Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 11, No. 03, September 2025.

ON THE OPTIMIZATION OF UAV SWARM ACO-BASED PATH PLANNING

Areej J. Alabbadi[1] and Belal H. Sababha[2]

(Received: 26-Jan.-2025, Revised: 13-Apr.-2025, Accepted: 6-May-2025)

ABSTRACT

Unmanned Aerial Vehicles (UAVs) play a crucial role in various operations, especially where human life must be protected. Efficient path planning and autonomous coordination are critical for UAV swarms in dynamic 3D cooperative missions, where real-time adaptability is essential. This work addresses the challenge of optimizing UAV swarm operations by proposing a novel hybrid navigation system based on Ant Colony Optimization (ACO). The system efficiently balances path optimization with dynamic formation control, adapting to mission-specific requirements. A key contribution is the hybrid navigation approach, which prioritizes the desired formation of the swarm or the path length and flight time through a threshold- based mechanism, allowing real-time adaptation to changing environments. The system also introduces a comprehensive cost function that evaluates the quality of the path, time consumption, mission completeness and formation divergence. The experiments show that the system consistently provides high-quality paths, achieving around 97% path quality in most cases and never declines below 90%, even in challenging scenarios. The collision avoidance module ensures the completeness of the 100% mission, successfully navigating drones around obstacles and maintaining an optimal path. Furthermore, the formation conservation mechanism effectively maintained the desired swarm configurations while dynamically adapting to obstacles, with the formation change staying within 30% of the allowable range in most scenarios, highlighting the system's ability to preserve the desired formation even in dynamic environments. This research advances UAV swarm intelligence, enabling efficient and autonomous operations in complex 3D environments for diverse cooperative missions. The system's adaptability to formation requirements opens new possibilities for UAV swarm applications, improving navigation efficiency and enhancing formation control.

KEYWORDS

Ant colony optimization (ACO), 3D dynamic environment, UAV swarm, Hybrid navigation approach, Collision avoidance mechanism.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) are revolutionizing the industry. They enable rapid and more cost-effective completion of industrial activities while ensuring safety primarily due to their small size, affordable density and general simplicity of management and operation [1]. UAVs are an effective tool for carrying out operations in locations that are difficult to access. Performing in groups or swarms offers additional benefits. The ability to perform tasks that require flying over large areas, reducing the time required for specific operations, area coverage and coordinated impacts are only a few of the operational benefits that UAV swarms have over non-swarm systems [2]-[5].UAV swarms leverage aerial mobility, high-speed maneuverability and extensive coverage capabilities, making them essential for a variety of applications [6]-[8]. Hundreds of thousands of agents can collectively be controlled by swarm systems, while a single operator or a small team is focused on carrying out mission objectives. Humans can maintain operational control while delegating low-level routine choices to UAV agents. UAV swarms can provide the capability for quick communication and decision-making, as detailed in [3]. A UAV swarm is considerably more effective than one or even several human decision-makers in many situations. Because of many advantages, autonomous swarms are often much more effective, timely and responsive than human or human-operated robot groups.

Centralized and distributed control architectures are the two main categories into which cooperative multi-UAV autonomous control architectures are typically classified [9]. With the benefit of obtaining a globally optimal solution, the centralized-control method has dominated early research. However, this strategy has a fundamental weakness: the multi-UAV system will become uncontrollable should the decision-making layer fail due to the high dependence on the communication link. The distributed

1. A. J. Alabbadi (Corresponding Author) is with Electrical Engineering Department, Princess Sumaya University for Technology, Amman, 11941, Jordan. Email: are20208172@std.psut.edu.jo
2. B. H. Sababha is with Computer Engineering Department, Princess Sumaya University for Technology, Amman, 11941, Jordan.

control approach, which has the advantages of increased dependability, less computation and communication, becomes a study focus as UAV performance and autonomous capabilities develop [10]-[11].

The capability of assigning targets and building a 3D trajectory for each UAV in the swarm is an essential part of its operation. The general problems associated with 3D path planning for a single UAV have been addressed using a variety of techniques, including probabilistic road maps, A* algorithms, artificial potential fields, probabilistic navigation functions and many other techniques [12]-[15]. Most of these algorithms use sampling-based and graph-based search techniques, which work well in high-dimensional configuration spaces and are relatively simple to implement. It is also known that, given enough time, they are probabilistically completed in a way that increases the probability of discovering a solution. Many of these techniques have drawbacks, such as potential exposure to local minima and limitations imposed by constraints connected to the grid's properties. These algorithms often need a balance between exploration and exploitation and are computationally demanding. Some of these algorithms lack robustness, which prevents them from functioning in situations with various dynamic obstacles and automated real-time applications [16]-[17].

Kennedy and Eberhart presented an introductory book on swarm intelligence, based on previous work on robot control and decentralized AI [18]. Swarm intelligence is the intelligent behavior that results from a collection of independent, heterogeneous agents acting as one system. In terms of the distribution of organizational structure, simplicity of individuals, flexibility of the action mode, and establishment of Swarm Intelligence (SI), various social organisms in nature (such as ant colonies, bee colonies, fish schools, and wolf packs) exhibit many characteristics that UAV swarms share [19]. The swarm can be conceptualized as a single entity or system in which intelligence develops through the specific behaviors of a group of people [20]. To develop novel distributed integrated algorithms for UAV swarm cooperative mission planning, some researchers simulated the sophisticated and structured collective behaviors of social organisms.

This research makes several significant contributions to the field of UAV swarm intelligence and distribution for cooperative missions. First, an ACO-based path-planning algorithm is developed. Then, a hybrid navigation and obstacle-avoidance algorithm is proposed. The hybrid navigation method adapts to different application requirements. By integrating a formation-conservation mechanism, the hybrid method monitors the relative positions of drones in real time and dynamically adjusts their positions to maintain a desired formation. This development adds versatility to the algorithm, as it can prioritize either formation conservation or optimized path planning based on the application's specific needs.

2. LITERATURE REVIEW

With advancements in electronic intelligence and control sub-systems, UAVs have gained popularity and are widely used in various professional and recreational applications [21]. Although initially used primarily for military purposes, UAVs have expanded their presence in the commercial and industrial sectors [22]. This expansion can be attributed to technological advancements and improved power capacities, enabling customized structures, configurations, and equipment customized to specific tasks[23]-[24].

Engaging in risky or laborious tasks often requires the deployment of multiple UAVs. This requirement arises from the significant time commitment and limited autonomy of these small unmanned vehicles. Using multiple drones concurrently, each vehicle assuming the role of a backup in the event of failure, tasks can be performed in parallel, resulting in reduced overall time requirements compared to sequential execution with individual drones. This collective approach improves efficiency, productivity, and the ability to tackle challenging endeavors effectively. This strategy draws inspiration from the remarkable group dynamics observed in various natural biological models, such as birds or ants [25]. These organisms exhibit remarkable coordination and interaction among individuals, as they work together toward a shared objective: migrating to warmer regions or efficiently transporting food to their colonies. Swarm-based systems aim to harness the power of coordinated action and adaptability to solve complex problems.

