

Article

Energy Efficiency Optimization in Swarm Robotics for Smart Photovoltaic Monitoring

Dimitris Ziouzos ^{1,2}, Nikolaos Baras ² , Minas Dasygenis ², Vayos Karayannis ^{1,*} 
and Constantinos Tsanaktsidis ¹ 

¹ Department of Chemical Engineering, University of Western Macedonia, ZEP Campus, 50100 Kozani, Greece; dziouzos@uowm.gr (D.Z.); ktsanaktsidis@uowm.gr (C.T.)

² Department of Electrical and Computer Engineering, University of Western Macedonia, ZEP Campus, 50100 Kozani, Greece; nbaras@uowm.gr (N.B.); mdasygenis@uowm.gr (M.D.)

* Correspondence: vkarayannis@uowm.gr

Abstract: Photovoltaic park (PV) and power generator monitoring is a crucial activity that calls for effective coverage path planning. Artificial intelligence and particularly swarm robotics have brought new methods to tasks such as coverage path planning by having multiple robots work together to cover a specific area. Nonetheless, enhancing energy efficiency in these systems continues to be a crucial obstacle, particularly with the growing focus on sustainability. This research investigates techniques to enhance energy efficiency in swarm robotics, focusing on coverage path planning assignments. The proposed approach merges advanced swarm robotics algorithms with energy-efficient methods to reduce power consumption while still ensuring effective coverage. Thorough simulations in simulated environments of Western Macedonia assess the efficiency of the proposed approach. Even though the proposed approach has a longer convergence time compared to a generic ACO approach, the findings of the simulations indicate that the MOACO approach has substantial enhancements up to 22% in path travel time, in terms of solution quality and energy consumption metrics. The findings of the present work offer valuable insights into the design of sustainable robotic systems and underscore the potential of swarm robotics in achieving efficient coverage path planning. This study adds to the overall objective of creating eco-friendly technologies in robotics, leading to upcoming advancements in the industry.

Keywords: energy optimization; artificial intelligence (AI); swarm robotics; photovoltaic (PV) monitoring; coverage path planning



Academic Editor: Anastassios M. Stamatielos

Received: 19 February 2025

Revised: 16 March 2025

Accepted: 19 March 2025

Published: 22 March 2025

Citation: Ziouzos, D.; Baras, N.; Dasygenis, M.; Karayannis, V.; Tsanaktsidis, C. Energy Efficiency Optimization in Swarm Robotics for Smart Photovoltaic Monitoring. *Energies* **2025**, *18*, 1587. <https://doi.org/10.3390/en18071587>

Copyright: © 2025 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 (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Swarm robotics has emerged as a transformative field in recent years, revolutionizing the way that researchers approach complex problems [1–3]. By harnessing the collective power of multiple robots working together, swarm robotics enables efficient and effective solutions to tasks that would be challenging or impossible for individual robots to accomplish alone. Progress in artificial intelligence (AI), computer science, and engineering has led to a major change in the creation of advanced robotic systems that can interact with one another and their surroundings. Swarm robotics began in the natural world, where groups of animals like birds, fish, and insects have evolved to work together for common goals. Long captivated by these natural systems, researchers' study of swarming behavior has motivated the creation of synthetic swarm systems. Studies on bird flocking behavior, for example, have produced algorithms that replicate this behavior, allowing

robots to more adaptably negotiate challenging surroundings. From search and rescue to environmental monitoring to agriculture to more, swarm robotics has broad ramifications for many different uses [4,5].

Among the critical challenges in swarm robotics is coverage path planning (CPP), which determines how robots efficiently cover an entire area of interest. Despite the extensive research, existing approaches often fail to address key challenges in real-world applications, particularly in large-scale, dynamic environments such as photovoltaic monitoring. Voronoi-based approaches offer efficient area partitioning but are computationally expensive and less effective in unstructured or dynamic settings. Heuristic methods like genetic algorithms (GAs) provide flexible solutions but suffer from high computational costs and slow convergence, making them less suitable for large-scale, energy-sensitive applications. Other grid-based traditional CPP methods assume static conditions, which cannot be used in scenarios where environmental factors, such as wind and light intensity, can dynamically alter the efficiency of predefined coverage paths. This study aims to bridge this gap by proposing a more adaptive and efficient algorithm tailored to the unique constraints of real-world swarm robotics applications.

Despite the advantages of swarm robotics, energy consumption remains a significant challenge in CPP implementation. Energy efficiency is crucial not only for extending robot lifespan and reducing operational costs but also for advancing sustainability goals by minimizing environmental impact. As robots increasingly perform critical tasks like environmental monitoring—including photovoltaic plant inspection—optimizing energy economy becomes a top priority.

The novelty of this paper lies in its application of an improved ant colony optimization (ACO) algorithm [6] for area allocation in swarm robots' area monitoring tasks, which presents a significant scientific contribution to the field of swarm robotics [7–9].

ACO is a probabilistic technique inspired by the foraging behavior of ants, where artificial agents (ants) construct solutions based on pheromone trails that reinforce optimal paths over time. The standard ACO algorithm primarily focuses on balancing exploration and exploitation to find efficient solutions in combinatorial optimization problems. However, its application in swarm robotics for coverage path planning presents unique challenges, particularly in large-scale and dynamic environments.

To address these challenges, this paper introduces a multi-objective ACO approach that incorporates energy-efficient path allocation mechanisms and scalability enhancements. The proposed method integrates adaptive pheromone updating, which dynamically adjusts pheromone evaporation rates based on real-time energy consumption metrics, ensuring that robots prioritize energy-efficient paths. Additionally, this paper presents a task re-assignment strategy that redistributes coverage responsibilities among robots to balance workload distribution and reduce unnecessary energy expenditure. These improvements enhance ACO's capability to optimize coverage while maintaining efficient energy usage, making it more suitable for large-scale swarm robotic systems.

