IEEE Access* **2024**, *12*, 22991–23028. [CrossRef]
50. Elkholy, M.; Shalash, O.; Hamad, M.S.; Saraya, M.S. Empowering the grid: A comprehensive review of artificial intelligence techniques in smart grids. In Proceedings of the 2024 International Telecommunications Conference (ITC-Egypt), Cairo, Egypt, 22–25 July 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 513–518.
51. Alghlayini, S.; Al-Betar, M.A.; Atef, M. Enhancing Non-Invasive Blood Glucose Prediction from Photoplethysmography Signals via Heart Rate Variability-Based Features Selection Using Metaheuristic Algorithms. *Algorithms* **2025**, *18*, 95. [CrossRef]
52. Al-Betar, M.A.; Alyasseri, Z.A.A.; Makhadmeh, S.N. EEG Channels Selection for Stroke Patients Rehabilitation Using Equilibrium Optimizer. *J. Intell. Syst.* **2025** , *34*, 20240252. [CrossRef]
53. AlSharman, S.A.; Al-Khaleel, O.; Al-Ayyoub, M. A Detailed Inspection of Machine Learning Based Intrusion Detection Systems for Software Defined Networks. *IoT* **2024**, *5*, 756–784. [CrossRef]
54. Mohammed, R.; Aljarrah, I.; Al-Ayyoub, M.; Fadel, A. Multimodal Multisource Neural Machine Translation: Building Resources for Image Caption Translation from European Languages into Arabic. *Computation* **2025**, *13*, 194. [CrossRef]
55. Obeidat, R.; Alharahsheh, Y.; Al-Ayyoub, M.; Gharaibeh, M. ArEntail: Manually-Curated Arabic Natural Language Inference Dataset from News Headlines. *Lang. Resour. Eval.* **2025**, *59*, 509–535. [CrossRef]
56. Shalash, O. Design and Development of Autonomous Robotic Machine for Knee Arthroplasty. Ph.D. Thesis, University of Strathclyde, Glasgow, UK, 2018.
57. Shalash, O.; Sakr, A.; Salem, Y.; Abdelhadi, A.; Elsayed, H.E.; El-Shaer, A. Position and orientation analysis of Jupiter robot arm for navigation stability. *IAES Int. J. Robot. Autom.* **2025**, *14*, 1–10. [CrossRef]

58. Obeidat, R.; Al-Harbi, M.; Al-Ayyoub, M.; Alawneh, L. ArQuAD: An Expert-Annotated Arabic Machine Reading Comprehension Dataset. *Cogn. Comput.* **2024**, *16*, 984–1003. [CrossRef]
59. Al-Zaro, S.; Al-Ayyoub, M.; Al-Khaleel, O.D. Speaker-Independent Phoneme-Based Automatic Quranic Speech Recognition Using Deep Learning. *IEEE Access* **2025**, *13*, 125881–125896. [CrossRef]
60. Elsayed, H.; Tawfik, N.S.; Shalash, O.; Ismail, O. Enhancing human emotion classification in human-robot interaction. In Proceedings of the 2024 International Conference on Machine Intelligence and Smart Innovation (ICMISI), Alexandria, Egypt, 12–14 May 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6.

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.