Metaheuristic algorithms have emerged as powerful tools in artificial intelligence and mathematical optimization, gaining significant attention over the past two decades [26]. These algorithms exhibit stochastic behavior and offer optimal solutions with reduced computational effort compared to conventional techniques. Metaheuristic algorithms are problem-independent and can be broadly classified into four categories: swarm-based (SI), physics-based, evolutionary-based and human-based algorithms. SI algorithms, particularly, harness the collective intelligence observed in natural systems, such as birds, ants, fish, wolves, and other social organisms. These algorithms strike a balance between exploration and exploitation within the search space. Exploration involves a global search for exploration, while exploitation involves a local search in areas identified as promising during the exploration phase. SI algorithms aim to find optimal solutions to a wide range of problems by emulating these social behaviors.

Multi-UAV cooperative path planning aims to meticulously determine an optimal flight path for each UAV, starting from its initial point and ending at the terminal point. This planning process involves minimizing overall flight costs while simultaneously satisfying various constraints, including the distance between UAVs, arrival time, safety requirements, and UAV Performance Criteria. Chen et al. tackled the air-ground cooperation problem of Unmanned Ground Vehicles (UGVs) and UAVs by combining the Genetic Algorithm (GA) with ACO [27]. Their method effectively decoupled the routes of UGVs and UAVs, optimizing the heterogeneous delivery problem and obtaining optimal routes.

Kyriakakis et al. introduced a novel dynamic optimization problem for UAV search and rescue scenarios [28]. They developed a multi-swarm framework with additional UAV constraints and evaluated seven optimization algorithms. Yu et al. proposed a mutation-constrained adaptive selection Differential Evolution Algorithm (DE) to handle the optimization problem [29]. The algorithm aimed to find the optimal solution while satisfying these constraints. To plan feasible paths that cover an entire area for a UAV to maintain a constant flight level relative to the ground, Gonzalez et al. developed a coverage algorithm [30]. They used DE to evaluate the resulting paths and select the best path based on distance costs.

Wu et al. developed an improved fast convergence Artificial Bee Colony (ABC) algorithm to obtain the optimal path in a battlefield environment, considering conflicts and constraints [31]. Xu et al. developed an improved multi-objective Particle Swarm Optimization (PSO) algorithm [32]. Their approach calculated feasible and collision-free trajectories with variable minimum altitude, length, and angle rates.

Phung and Ha addressed the path-planning problem for multiple UAVs in complex environments with multiple conflicts [33]. They proposed the Spherical Vector-based PSO, which efficiently explores the configuration space of UAVs to generate the optimal path that minimizes the cost function. Tong et al. integrated the Pigeon-inspired Optimization (PIO) algorithm with DE mutation strategies for path-planning optimization [34]. Their approach considered three indices: path length, path sinuosity, and path risk. Qu et al. combined hybrid Grey Wolf Optimization (GWO) with a modified Symbiotic Organism Search (SOS) algorithm [35]. They simplified the GWO phase to improve the convergence rate and maintain the population's exploration ability. The SOS phase was synthesized with GWO to enhance the ability to exploit.

There have been significant recent advancements in UAV swarm research in the integration of AI algorithms to enhance decision-making and adaptability [36]-[37]. However, challenges remain in achieving robust solutions for complex tasks, especially in dynamic and uncertain environments. Key research gaps include the need for improved collision avoidance, navigation strategies, and path-planning algorithms that can effectively address real-world constraints, such as uncertainty, security restrictions, and dynamic obstacles, which until now were discussed as an open issue and a research challenge [38]. While existing studies have explored these areas individually, there is a need for integrated systems that can comprehensively address these challenges. The proposed system significantly contributes to UAV swarm research by integrating several essential components, including a collision-avoidance algorithm, a hybrid navigation approach, and a path-planning algorithm based on Ant Colony Optimization. The system showcases cooperative detection and avoidance capabilities, enabling UAV entities to collaborate effectively in detecting and avoiding collisions with both obstacles and other UAVs. It functions in a 3D dynamic environment, addressing uncertainties, security restrictions, and multiple objects. Utilizing ACO, the path-planning algorithm exhibits distributed-planning behavior, as it is applied to each target in the mission, ensuring optimized safety and cost objectives. The system's ability to maintain formations enables UAV swarms to preserve their desired shapes and spatial dimensions. These features set the system apart from other studies in the literature, demonstrating its versatility and potential for real-world applications in various cooperative missions.

3. PROPOSED SYSTEM

The proposed system consists of four key modules: the ACO-based path-planning module, the hybrid-path navigation module, the collision-avoidance module, and the messaging module. Each module serves a specific purpose in cooperative mission planning. These components work together to optimize the mission performance of the UAV swarm. Figure 1 illustrates the key components of the proposed system.

The ACO module forms the core of the system, drawing inspiration from ants' foraging behavior. Using pheromone-based communication and local heuristics, it guides the decision-making process of individual UAVs. By balancing exploration and exploitation, the ACO module facilitates the search for optimal paths within the swarm. To enhance the adaptability and flexibility of the system, a developed approach called the Hybrid Approach is proposed. The Hybrid Approach introduces adaptability to the system by dynamically adjusting the path-planning strategy based on the desired swarm shape. The Obstacle Avoidance module integrates real-time obstacle detection and intelligent decision-making to ensure safe navigation. By employing collision-avoidance algorithms, the module guides UAVs to navigate around obstacles and complete their missions. The Messaging System facilitates effective communication and information sharing among UAVs.

3.1 ACO-Module

Ant Colony Optimization (ACO) was initially proposed by Dorigo et al. as a powerful multi-dimensional optimization algorithm that draws inspiration from the foraging behavior of specific species of ant [39]-[40].

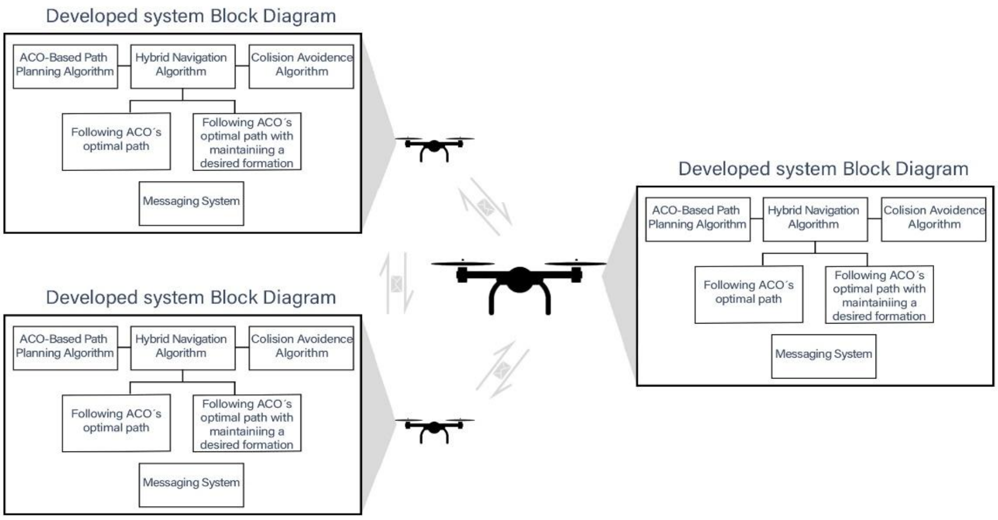

Figure 1. System block diagram.