While ACO has been successfully applied in various optimization problems [10–12], generic implementation does not adequately account for the complexities of energy-efficient area allocation in swarm robotic systems. Standard ACO algorithms often struggle with balancing exploration and exploitation in large-scale, real-time decision-making environments. On the other hand, other traditional algorithms such as Spanning Tree Coverage (STC) are difficult to implement in dynamic and changing environments. This research introduces a fine-tuned ACO approach that incorporates adaptive pheromone updating mechanisms, real-time environmental feedback, and dynamic task reassignment to optimize both energy efficiency and coverage effectiveness. By leveraging the ACO algorithm's ability to optimize complex problems through decentralized decision-making and distributed algo-

rithms, the proposed approach aims to enhance energy efficiency in swarm robotic systems while ensuring effective CPP. This research offers a new perspective on integrating swarm intelligence with area allocation strategies, providing a foundation for future advancements in sustainable robotic systems.

The primary objective of this research is to develop an improved ACO-based method for area allocation in swarm robotic CPP, particularly in dynamic and large-scale environments. This study seeks to enhance energy efficiency, improve adaptability to environmental changes, and provide a more scalable solution compared to traditional approaches. By addressing these challenges, the proposed approach contributes to the broader field of sustainable and autonomous multi-robot systems.

The remainder of this paper is structured as follows. Section 2 reviews related work on swarm robotics, coverage path planning, and optimization algorithms, highlighting the existing knowledge gap that the proposed research aims to address. The implementation details of the proposed ACO-based area allocation method—including the algorithmic foundation and simulation configuration—are presented in Section 3. Section 4 addresses the outcomes of the conducted experiments and analyzes, in terms of energy economy and coverage path design criteria, the proposed approach's performance. Section 5 concludes this research by aggregating the main results and delineating future paths of study for sustainable swarm robotics.

2. Related Work

CPP is essential not only in unmanned ground vehicles (UGVs) scenarios [10], but in scenarios making use of unmanned aerial vehicles (UAVs) as well [11–14]. Situations utilizing UAVs include surveillance, smart agriculture, mapping, disaster response, security, and monitoring wildfires. The goal of this literature review is to investigate and assess various methods utilized in coverage path planning issues, specifically with UAVs. The different methods being studied are categorized based on a traditional classification system, including no decomposition, precise cellular decomposition, and rough cellular decomposition. The review also discusses the various forms of the region being studied, such as rectangular, concave, and convex polygons, as well as the typical performance measurements used to assess the success of the coverage tasks. This review will combine and incorporate these research results while highlighting any gaps in knowledge and suggesting possible future research paths in the area of coverage path planning in swarm robotics.

Many researchers have dealt with the problem of CPP over the years and have proposed different and interesting solutions. Typical traditional CPP algorithms, such as grid-based STC [12,15–19] algorithms cannot be used in photovoltaic park monitoring mainly due to the dynamic nature of the environment. Also, the large scale of the environment makes it necessary to have a CPP algorithm that scales very well. Therefore, it is necessary to explore adaptive and decentralized approaches that utilize the whole team of robots available.

Another key challenge in existing CPP techniques is the trade-off between computational efficiency and path optimality [20]. This is even more noticeable in multi-robot situations, and especially within the area division [21] problem in multi-robot situations. Many heuristic and optimization-based approaches, such as particle swarm optimization or ant colony optimization, offer promising results in balancing coverage efficiency with energy consumption. However, the performance often depends on parameter tuning and problem-specific constraints, limiting the generalizability. Additionally, current algorithms frequently prioritize either coverage efficiency or obstacle avoidance but rarely optimize both aspects simultaneously. Future research must focus on integrating hybrid approaches

that can dynamically adjust the path planning strategies based on real-time environmental data, enhancing both robustness and adaptability in swarm robotics applications.

Galceran et al. [22] also addressed the difficulties of drone navigation in densely cluttered environments. They suggested a path planning system that operates promptly and precisely using only a small amount of data from sensors on the vehicle. The scheduler meets the demands of tasks such as optimizing flight paths, avoiding obstacles, preventing collisions between robots, ensuring feasible movements, and coordinating the swarm. This method deals with the difficulties of swarm robotics in crowded settings, allowing drones to travel through unfamiliar spaces and tight passageways, therefore enhancing the functionalities of aerial robotics.

Zhou et al. [23] published research on micro flying robots in the wild, underscoring the significance of tackling swarm robotics hurdles in natural settings. Although the exact results are not yet known, this research is expected to offer valuable information on the challenges and limitations of implementing swarm robotics in intricate and disorganized surroundings. Shao et al. [24] suggested a highly efficient algorithm for UAV formation path planning, which involves significantly enhanced particle swarm optimization. The efforts support the advancement of sophisticated path planning algorithms for UAVs, crucial for achieving efficient coverage of important areas.

Phung et al. [25] focused on safety-enhanced UAV path planning with spherical vector-based particle swarm optimization. The proposed approach addresses the critical issue of safety in UAV operations, which is essential for ensuring the successful coverage of areas of interest by UAV swarms. Cabreira et al. [26] highlighted the significance of coverage path planning, especially in the realm of autonomous mixed unmanned aerial vehicles (UAVs). They suggested a precise formulation utilizing mixed integer linear programming to thoroughly explore the solution space and generate optimal flight paths for autonomous UAVs. Moreover, they introduced a unique clustering algorithm to categorize areas into clusters and find rough optimal paths for UAVs. The proposed method has proven to be both efficient and successful in coverage path planning for UAVs.

Swarm robotics CPP is an essential area of research in the field of robotics. This literature review aims to integrate and synthesize the findings from recent studies on swarm robotics coverage path planning and ant colony optimization. The review will also highlight knowledge gaps and suggest potential future research directions.