Through collective intelligence, the ACO collaboratively determines the shortest path based on the density of the pheromone trail [41]. The strength of ACO lies in its ability to balance exploration and exploitation effectively. Randomly exploring ants ensures a diverse search-space coverage, enabling the algorithm to discover potential solutions. At the same time, the exploitation of the pheromone trails by other ants reinforces the convergence towards promising paths, promoting the identification of optimal solutions. This inherent balance between exploration and exploitation makes ACO highly robust and adaptive in dynamic problem domains.

To simulate the behavior of real ants, ACO models employ equations or algorithms to update and propagate the pheromone values dynamically. These updates reflect the collective behavior of the artificial ants and play a critical role in the convergence of the algorithm toward optimal or near-optimal solutions. The equation for the pheromone update is as follows:

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 11, No. 03, September 2025.

$$\tau_{ij}(t+1) \leftarrow (1-\rho) * \tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{1}$$

where:
- $\tau_{ij}(t+1)$: Represents the updated pheromone value on the path of component i to j at time t+1.
- $\tau_{ij}(t)$: Represents the current pheromone value on the path of component i to j at time t.
- ρ: The pheromone evaporation rate is a control of the rate at which pheromones decay.
- $\Delta\tau_{ij}(t)$: The pheromone deposit rate represents the amount of pheromone deposited on the path from component i to j at time t by the artificial ants constructing solutions.

ACO algorithms use mathematical models for simulating ant decision-making. Various models exist, often relying on state-transition rules and probabilistic methods. One widely used model is the ant system, which employs probabilities to choose paths. It balances pheromone intensity and heuristics, achieving the exploration-exploitation trade-off. The probability equation used in ant decision-making is as follows:

$$P_{ij} = \frac{(\tau_{ij}(t))^{\alpha} * (\eta_{ij}(t))^{\beta}}{\sum_{0}^{Allpossiblepaths} (\tau_{ij}(t))^{\alpha} * (\eta_{ij}(t))^{\beta}} \tag{2}$$

where:
- $P_{ij}(t)$: Represents the probability of selecting the path from component i to j at time t.
- $\eta_{ij}(t)$: Represents a problem-specific heuristic value associated with the path of component i to j at time t.
- α and β: Are parameters that control the relative importance of the pheromone trail and heuristic information, respectively.
- The denominator $[\sum_{0}^{Allpossiblepaths} (\tau_{ij}(t))^{\alpha} * (\eta_{ij}(t))^{\beta}]$ represents the sum of the probabilities for all possible paths or components at time t.

In decision-making, artificial ants consider pheromone information and problem-specific heuristics. Pheromone information, encoded in the pheromone trails, provides a collective memory of the paths previously explored by the ants. The higher the pheromone concentration on a path, the more attractive it becomes to subsequent ants.

The equation used to calculate the heuristic value, $\eta_{ij}(t)$, is problem-specific and depends on the characteristics of the path or component. One example of a commonly used heuristic is the inverse of the distance between two points, represented as:

$$\eta_{ij}(t) = \frac{1}{D_{ij}} \tag{3}$$

Where, D_{ij}: Represent the distance between the two points *i* and *j*.

In this research, the characteristics of swarm UAV path planning and the parameter values accordingly are considered carefully, as shown in Table 1.

Table 1. Parameter values for ACO in the proposed algorithm.

ACO Parameter	**Value**
Evaporation Rate	0.5
Pheromone Deposit Rate	1/Path length
Heuristic Information (β)	5
Importance of Pheromone Trails (α)	1
Initial Pheromone Rate	0.01
Number of Iterations	50

At initialization, each drone establishes its colony by populating several ants. These ants are then tasked with finding the optimal path from the drone’s start to its target point. The information sharing and cooperation among ants occurs exclusively within the bounds of the same colony, which belongs to a specific UAV. Each ant performs its path exploration within a colony, utilizing local and global search strategies to identify the most efficient route toward the target. The local search involves making decisions based on the immediate surroundings and information available locally within the drone’s colony. Meanwhile, global search entails updating pheromone trails to incorporate valuable information gathered during exploration. As a result, the swarm of drones operates with high degrees of decentralization and parallelism, significantly enhancing the overall efficiency and scalability of the system. The algorithm’s key steps are shown in Algorithm 1.

Algorithm 1 ACO-based Path Planning Algorithm

```
Initialize algorithm parameters
Set starting and target positions for the ant's paths
Create a list of random points representing the map, including start and target nodes
Connect nodes with edges and set initial pheromone values
Initialize the pheromone matrix
Number of iterations ⇐ 0
while Number of iterations < desired number of iterations do
    Populate ants on the map
    for each ant in the ant-list do
        Create a visit list and add the start point to it
        while ant is not at the target node do
            Move the ant to the next node based on Eq2
            Add the chosen point to the visit list
            Apply local search
            Apply global search
            Update the pheromone matrix based on Eq1
        end while
    end for
    Number of iterations ⇐ Number of iterations + 1
    if Number of iterations desired number of iterations then
        Calculate the distance of each ant's shortest path
        Compare distances of shortest paths and output the optimal path
    end if
end while
```

3.2 Collision Avoidance Algorithm

The collision-avoidance process within the UAV swarm navigation system is an accurately designed multi-step procedure, supporting optimal path planning and obstacle avoidance. The collision-avoidance algorithm implemented in the proposed system builds upon a well-established approach presented in [42]-[43], known for its effectiveness in handling complex scenarios. To work for a swarm of UAVs instead of a single UAV, the modified obstacle avoidance algorithm is illustrated in Algorithm 2.

Algorithm 2 **Collision Avoidance**

```
Initialize each UAV with start point, target point, speed, rotation, scale and priority
The UAV moves to its current target
while UAV is moving to the target do
    Check if there is a potential collision on the UAV path
    if No potential collision then
        The UAV keeps moving to its target normally
    else
        Send a message to alert all drones in the swarm about the collision possibility
        Check if the UAV is considered to have the highest priority
        if UAV has the highest priority then
            Go back to The UAV moves to its current target and repeat
        else
            Generate a number of random points around the current position
            Calculate the distance to the target through the waypoints
            Find the nearest point with the minimum distance
            Check if the chosen point eliminates the potential collision
            if No, if the newly chosen point still leads to a potential collision then
                Go back to Generate a number of random points and repeat
            else
                Store the original target in the temporary target variable
                Set the target to the nearest point
                Go back to "UAV moves to its current target" to move the UAV to the nearest point
                Check if the nearest point is reached
                if The nearest point is not reached then
                    Go back to "UAV moves to its current target"
                else
                    Restore the original current target
                    Go back to "UAV moves to its current target"
                end if
            end if
        end if
    end if
end while
```

3.3 Messaging Module

The messaging module in the system facilitates effective communication and coordination between drones within the UAV swarm. The messaging module implemented in the proposed system is based on a well- established approach presented in [43]. It is crucial to enable the swarm to operate as a cohesive unit, dynamically adapting to changing conditions and avoiding collisions while pursuing its mission objectives. Significant updates are made to enhance dynamic adaptability and swarm robustness. The system now adopts a distributed-path planning and hybrid navigation approach, allowing for more efficient and resilient performance.

The messaging module enables drones within the swarm to share their real-time positions. This continuous data exchange is essential for maintaining the desired formation during cooperative missions. Knowing the positions of the other drones, each UAV can adjust its trajectory to stay in the designated formation.

3.4 System Design

The system described in this sub-section is designed to control a swarm of drones operating within a specified environment. Its primary objective is to optimize the movement and coordination of the drones to achieve efficient and effective task completion. The system aims to minimize the distance traveled, maximize productivity and ensure safe operation by intelligently guiding the drones through commands and paths. The drone-swarm navigation system can be adapted to two different options based on application requirements. In the first option, formation conservation is not an application requirement, while in the second option, the application requires maintaining a specific formation or shape. In both options, the drones follow the optimal path generated by the ACO module, ensuring efficient navigation and collision avoidance within the environment.

- **Option one:** The system coordinates the movement of the drones, optimizes their paths using ACO, controls their movement using PID controllers and performs collision avoidance to ensure safe operation within the swarm, as shown in Algorithm 3.
- **Option two:** In the second option shown in Algorithm 4, additional functionality is introduced when the application requires maintaining a specific formation or shape.

3.5 Cost Function Evaluation

For an objective evaluation of the overall performance of the swarm, the following data is collected before the evaluation parameters are computed:

- Minimum Distance: The straight-line distance between each drone's initial and final target positions.
- Total Travelled Distance: The cumulative distance traveled by each drone from its initial position to its final target.
- Total Travelled Time: The duration a drone needs to reach its final target.
- Number of Divergences: A divergence occurs when a drone deviates from its intended path.
- Number of Collisions: When two drones come into physical contact.

In addition, for option two, where formation conservation is required, an extra parameter is calculated:

- Average Distance Change: Measures how much each drone deviates from the desired formation.

The following evaluation parameters are formulated to capture the mission's quality, efficiency, completion and formation conservation during cooperative missions:

- Path Quality (PQ): Evaluates the efficiency of the path-planning module. It is calculated using the following equation (Eq. 4):

$$PQ = \frac{1}{N} * \sum_{i=0}^{N-1} \frac{MinTravelledDistance_i}{TotalTravelledDistance_i} * 100\% \tag{4}$$

where:

- N: the total number of drones in the swarm.
- $MinTravelledDistance_i$: is the minimum distance traveled by drone i from its initial position to its target.
- $TotalTravelledDistance_i$: is the total distance traveled by drone i during its mission.

A higher value for this parameter indicates that the drone successfully optimizes its path, following the shortest route to its target.

Algorithm 3 System Behavior – Option 1

```
1: UAVs receive important mission information from the ground station, including start and target points, speed, rotation, scale, formation and priority.
2: Apply the ant colony algorithm for each UAV.
3: while the optimal path is not generated do
4:      Keep waiting
5: end while
```

```
Begin the main loop of the system
for each UAV do
    Each UAV's initial target is set to the first node on its optimal path
    Each UAV sets the last point in the optimal path as the destination target
    Use a PID controller to calculate the drive forces for each axis (x, y, z)
    UAV moves to its current target
    while UAV is moving to the target do
        UAV checks if there is a potential collision on its path
        if No potential collision then
            Check if the current target has been reached
            if the current target is reached then
                Check if the current target is the destination target
                if the current target is the destination target then
                    Mission ends
                else
                    Update the target position to be the next node in the optimal path
                    Go back to UAV moves to its current target
                end if
            end if
        else
            Send a message to alert all drones in the swarm about the collision possibility
            Check if the UAV has the highest priority
            if UAV has the highest priority then
                Go back to UAV moves to its current target
            else
                Generate several random points around the current position
                Calculate the distance to the target through the waypoints
                Find the nearest point with the minimum distance
                Check if the chosen point eliminates the potential collision
                if Chosen point eliminates collision then
                    Store the original target in the temporary target variable
                    Set the target to the nearest point
                    Go back to UAV moves to its current target
                    Check if the nearest point is reached
                    if the nearest point is not reached then
                        Go back to UAV moves to its current target
                    else
                        Restore the original current target
                    end if
                else
                    Go back to the step of generating several random points and repeat
                end if
            end if
        end if
    end while
end for
Repeat the main loop until the UAV reaches its target
```

Algorithm 4 System Behavior – Option 2

```
UAVs receive mission information from the ground station, including start and target points, speed, rotation, scale, formation
  and priority.
Each UAV reads the start point for all other UAVs in the swarm.
Create a reference distance array that captures the distances between drones in the desired formation.
Apply the ant colony algorithm for all UAVs in the swarm.
Set the current target as the first node in the optimal path for the UAV and set the last point in the optimal path as the
  destination target.
Use a PID controller to calculate drive forces for each axis (x, y, z).
UAV moves to its current target.
while UAV is moving to the target do
    Read the current positions for all drones and create a current distance array, representing the current formation distances
  for the swarm.
    if The current distance array equals the reference distance array then
        The UAV keeps moving to its current position while checking for potential collisions and if the UAV reaches its
        target, continue to the next step.
    else
        Calculate the difference in distance between the UAV and all other UAVs in the swarm.
        if The difference in distances is less than the threshold then
            The UAV keeps moving to its current position while checking for potential collisions and if the UAV reaches its
            target, continue to the next step.
        else
            Generate several random points around the current position.
            Choose the nearest point.
            Check if the nearest point will maintain the UAV position.
            if The nearest point maintains the UAV position then
                Store the original target in the temporary target variable.
                Set the target to the nearest point.
                Go back to "UAV move to its current target".
                Check if the UAV reaches the nearest point.
                if UAV is not at the nearest point then
                    Go back to UAV move to its current target and repeat.
                else
                    Restore the original target.
                end if
            else
                Go back to Generate several random points and repeat.
            end if
        end if
    end if
end while
Repeat the main loop until the UAV reaches its target
```

- Mission Completeness (MC): Evaluates the collision-avoidance module's effectiveness and the UAV swarm's adaptability in successfully achieving its mission objectives. It is calculated using Eq. 5.

$$MC = \frac{N_{ReachedItsTarget}}{N} * 100\% \tag{5}$$

Where, $N_{ReachedItsTarget}$: is the count of drones successfully reaching their targets.

A higher value for this parameter indicates a success rate in achieving mission objectives, as many drones have reached their targets without collisions.

- Average of Divergence (AD): Measures how much each drone deviates from its original path to avoid collisions with other drones or with obstacles. It quantifies the quality of the new routes generated by the collision-avoidance module. Eq. 6 shows how this is calculated.

$$AD = \frac{\sum_{i=0}^{N-1} NumberOfDivergences_i}{N} \tag{6}$$

Where, *NumberOfDivergences*$_i$: is the number of times that drone i deviates from its original path.

- Swarm Flight Time (FT): Quantifies the efficiency of the UAV swarm in completing the mission, referring to a predefined time frame. It reflects how effectively all drones in the swarm work together to achieve mission objectives. This parameter is calculated as shown in Eq. 7.

$$FT = \frac{T}{TimeFrame} \tag{7}$$

Where, T: is the total time taken for all drones in the swarm to reach their respective targets.

A smaller value indicates a more cohesive and cooperative swarm, where drones work towards mission completion with minimal delays and divergences.