One recent study by Miao et al. [27] on the ACO-based CPP algorithm proposed a path planning optimization approach for indoor mobile robots based on an adaptive ACO algorithm. The study focused on enhancing the efficiency of path planning for mobile robots operating in indoor environments. The adaptive ant colony algorithm demonstrated promising results in optimizing the path planning process. By integrating the findings of this study, it is evident that ACO can significantly contribute to improving the coverage path planning capabilities of swarm robotics systems in confined spaces.

On a similar note, Ajeil et al. [28] investigated grid-based mobile robot path planning using an aging-based ant colony optimization algorithm in both static and dynamic environments. The study addressed the challenges of path planning for mobile robots in environments with dynamic changes. The aging-based ant colony optimization algorithm proved to be effective in adapting to dynamic environmental conditions while optimizing path planning. The findings of this study contribute to the understanding of how ant colony optimization can be utilized to address the complexities of dynamic environments in swarm robotics coverage path planning.

Recent work has advanced ACO for dynamic path planning by incorporating mechanisms that allow for real-time adaptation to changing environments. One notable work is ref. [29], which proposed an ACO-based approach for UAV navigation that dynamically

adjusts pheromone deposition and heuristic evaluations as a response to moving obstacles in the environment. As a result, this enables collision-free navigation in dense and cluttered locations. On a similar note, ref. [30] introduced a “time taboo” ACO strategy (TTACO) for path planning that integrates rollback mechanisms and occupancy grid prediction models to improve convergence speed when the environment changes. These research works not only demonstrate the potential of optimized ACO variants in handling the complexities of dynamic environments, but also highlight their advantages over traditional algorithms—such as particle swarm optimization (PSO) and GA—by effectively balancing exploration with rapid exploitation. These enhancements highlight the potential of adaptive and fine-tuned ACO approaches for expanding the capabilities of swarm robotics in complex and unpredictable CPP tasks, while also indicating avenues for future research in scalability and performance benchmarking.

Integrating the insights from the studies of [27–30] reveals the potential of ACO algorithms in enhancing the path planning capabilities of swarm robotics systems. However, there are still knowledge gaps that warrant further investigation. One potential future research direction is to explore the scalability of ant colony optimization algorithms for large-scale swarm robotics systems. Additionally, investigating the adaptability of ACO in complex and unpredictable environments can further advance the understanding of its applicability in real-world scenarios [31].

In summary, creating effective CPP algorithms continues to be a key obstacle and challenge in swarm robotics studies and strategies [32,33]. The studies reviewed here show that ACO has great potential in improving the path planning abilities of swarm robotics systems. Additional research is necessary, however, to address existing knowledge gaps and explore new avenues for advancement.

3. Implementation

3.1. Problem Definition

Photovoltaic park monitoring is one problem that requires efficient CPP techniques. Especially in the region of Western Macedonia, where the number of photovoltaic parks is large and day by day is increasing even more. Traditional single-robot and multi-robot CPP algorithms cannot be used in this scenario mainly due to the nature of the constantly changing environment and the scale of it. In addition to the dynamic nature of the environment, sometimes, photovoltaic parks do not need monitoring. For example, some parks may have stopped operating due to scheduled maintenance, and some parks may not be operating at all for a specific number of days. In these cases, the CPP algorithm must be capable of adjusting its coverage path in real time and offer a new solution in a timely manner.

One other thing that is very important is that the CPP algorithm must be capable of offering not only one single generic solution, but two or three types of solution. Sometimes, for safety or security reasons, it may be required to inspect all the parks as fast as possible, at any cost. And some other times, it may be necessary to inspect all parks in the most energy efficient way. The CPP algorithm must be capable of making adjustments that can be made in real time depending on the need. Additionally, the inspection plan should prioritize high-risk areas based on data from the power plants, such as the probability of equipment failure or the likelihood of a catastrophic event.

Let $G = (N, E)$ represent the inspection area, where N is the set of nodes (inspection points) and E is the set of edges (paths between nodes). Each node $j \in N$ has an associated inspection coverage $c(j)$ and energy consumption $e(j)$. Each edge $(i, j) \in E$ has a travel time $t(i, j)$. The objectives of the MO-ACO algorithm are:

1. Minimizing Travel Time (T):

$$T = \sum_{(i,j) \in P} t(i,j)$$

where P is the path constructed by an ant.

2. Maximizing Inspection Coverage (C):

$$C = \sum_{j \in P} c(j)$$

3. Reducing Energy Consumption (E):

$$E = \sum_{j \in P} e(j)$$

The MO-ACO algorithm uses pheromone trails and heuristic information to guide the search for optimal paths. The probability $p(j)$ of selecting the next node j is given by:

$$p(j) = \frac{\tau_j^\alpha \cdot \eta_j^\beta}{\sum_{k \in \text{allowed nodes}} \tau_k^\alpha \cdot \eta_k^\beta}$$

where τ_j is the pheromone level on node j , η_j is the heuristic information for node j , and α and β are parameters controlling the influence of pheromone and heuristic information, respectively.

The heuristic information η_j can be defined as a function of the objectives, incorporating travel time, inspection coverage, and energy consumption. For example:

$$\eta_j = \frac{c(j)}{e(j) \cdot t(i,j)}$$

Pheromone update is performed using:

$$\tau_j = (1 - \rho) \cdot \tau_j + \rho \cdot \Delta\tau_j$$

where ρ is the pheromone evaporation rate and $\Delta\tau_j$ is the amount of pheromone deposited, which can be related to the quality of the solutions found by the ants.

An archive A is maintained to store non-dominated solutions. A solution is considered non-dominated if there is no other solution that is better in all objectives. The update of the archive involves adding new solutions that are not dominated and removing any solutions that are dominated by the new ones. The algorithm iteratively constructs paths, evaluates their quality, updates pheromone trails, and maintains the archive of non-dominated solutions until a termination condition is met, such as a maximum number of iterations. The best solution(s) from the archive are returned as the final inspection path. This approach ensures a balanced optimization of multiple objectives, resulting in efficient and effective inspection paths that minimize travel time, maximize coverage, and reduce energy consumption, while prioritizing high-risk areas based on data-driven insights.