- Formation Change (FC): Evaluates how effectively drones in the swarm maintain their desired formation during cooperative missions. This parameter is calculated as shown in Eq. 8.

$$FC = \frac{1}{N} * \sum_{i=0}^{N-1} \frac{AverageofDistanceChange_i}{DefindThreshold} * 100\% \tag{8}$$

Where:

- *Averageof DistanceChange*$_i$: is the average distance change for each drone relative to the desired formation of the swarm.
- *DefinedThreshold*: is a predefined value that determines acceptable deviations from the desired formation.

A lower formation change value indicates better performance of the hybrid module, as it indicates that the drones successfully maintain their formation with minimal deviations from the desired configuration.

A cost function is formulated as a weighted sum (α,β,ω,γ,μ) of the four parameters in the first option and five parameters in the formation-conservation option, with each parameter assigned a specific weight to reflect its relative importance. The formula for the comprehensive cost function is given by Eq.9 and Eq.10 for option one and option two, respectively.

Option one:

$$CF = \alpha PQ + \beta(1 - FT) + \omega(1 - AD) + \gamma MC \tag{9}$$

Option two:

$$CF = \alpha PQ + \beta(1 - FT) + \omega(1 - AD) + \gamma MC + \mu(1 - FC) \tag{10}$$

These formulations ensure that the algorithm is evaluated based on its ability to optimize multiple key aspects simultaneously. A higher comprehensive cost function value indicates better performance.

3.6 System Complexity

The algorithm complexity measures how the performance and execution time of the algorithm scale with the increasing number of drones in the swarm. As the swarm size grows, the algorithm's efficiency becomes critical in ensuring real-time operation and mission success. Efficient algorithms with lower complexity ensure that the swarm can handle larger numbers of drones without compromising performance.

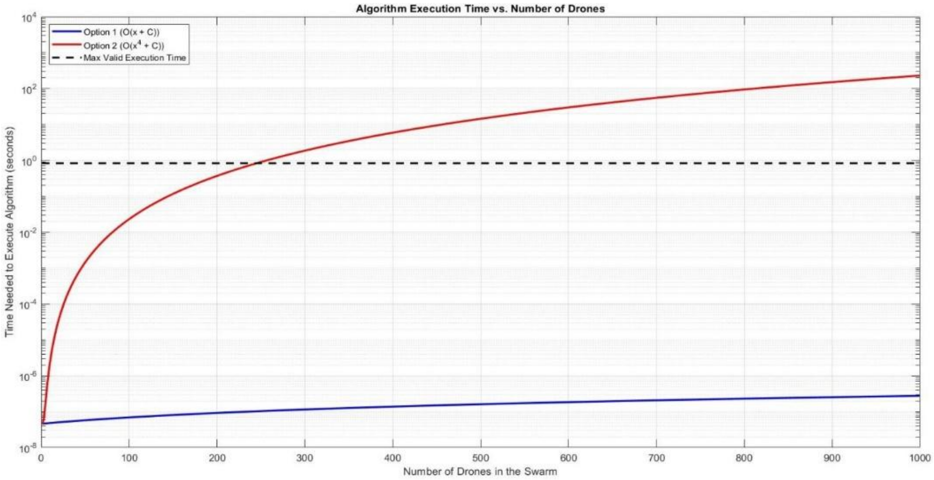

Figure 2. System execution time for both options as the number of drones in the swarm increases.

The execution time of the algorithm is a critical aspect that directly impacts the real-time operation of UAV swarms. In the first option, where formation conservation is not a specific application requirement, the algorithm complexity is O(X + C), where X is the number of drones in the swarm and C is the number of execution cycles, which remains constant regardless of the number of drones. The algorithm's scalability in this option is relatively better due to the linear complexity, making it suitable for swarms with a large number of drones.

In the second option, where formation conservation is essential, the algorithm complexity becomes $O(X^4 + C)$. This increase in complexity is due to the additional calculations and coordination required to maintain the desired formation during cooperative missions. The formation-conservation constraint introduces non- linearity in the algorithm, which impacts its scalability as the number of drones increases.

To ensure real-time execution in both options, an upper bound for the execution time of the algorithm is established as follows:

- For Option One:

$$(X + C) * (executioncycletime) < \left(\frac{sensingrange}{dronespeed}\right) \tag{11}$$

- For Option Two:

$$(X^4 + C) * (executioncycletime) < \left(\frac{sensingrange}{dronespeed}\right) \tag{12}$$

By meeting this condition, the algorithm can guarantee safe and efficient navigation for the entire swarm, even in dynamic and densely populated environments. Figure 2 illustrates the algorithm execution time for both options as the number of drones in the swarm increases.

4. SIMULATION AND RESULTS

The proposed system is implemented using the UTSim simulator, which offers an adaptable platform for creating and configuring multiple instances of UAVs [43]. The simulation setup involved the implementation of flight scenarios in a 3D environment, where the UAVs were controlled using the proposed system. Before each mission, the initial locations and destinations of the UAVs were defined based on the specific scenario. The UAVs used in the experiments were all fixed in size, with a half-meter diameter. Their speeds were maintained at a constant value of 6 m/s throughout the missions. Due to the inherent characteristics of rigid bodies, the speed decreased when the UAVs changed direction or reached their destinations.

Each run was performed 35 times in the 3D space to ensure reliable results. In an obstacle-free environment where formation maintenance is not a mission requirement, the swarm exhibits perfect consistency across all 35 experimental runs, with zero variability within a confidence interval of 95%.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 11, No. 03, September 2025.

However, when operating in a dense obstacle environment with a 0.1 allowable distance change, the system displays slight variability. For a 30-drone swarm, the error margin remains below 0.007 for distance change and 0.009 for flight time. As the swarm size increases to 40 drones, the error in total distance remains below 0.12. Even with a swarm of up to 80 drones, the error margin for distance change stays below 0.01. These consistently low error margins across all test conditions provide strong evidence of the system's robustness and reliability in both controlled and complex environments.

4.1 Algorithm Constraints and Assumptions

Constraints play a vital role in shaping the behavior and performance of UAV swarms during missions. They are essential elements that impose limits and restrictions on various aspects of the swarm's operation, ensuring safe, efficient, and coordinated behavior. Several constraints were considered to study the system's performance under different scenarios:

- Maneuverability Constraints: The maximum turning angle (θmax) was set to 30 degrees on the x-axis while remaining unrestricted in the y and z-axes.

$$\Theta_i(T) - \Theta max \leq 0 \tag{13}$$

  where: $\Theta_i(T)$: The turning angle of the i^{th} UAV at time T.

- Sensing Range Constraint: Each UAV sampled 25 points every time a reroute was computed. Rerouting was triggered when a passive obstacle was detected or when a higher-priority UAV was sensed. The sample points were taken within a customizable radius (Rs) of a circle/sphere set at 5 meters.

$$Rs - Dij(T) \geq 0 \tag{14}$$

  where, $Dij(T)$: The distance between the i^{th} UAV and the j^{th} UAV or obstacle at time T.

- Collision-avoidance Constraints: The algorithm incorporates a safe distance, denoted as Dmin, between two UAVs or between a UAV and an obstacle. This distance defines the collider sensing range, represented by the radius of a circle or sphere centered at the UAV.

$$Dmin - Dij(T) < 0 \tag{15}$$

  where, $Dij(T)$: The distance between the i^{th} UAV and the j^{th} UAV or obstacle at time T.

- Operating-range Constraint: The flight operation area was defined as 1 km * 1 km, providing a bounded environment for the swarm's missions.
- Time frame: the time frame is set to be a one-minute flight.
- For the first option's cost function, the (α,β,ω and γ) are 0.3, 0.3, 0.2, 0.2, respectively.
- For the second option's cost function, the (α,β,ω,γ and μ) are 0.2, 0.2, 0.2, 0.2, 0.2, respectively.

The experiment scenarios were designed to vary the number of drones within the flight area, ranging from 5 to 80 drones. The number of obstacles (moving and static) gradually increased, with the maximum number exceeding the total number of UAVs in the swarm, which is moving randomly in the environment. In the second option, various thresholds were tested to evaluate the performance of the hybrid navigation approach.

4.2 Effects of Different Safe Distances

This sub-section investigates the influence of varying safe distances on swarms of sizes ranging from 5 to 80 UAVs. The safe distance is incrementally increased from 1 meter to 3 meters for each case. This analysis provides insights into the optimal safe distance setting that maximizes the UAV swarm's efficiency and effectiveness in different scenarios. In this sub-section, all tests were conducted in obstacle-free environments and the safe-distance parameter of the system was adjusted and controlled from the ground station before each mission. The mission is designed, allowing tuning for a safe distance based on the distances between the drones and the total travel distance for each drone between the starting and target points, without considering the number of obstacles as a part of the mission design. This will be considered a design-preparation phase to set the safe distance to the next sections. These evaluations provided valuable insights into the system's performance and how the adjustable parameters influenced its behavior when encountering unexpected obstacles during missions.

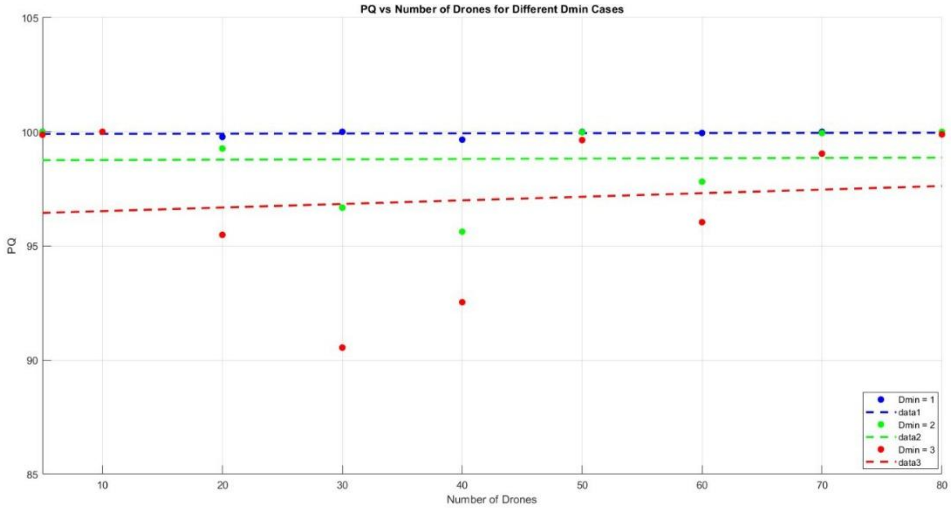

Figure 3. The path quality *vs.* the number of drones for different safe distances.

- Path Quality: The ACO module achieves a path quality of over 99% with 1m safe-distance scenarios. However, it reached 96% when the safe distance increased to 2m and 92% for the 3m safe distance, as shown in Figure 3. As the safe distance for drones increased, more drones had to make route diversions to avoid potential collisions, which increased the average of divergence, as shown in Figure 4, increasing the total traveled distance for each drone, which decreased the path quality. The formation and the distances between the start and target points for the drones are different from swarm to swarm, which explains the path quality and average of divergence behavior change for the same value of safe distance, since the distances between drones in the case with twenty UAVs are less than the distances between the drones in ten-UAVs. This increased the influence of large safe distances where the UAV needed to increase the number of divergences to save the safe distance simultaneously to avoid any potential collisions between the other UAVs in the swarm, which decreased the path quality. However, the path-quality values are close for all swarms because of the distributed approach in the ACO-based path-planning algorithm. The algorithm generates the optimal path for each drone based on its start and target point without considering the number of drones in the swarm.
- Swarm flight time: The increase in the average number of divergences leads to a greater total travel distance. This typically results in longer flight times for the swarms, as illustrated in Figure 5.

4.3 Effects of Number of Obstacles

The number of obstacles gradually increases. The number, speed, direction and all information of obstacles are unknown for the drones in the swarm to evaluate the system's adaptability to uncertainties. The obstacles move randomly in different directions and elevations. All cases are tested at a safe distance of 1 m.

- Path Quality: As illustrated in Figure 6, an increase in the number of obstacles does not significantly affect path quality in swarms with a small number of drones. This is because the drones maintain safe distances from each other and have a wide space within their operating range to locate the nearest point for collision avoidance. However, as the number of drones in the swarm increases, the distances between them decrease and the available operating space narrows, as shown in Figure 7. Consequently, the drones must find the nearest point to avoid collisions with obstacles while also considering a safe distance from other drones in the swarm. This necessity often increases the average number of divergences. Additionally, since the obstacles move randomly within the flight environment, their effects may vary across different scenarios.
- Swarm FT: Increasing the number of obstacles affected the mission time and the values of the

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 11, No. 03, September 2025.

cost function. However, since the obstacles are moving randomly in the flight environment, the effect does not show the same behavior in all scenarios, as shown in Figure 8 and Figure 9.

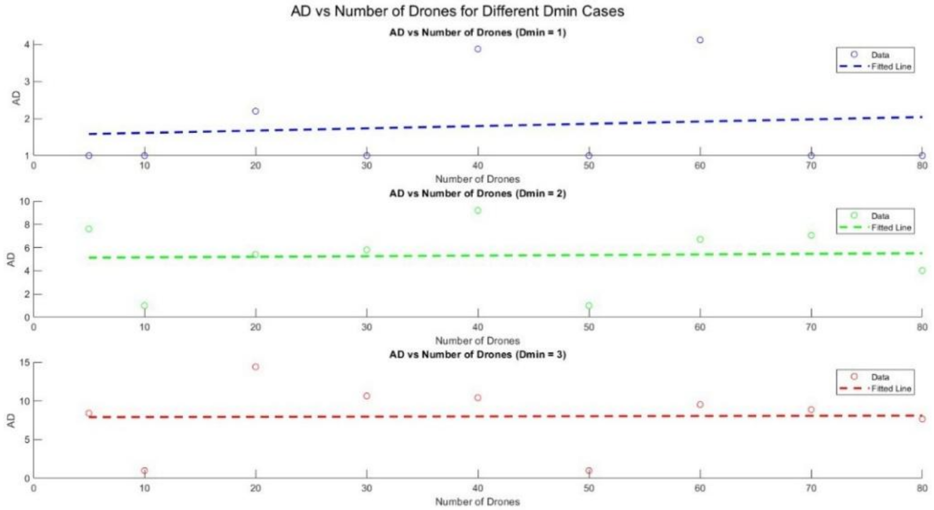

Figure 4. Average of divergence *vs.* the number of drones for different safe distances.