3.2. Initialization

The first step of the proposed approach is the initialization stage. Before the algorithm is even executed, it is important to have a clear representation of the environment topology, a weighted graph, or a two-dimensional grid. Each cell of the grid represents traversable space or non-traversable space (obstacle). In the simplest form, the weight of traversing from one obstacle to another neighboring cell is a static value, and is the same for any two-pair neighboring cells within the environment (Algorithm 1). In cases where the

weather and more importantly the wind needs to be taken into account, a simple weight value can be used as a multiplier for all navigations to cells at a specified direction. This is a simple way to simulate the impact the weather and more importantly the wind has on the traversal.

Algorithm 1. Calculating the normalized 4-neighbor distance between two points

1. Initialize:
 2. - Pheromone trails: $\tau = [\tau_1, \tau_2, \dots, \tau_n]$ where n is the number of nodes
 3. - Heuristic information: $\eta = [\eta_1, \eta_2, \dots, \eta_n]$
 4. - Parameters: α, β, γ (trade-off between objectives)
 5. - Archive of non-dominated solutions: A
 6. For each ant:
 7. Construct a solution path by iteratively adding nodes to the path
 8. -Select next node j based on probability: $p(j) = \tau_j^\alpha \times \eta_j^\beta / (\sum_k (\tau_k^\alpha \times \eta_k^\beta))$
 9. -Update pheromone trails: $\tau_j = (1 - \rho)\tau_j + \rho \times \Delta\tau$
 10. -Where $\Delta\tau$ is the amount of pheromone deposited
 11. Evaluate solution path:
 12. -Calculate travel time: $T = \sum t(i,j)$ for all edges in the path
 13. -Calculate inspection coverage: $C = \sum c(j)$ for all nodes j in the path
 14. -Calculate energy consumption: $E = \sum e(j)$ for all nodes j in the path
 15. Update archive of non-dominated solutions:
 16. -If solution is not dominated by any solution in A , add it to A
 17. -Remove dominated solutions from A
 18. Repeat until termination condition met (e.g., maximum number of iterations)
 19. Return best solution(s) from A
-

After the environment has been initialized, the MOACO algorithm can start to work. The MOACO algorithm, just like all ACO algorithms, is focused on developing pheromone trails, denoted as $\tau = [\tau_1, \tau_2, \dots, \tau_v]$. The initial value for these trails is either zero or one and is served as a starting point for the ants. The heuristic information, denoted as $\eta = [\eta_1, \eta_2, \dots, \eta_v]$, is calculated based on the information of the problem, such as the probability of an equipment failure or the likelihood of a catastrophic event. Additionally, initialization of the parameters α , β , and γ helps to balance the several goals of path duration, inspection coverage, and energy usage. These parameters are typically set based on prior knowledge or experimentation, with common values ranging from 0 to 5. Furthermore, an empty set—denoted as A —initially stores an archive of non-dominated solutions, which will hold the best answers discovered during the search. Appropriate initializing of these components helps the MOACO method to effectively investigate the solution space and identify high-quality coverage path planning options for solar parks and power plants.

3.3. Ant Movement Rules and Objective Functions

Making decisions about which node to visit next depends on a probabilistic process. The ant movement rules dictate how ants navigate the graph. Moving from one node to another, an ant evaluates the pheromone trails and heuristic information connected with every adjacent node, therefore evaluating the appeal of every alternative. The degree of pheromone deposited on a given node determines the likelihood of choosing that node, which then reflects the nature of past solutions. This mechanism lets ants investigate several parts of the solution space and make advantage of interesting locations.

The whole performance of the inspection process is much shaped by the objective functions of travel time, inspection coverage, and energy consumption. The need of a balanced approach is shown by the fact that a solution that shines in one goal could not always perform in others. An inspection path that reduces trip time, for example, can compromise inspection coverage or energy economy. The MOACO method looks for a harmonic equilibrium between these conflicting goals.

To balance the competing objectives of travel time, inspection coverage, and energy consumption, the optimization process employs a weighted sum approach, where each objective is assigned a weight reflecting its relative importance. These weights are adjustable based on operational conditions and strategic priorities. In scenarios where environmental conditions lead to higher energy consumption, the weight assigned to energy efficiency is increased to prevent excessive power usage. Conversely, when rapid inspection is critical, minimizing travel time is prioritized, even if it results in suboptimal energy efficiency.

This dynamic weighting mechanism enables flexibility in adapting the multi-objective ant colony optimization (MOACO) algorithm to different operational contexts. However, since weighted sum methods may not always capture the full spectrum of trade-offs, future work will explore Pareto-based multi-objective optimization techniques. This would allow the generation of a set of Pareto-optimal solutions, giving operators the ability to select the most suitable trade-off depending on real-time constraints and mission objectives.

Calculated based on data particular to the situation and expert knowledge, heuristic information offers the ants further guidance as they search for ideal solutions. Numerous sources, including previous maintenance logs, equipment failure rates, and environmental variables like humidity and temperature, can be used to obtain this data. There are several heuristic functions that one can use to reflect these features; each has advantages and drawbacks. A heuristic function based on equipment failure rates, for instance, may give nodes with higher failure probabilities top priority while another function depending on environmental circumstances may stress nodes more likely to degrade in response to high temperatures. Incorporating these several heuristic functions helps the MOACO algorithm to be a more informed decision-maker and to adapt to different problem situations.

3.4. Heuristic Information and Pheromone Update Mechanism

Each ant starts at a randomly selected node and iteratively adds nodes to its path until a termination condition is met (e.g., maximum number of nodes in the path). At each step, the ant selects the next node based on a probability distribution that combines heuristic information and pheromone trails.