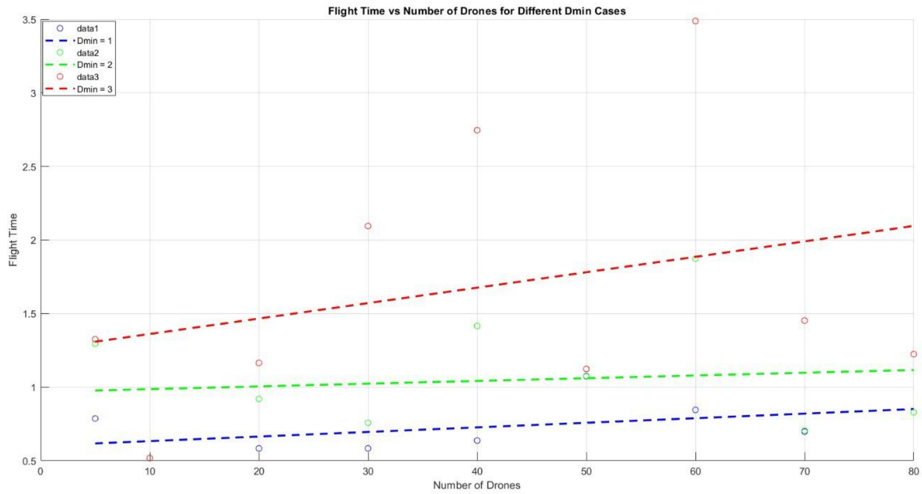

Figure 5. Swarm flight time divergence *vs.* the number of drones for different safe distances.

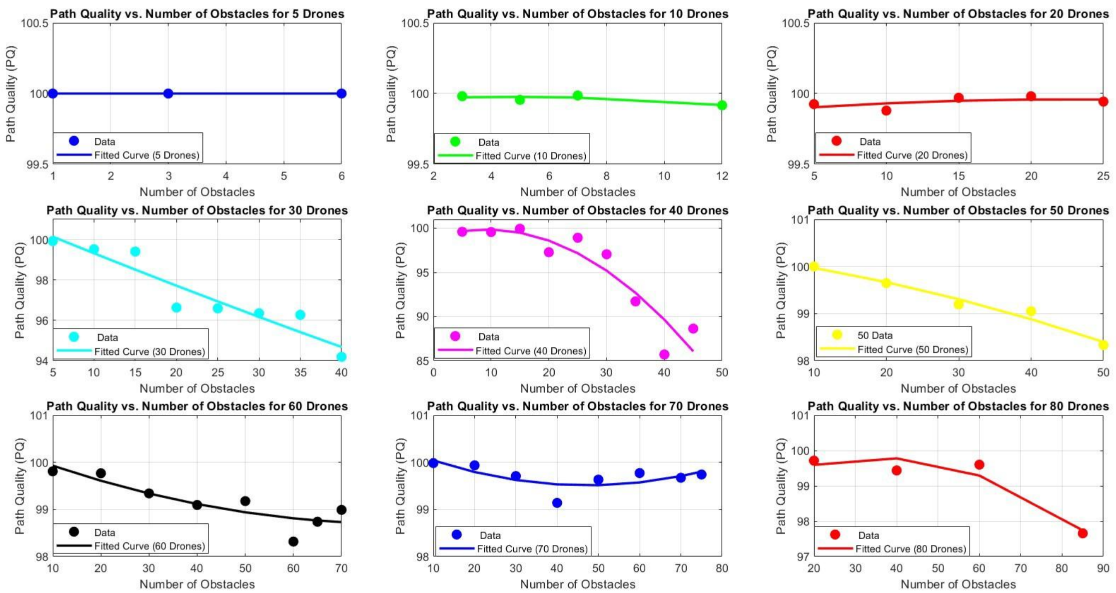

Figure 6. Path quality *vs.* the number of obstacles for different swarm sizes.

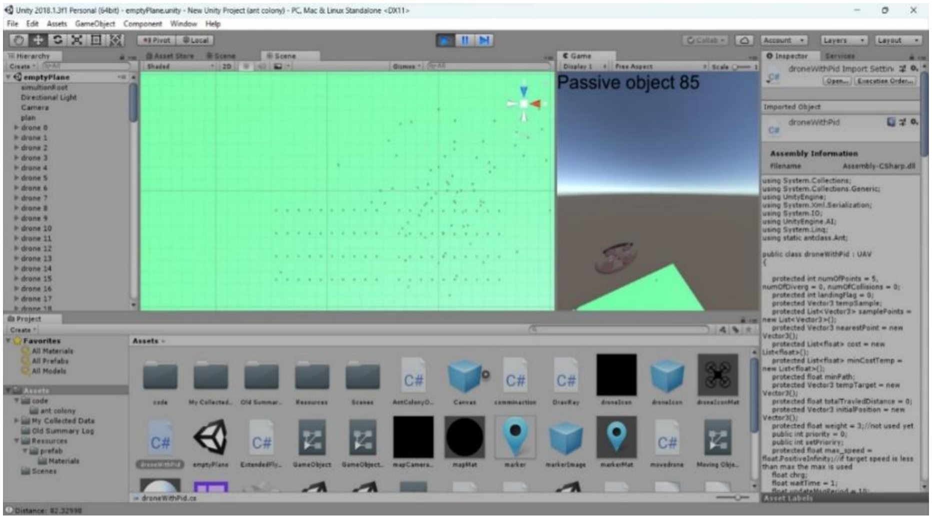

Figure 7. Eighty-UAV formation with 85 obstacles.

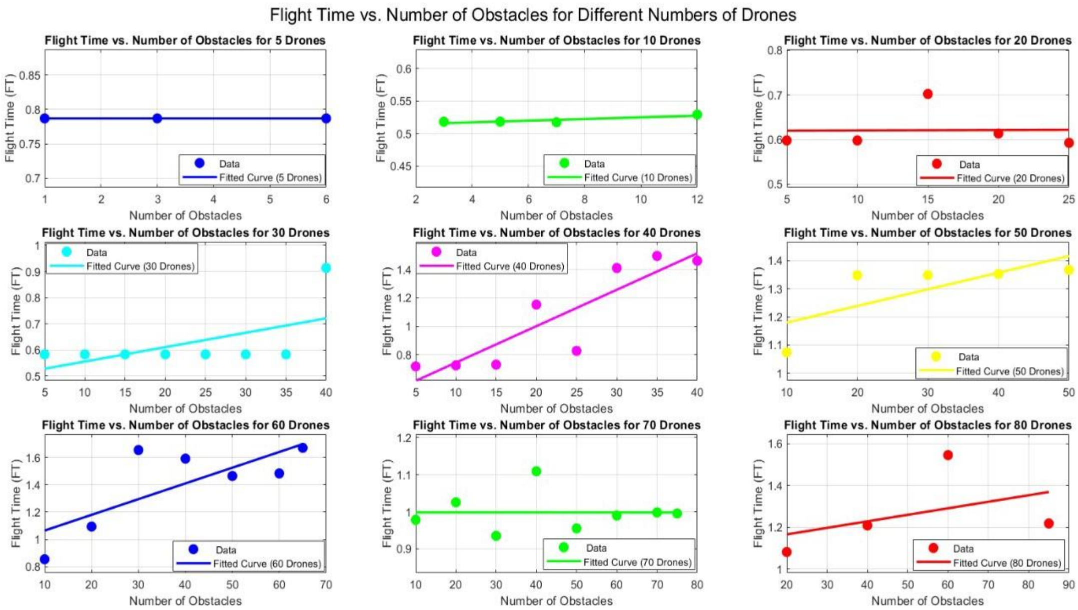

Figure 8. Swarm flight time *vs.* the number of obstacles for different swarm sizes.