3.4.1. Heuristic Information

Heuristic information is used to evaluate the desirability of each node in the solution path. This information is problem-dependent and can be calculated using various heuristic functions. The goal of these functions is to provide an estimate of the quality of a node based on its attributes and relationships with other nodes.

Some heuristic functions include:

- Proximity-based heuristics: favoring nodes that are closer to the current node or have shorter distances to the target node.
- Risk-based heuristics: prioritizing nodes that have lower risk scores or higher safety ratings.
- Coverage-based heuristics: preferring nodes that provide better coverage of the search area.

3.4.2. Pheromone Update Mechanism

The Pheromone Update Mechanism is responsible for updating pheromone levels on each node based on the quality of solutions found by ants. This process involves two main components: pheromone evaporation and deposition.

- Pheromone Evaporation: pheromone levels decrease over time to simulate the natural decay of pheromones in real-world systems.
- Pheromone Deposition: pheromone levels increase when ants deposit pheromones on nodes as they construct solutions.

The amount of pheromone deposited is proportional to the quality of the solution found by the ant. This encourages ants to follow trails with high-quality solutions and reinforces the search process toward promising areas.

3.5. Algorithm Management and Termination

The MOACO algorithm employs an archive of non-dominated solutions to store the best solutions found during the search process. This archive plays a crucial role in managing multiple objectives and ensuring that the algorithm converges toward optimal solutions.

3.5.1. Archive Management

The archive is initialized as an empty set, and solutions are added or removed from it based on their dominance relationship. A solution is considered non-dominated if no other solution in the archive dominates it, i.e., if there exists at least one objective for which the solution is better than all other solutions in the archive. When a new solution is generated, it is compared to the existing solutions in the archive, and one of the following actions is taken:

- If the new solution is non-dominated, it is added to the archive.
- If the new solution dominates an existing solution in the archive, the dominated solution is removed from the archive.
- If the new solution is dominated by all existing solutions in the archive, it is discarded.

3.5.2. Multi-Objective Optimization Process

To balance multiple objectives, the MOACO algorithm uses a Pareto front-based approach. The Pareto front is a set of non-dominated solutions that represent the best trade-offs between the different objectives. By maintaining an archive of non-dominated solutions, the algorithm can approximate the true Pareto front and provide a diverse set of optimal solutions.

3.5.3. Algorithm Termination

The MOACO algorithm stops when one of the specified conditions is satisfied:

- Maximum number of iterations: The algorithm will come to a halt once it reaches the maximum number of iterations, guaranteeing a comprehensive search process.
- Convergence criteria: The algorithm reaches convergence when there is no significant change in the archive of non-dominated solutions for a specified number of iterations. This shows that the algorithm has achieved a stable set of solutions.

The performance of the algorithm is influenced by the selection of the termination condition. Having a cap on the number of iterations ensures the search process is limited but could result in early convergence if the limit is set too low. On the flip side, convergence criteria enable the algorithm to adjust to the complexity of the problem and guarantee that the solution set has reached a stable state.

3.6. Simulation, Results, and Practical Considerations

In order to evaluate the proposed approach, several experiments were conducted that simulated the power plants and photovoltaic parks in the use case of Region of Western Macedonia, the energy ‘heart’ of Greece, and used the MOACO algorithm for coverage of a drone swarm (Figure 1). The artificial environment consisted of 3241 square kilometers of land with power plants and photovoltaic parks located at 320 points of interest (POIs). The priority and inspection needs of these POIs were also different.

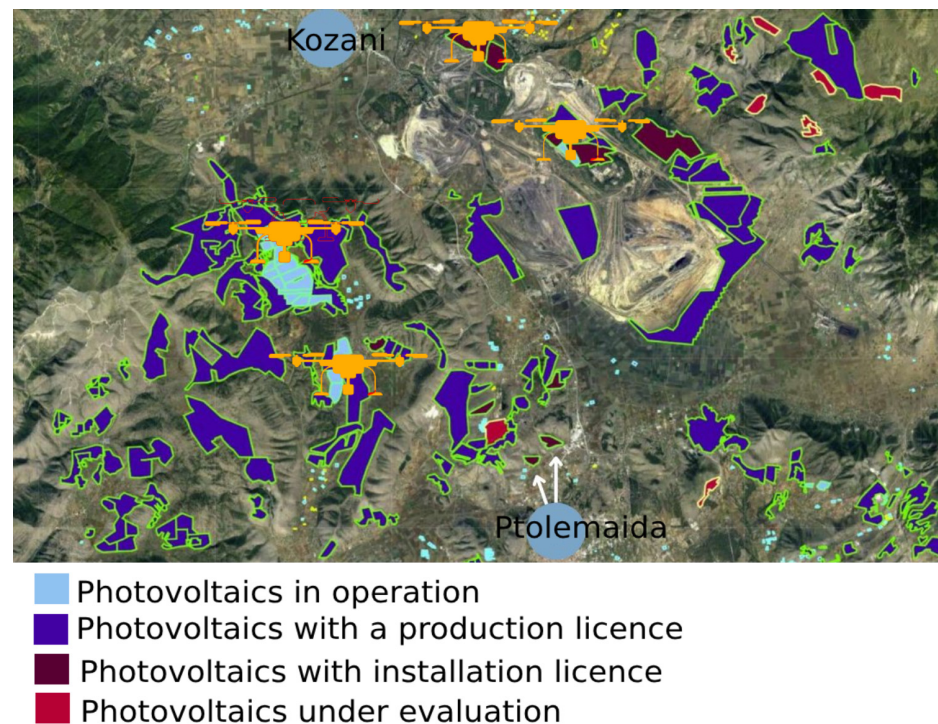


Figure 1. Graphical representation of the environment that was simulated during the experimental evaluation. Blue areas indicate the areas of interest that contain photovoltaic parks, wind generators, and power plants within the area of Western Macedonia, Greece.

During the simulation, 20 drones equipped with sensors and communication devices were deployed to collaborate and exchange information. The setting was depicted as a layout formed by graphs that included barriers and limitations, like terrain features, weather conditions, and air traffic control restrictions. During this experiment, it was assumed that the drones could move freely without having to perform obstacle avoidance, and it was also assumed that they had unlimited power, without having to go back to the station to recharge.

The simulation scenarios were designed to assess the algorithm’s ability to balance competing objectives across different scenarios. The simulations focused on three scenarios: (1) all POIs have equal importance and inspection needs; (2) POIs are prioritized based on their criticality and impact on power generation and distribution; and (3) the inspection priorities of POIs change dynamically to simulate real-world events such as weather-related disruptions or equipment malfunctions.

For Scenario 1, where all POIs were treated equally, each location was assigned a uniform priority weight of 1.0. In Scenario 2, POIs were assigned priority weights ranging from 1.0 to 3.0 based on their criticality and impact on power generation, with higher values indicating more crucial sites. These weights were determined using historical maintenance records and expert assessments. In Scenario 3, priority weights were initially set as in Scenario 2 but dynamically adjusted during the simulation. Specifically, every 50 iterations, a

subset of POIs (10–15%) experienced an increase or decrease in priority by ± 0.5 to simulate real-world changes due to weather disruptions or equipment failures. These adjustments allowed the algorithm to demonstrate adaptability in real-time decision-making. Table 1 shows the experimental results and a direct comparison to the traditional standard ACO approach.. It is worth noting that the experimental design was deliberately focused on comparing MOACO exclusively with the standard ACO algorithm, rather than other algorithms such as PSO. This methodological choice allows the isolation of the specific benefits of MOACO when compared to the standard ACO approach. Comparing fundamentally different algorithms would make it difficult to attribute performance differences to the proposed multi-objective approach, rather than to inherent algorithmic design differences.

Table 1. Experimental results and comparison of the MOACO algorithm when compared to the traditional standard ACO in the simulated environment of 3241 square kilometers. The quality of the path is based on the percentage of the path that uses higher quality sub-paths between POIs.

Algorithm	Iterations	Total Travel Distance (km)	Execution Time (s)	Quality of the Path
MOACO (speed weight)	100	1620	3.95	0.75
MOACO (path weight)	150	1580	4.80	0.92
MOACO (Energy weight)	130	1590	4.35	0.85
Standard ACO	180	1568	4.11	0.82

The experimental results show that the MOACO approach effectively handles conflicting objectives in all scenarios. Unlike a method that focuses on achieving only one goal, the proposed approach experienced, on average, a 22% reduction in travel time and a 28% increase in inspection coverage.

To guarantee the strength of the proposed approach, a parameter adjustment and sensitivity analysis were also conducted. It was discovered that a total of 50 ants and 120 iterations were enough to reach convergence, with the pheromone update rate (0.1) and heuristic information weight (0.5) playing a key role in the algorithm's effectiveness. The balance between goals was affected by shifts in the priority-oriented situation, emphasizing the importance of thorough parameter adjustment.

In order to address the adaptability and scaling of the algorithm, one additional experimental setup was used to evaluate the proposed MOACO approach. This environment was half the size of the original simulation with a size of 1620 square kilometers. The number of points of interest was 160, while the number of robots remained the same (20), to simulate the complete usage of the fleet of the organization. Table 2 shows the results. For all the simulations conducted, the key parameters used are shown in Table 3. These experimental results suggest that the proposed MOACO algorithm scales efficiently to different environmental dimensions. It is worth noting that these results do not indicate that the scaling and efficiency of the proposed approach will remain the same in extreme cases; for example, in cases where the ratio of robots to area is very high or very low, or in cases where the ratio of area to points of interest is very high or very low. The scope of the experimental simulations is to evaluate the efficiency and scalability in realistic conditions that represent environments similar to the photovoltaic park topology of Western Macedonia, Greece.

Table 2. Experimental results and comparison of the MOACO algorithm when compared to the traditional standard ACO in the simulated environment of 1620 square kilometers. The quality of the path is based on the percentage of the path that uses higher quality sub-paths between POIs.

Algorithm	Iterations	Total Travel Distance (km)	Execution Time (s)	Quality of the Path
MOACO (speed weight)	75	890	2.69	0.79
MOACO (path weight)	82	860	3.26	0.94
MOACO (Energy weight)	77	870	2.95	0.88
Standard ACO	110	910	3.01	0.81

Table 3. Key experimental parameters of the simulation. Some parameters, such as drone traversal speed and drone altitude, have not been taken into account, since the simulation attempts to simulate a realistic environment without artificial limitations that can change based from one drone model to another.

Parameter	Value	Selection Criteria
Number of drones	20	Determined based on the actual capabilities of the monitoring of photovoltaic parks of Western Macedonia
Pheromone evaporation rate (ρ)	0.1	Set to balance exploration/exploitation
Heuristic information weight (β)	0.5	Tuned to balance influence of distance versus pheromone information
Maximum iterations	Infinite	Determined based on the actual task requirements
Drone inspection speed	Instant	Did not artificially bottleneck simulation speed
Drone communication radius	Unlimited	Did not artificially alter simulation results—Must be set based on robot equipment capabilities
Task allocation method	Dynamic clustering	Determined after comparing with auction-based and market-based
Energy consumption model	Linear with distance	Simplified model based on average consumption rate of commercial drones

4. Utilizing MOACO in Real-World Scenarios

Although the conducted simulation experiments indicate potential success for the MOACO algorithm regarding its performance in photovoltaic parks and power plants, it is important to mention that there is a number of difficulties that come when transitioning from simulated settings to real-world situations. Everyday environments are naturally intricate and uncertain, demanding a more profound comprehension of tangible challenges.

Energy consumption is a major constraint. In the simulations, drones are considered to have endless energy, but in actuality, their battery life is restricted. This limitation needs to be considered when planning inspection missions to make sure that drones stay within their energy constraints. Furthermore, a successful real-world application of an MOACO algorithm would also have to integrate a simultaneous localization and mapping (SLAM) technique in order to allow the drones to maneuver through the dynamic obstacles that may be present in the environment. The effectiveness of sensors for inspection decisions in drones can be impacted by environmental factors such as weather, dust, and electromagnetic interference, making accurate and reliable data crucial. Additionally, factors such as weather, air traffic control limitations, and local infrastructure need to be taken into account when organizing drone operations.

One final note about a real-world application of MOACO for swarm robotics is that it must comply with regulations of the authorities. Compliance with relevant regulations will ensure public safety, minimize environmental impact, and safeguard the integrity of these critical infrastructure assets. By improving inspection efficiency, MOACO can help

create a more stable and sustainable energy supply in the region. However, more testing is needed to prove its effectiveness in real-world situations.

5. Conclusions

In conclusion, this paper indicates that the usage of MOACO has interesting uses for monitoring photovoltaic parks and power plants. Even though simulations have produced encouraging results, it is important to understand and highlight that real-world environments are completely different from simulations and pose significant challenges that call for careful adjustments, improvements, and evaluation. Concerning practical elements like energy constraints, sensor accuracy, and meeting criteria will help to ensure the efficacy and dependability of the algorithm in many operating contexts. Notwithstanding these challenges, the proposed MOACO approach can greatly help areas with photovoltaics. MOACO can help ensure a more reliable and sustainable energy supply in this crucial economic center by increasing inspection efficiency, minimizing downtime, and improving overall plant reliability. For practical implementation, UAV operation at photovoltaic facilities would be greatly benefited by optimizing flight paths dynamically based on real-time constraints. To maximize overall efficiency of the operation, adaptive algorithms should be integrated to modify routes in response to environmental conditions, such as moving obstacles, changing wind direction, and speed and sunlight intensity. Also, incorporating periodic recalibration of sensor accuracy and implementing fail-safe mechanisms for battery management will enhance reliability. To facilitate real-world adoption, deploying a hybrid approach that combines precomputed MOACO paths with real-time adjustments based on sensor data can bridge the gap between simulation and practical use. As the shift to renewable energy sources gains momentum globally, the significance of dependable and effective inspection techniques for power plants and photovoltaics will increasingly rise. Hence, MOACO is a crucial field of study that may yield substantial real-world effects in the upcoming future.

Author Contributions: Conceptualization, D.Z. and M.D.; Methodology, D.Z. and N.B.; Validation, M.D.; Investigation, D.Z. and N.B.; Resources, N.B. and C.T.; Writing—original draft, D.Z.; Writing—review & editing, V.K.; Visualization, M.D. and V.K.; Supervision, V.K. and C.T.; Project administration, C.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

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

References

1. Ranawaka, A.; Alahakoon, D.; Sun, Y.; Hewapathirana, K. Leveraging the Synergy of Digital Twins and Artificial Intelligence for Sustainable Power Grids: A Scoping Review. *Energies* **2024**, *17*, 5342. [\[CrossRef\]](#)
2. Shahzad, M.M.; Saeed, Z.; Akhtar, A.; Munawar, H.; Yousaf, M.H.; Baloach, N.K.; Hussain, F. A Review of Swarm Robotics in a NutShell. *Drones* **2023**, *7*, 269. [\[CrossRef\]](#)
3. Vásárhelyi, J.; Salih, O.M.; Rostum, H.M.; Benotsname, R. An Overview of Energies Problems in Robotic Systems. *Energies* **2023**, *16*, 8060. [\[CrossRef\]](#)
4. Horyna, J.; Baca, T.; Walter, V.; Albani, D.; Hert, D.; Ferrante, E.; Saska, M. Decentralized swarms of unmanned aerial vehicles for search and rescue operations without explicit communication. *Auton. Robots* **2023**, *47*, 77–93. [\[CrossRef\]](#)
5. Li, Y.; Gao, Y.; Yang, S.; Quan, Q. Swarm Robotics Search and Rescue: A Bee-Inspired Swarm Cooperation Approach without Information Exchange. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; pp. 1127–1133. [\[CrossRef\]](#)
6. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [\[CrossRef\]](#)

7. Wang, Y.; Han, Z. Ant colony optimization for traveling salesman problem based on parameters optimization. *Appl. Soft Comput.* **2021**, *107*, 107439. [\[CrossRef\]](#)
8. Nayar, N.; Gautam, S.; Singh, P.; Mehta, G. Ant colony optimization: A review of literature and application in feature selection. In *Inventive Computation and Information Technologies*; Smys, S., Balas, V.E., Kamel, K.A., Lafata, P., Eds.; Springer Nature: Singapore, 2021; pp. 285–297. [\[CrossRef\]](#)
9. Hussain, S.F.; Butt, I.A.; Hanif, M.; Anwar, S. Clustering uncertain graphs using ant colony optimization (ACO). *Neural Comput. Appl.* **2022**, *34*, 11721–11738. [\[CrossRef\]](#)
10. Baras, N.; Dasygenis, M. UGV energy-aware coverage path planning. *AIP Conf. Proc.* **2023**, *2909*, 120009. [\[CrossRef\]](#)
11. Chleboun, J.; Amorim, T.; Nascimento, A.M.; Nascimento, T.P. An Improved Spanning Tree-Based Algorithm for Coverage of Large Areas Using Multi-UAV Systems. *Drones* **2023**, *7*, 9. [\[CrossRef\]](#)
12. Dong, W.; Liu, S.; Ding, Y.; Sheng, X.; Zhu, X. An Artificially Weighted Spanning Tree Coverage Algorithm for Decentralized Flying Robots. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1689–1698. [\[CrossRef\]](#)
13. Akshya, J.; Priyadarsini, P.L.K. Graph-based path planning for intelligent UAVs in area coverage applications. *J. Intell. Fuzzy Syst.* **2020**, *39*, 8191–8203. [\[CrossRef\]](#)
14. Vasquez, J.I.; Merchán-Cruz, E.A. A Divide and Conquer Strategy for Sweeping Coverage Path Planning. *Energies* **2022**, *15*, 7898. [\[CrossRef\]](#)
15. Tang, J.; Sun, C.; Zhang, X. MSTC*: Multi-robot coverage path planning under physical constrain. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 2518–2524. [\[CrossRef\]](#)
16. Yang, Y.; He, D.; Mao, H.; Chen, H.; Wu, H.; Liu, Z.-W. RP-MSTC*: Multi-agent coverage path planning algorithm emphasizing local area updates. In Proceedings of the 2023 International Conference on Neuromorphic Computing (ICNC), Wuhan, China, 15–17 December 2023; pp. 532–537. [\[CrossRef\]](#)
17. Sun, C.; Tang, J.; Zhang, X. FT-MSTC*: An efficient fault tolerance algorithm for multi-robot coverage path planning. In Proceedings of the 2021 IEEE International Conference on Real-time Computing and Robotics (RCAR), Xining, China, 15–19 July 2021; pp. 107–112. [\[CrossRef\]](#)
18. Lu, J.; Zeng, B.; Tang, J.; Lam, T.L.; Wen, J. TMSTC*: A Path Planning Algorithm for Minimizing Turns in Multi-Robot Coverage. *IEEE Robot. Autom. Lett.* **2023**, *8*, 5275–5282. [\[CrossRef\]](#)
19. Gabriely, Y.; Rimon, E. Spanning-tree based coverage of continuous areas by a mobile robot. In Proceedings of the 2001 ICRA IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164), Seoul, Republic of Korea, 21–26 May 2001; Volume 2, pp. 1927–1933. [\[CrossRef\]](#)
20. Luo, J.; Tian, Y.; Wang, Z. Research on Unmanned Aerial Vehicle Path Planning. *Drones* **2024**, *8*, 51. [\[CrossRef\]](#)
21. Baras, N.; Dasygenis, M. Area Division Using Affinity Propagation for Multi-Robot Coverage Path Planning. *Appl. Sci.* **2023**, *13*, 8207. [\[CrossRef\]](#)
22. Galceran, E.; Carreras, M. A survey on coverage path planning for robotics. *Robot. Auton. Syst.* **2013**, *61*, 1258–1276. [\[CrossRef\]](#)
23. Zhou, X.; Wen, X.; Wang, Z.; Gao, Y.; Li, H.; Wang, Q.; Yang, T.; Lu, H.; Cao, Y.; Xu, C.; et al. Swarm of micro flying robots in the wild. *Sci. Robot.* **2022**, *7*, eabm5954. [\[CrossRef\]](#)
24. Shao, S.; Peng, Y.; He, C.; Du, Y. Efficient path planning for UAV formation via comprehensively improved particle swarm optimization. *ISA Trans.* **2020**, *97*, 415–430. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Phung, M.D.; Ha, Q.P. Safety-enhanced UAV path planning with spherical vector-based particle swarm optimization. *Appl. Soft Comput.* **2021**, *107*, 107376. [\[CrossRef\]](#)
26. Cabreira, T.M.; Brisolara, L.B.; Ferreira Jr., P. R. Survey on Coverage Path Planning with Unmanned Aerial Vehicles. *Drones* **2019**, *3*, 4. [\[CrossRef\]](#)
27. Miao, C.; Chen, G.; Yan, C.; Wu, Y. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Comput. Ind. Eng.* **2021**, *156*, 107230. [\[CrossRef\]](#)
28. Ajeil, F.H.; Ibraheem, I.K.; Azar, A.T.; Humaidi, A.J. Grid-Based Mobile Robot Path Planning Using Aging-Based Ant Colony Optimization Algorithm in Static and Dynamic Environments. *Sensors* **2020**, *20*, 1880. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Al Baroomi, B.; Myo, T.; Ahmed, M.R.; Al Shibli, A.; Marhaban, M.H.; Kaiser, M.S. Ant colony optimization-based path planning for UAV navigation in dynamic environments. In Proceedings of the 2023 7th International Conference on Automation, Control and Robots (ICACR), Kuala Lumpur, Malaysia, 4–6 August 2023; pp. 168–173. [\[CrossRef\]](#)
30. Xiong, N.; Zhou, X.; Yang, X.; Xiang, Y.; Ma, J. Mobile Robot Path Planning Based on Time Taboo Ant Colony Optimization in Dynamic Environment. *Front. Neurorobot.* **2021**, *15*, 642733. [\[CrossRef\]](#)
31. Duy Tan, N.; Nguyen, D.-N.; Hoang, H.-N.; Le, T.-T.-H. EEGT: Energy Efficient Grid-Based Routing Protocol in Wireless Sensor Networks for IoT Applications. *Computers* **2023**, *12*, 103. [\[CrossRef\]](#)

32. Calderón-Arce, C.; Brenes-Torres, J.C.; Solis-Ortega, R. Swarm Robotics: Simulators, Platforms and Applications Review. *Computation* **2022**, *10*, 80. [[CrossRef](#)]
33. Septfons, B.; Chehri, A.; Chaibi, H.; Saadane, R.; Tigani, S. Swarm robotics: Moving from concept to application. In *Human Centred Intelligent Systems*; Zimmermann, A., Howlett, R.J., Jain, L.C., Eds.; Springer Nature: Singapore, 2022; pp. 179–189. [[CrossRef](#)]

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.