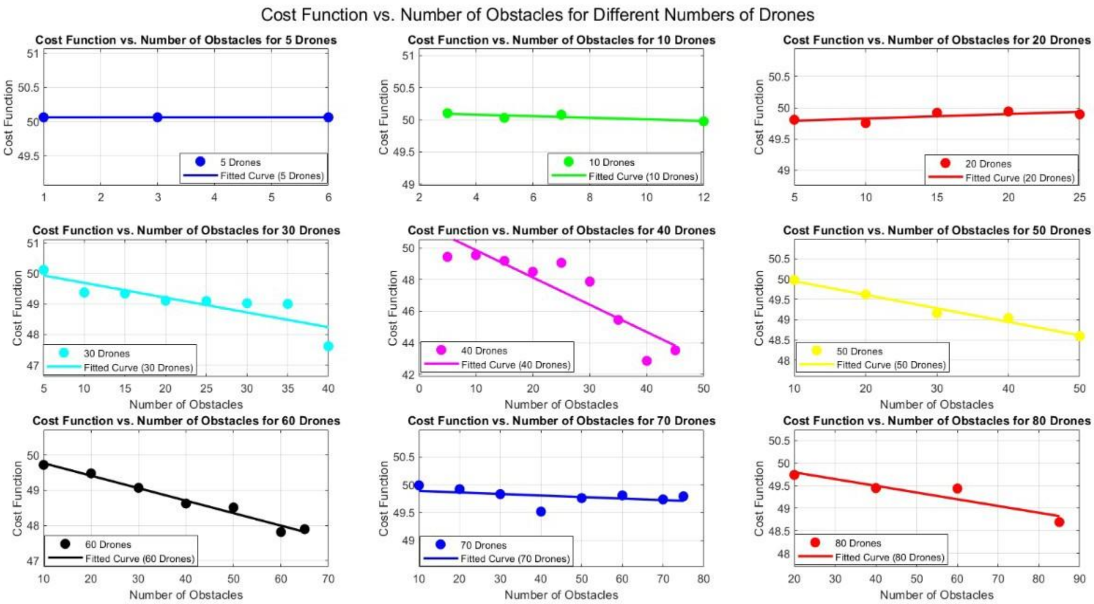

Figure 9. Cost function *vs.* the number of obstacles for different swarm sizes.

4.4 The Formation Threshold Effects

One of the important contributions of this study is the hybrid navigation approach, where application requirements are evaluated to prioritize following the optimal ACO path or maintaining a desired formation. In this sub-section, different thresholds (allowable change distance) are tested to evaluate the system's performance, where each case has a different formation with different distances between the drones within the formation and different distances between the start and destination points for each drone. All cases will be tested at a safe distance of 1 m.

- Path Quality: In the second option, with different threshold values, the system can manage the trade-off between maintaining formation and following the optimal path, resulting in optimal flight trajectories and minimal divergence. The system showed its ability to choose the nearest points to preserve the formation. As Figure 10 illustrates, the quality of the path is above 97% in all cases.
- Swarm Flight Time: Increasing the threshold allowed the drones more movement flexibility, reducing the time needed to complete the mission, as shown in Figure 11.

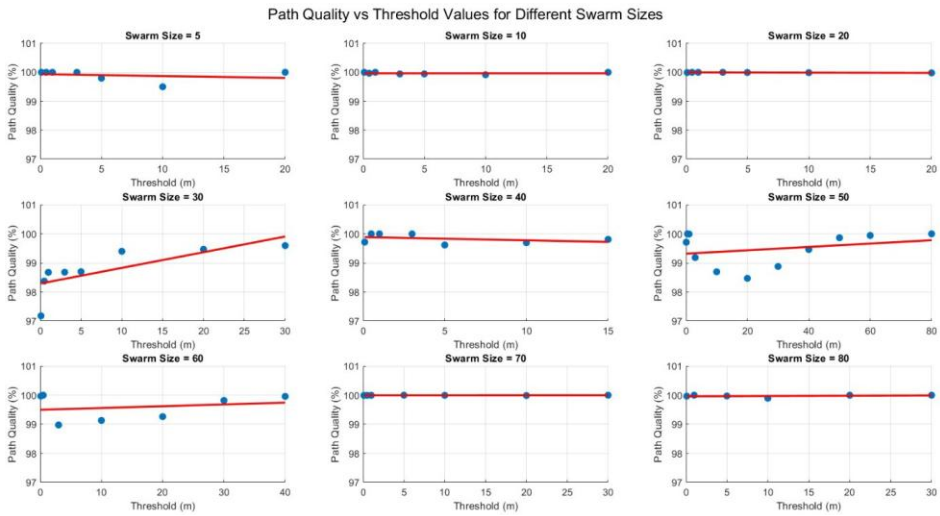

Figure 10. Path quality *vs.* the threshold values for different swarm sizes.

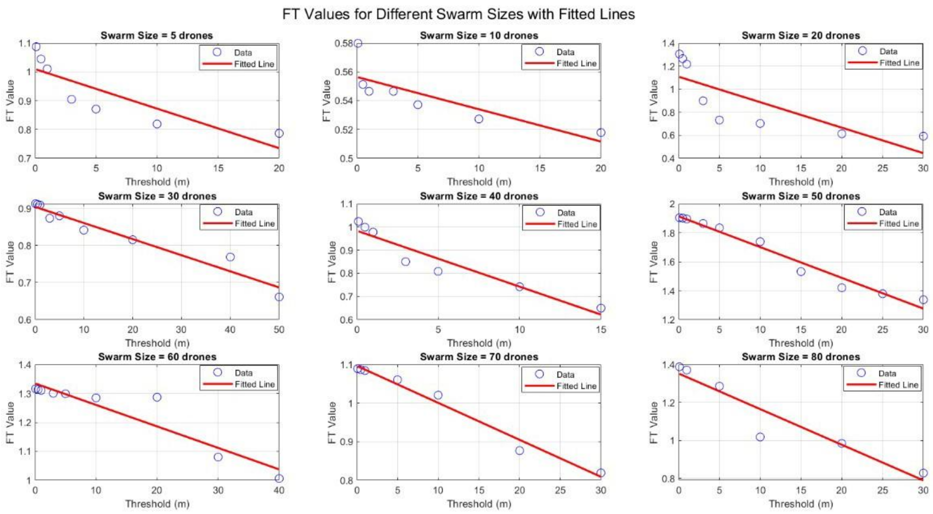

Figure 11. Swarm flight time *vs.* the threshold values for different swarm sizes.

- Formation Change: The formation change parameter evaluates the swarm's ability to maintain its desired formation during cooperative missions. The experiments demonstrated the success of the hybrid approach, as the formation change remained below 25% of the allowable change in all cases, as shown in Figure 12 and this percentage decreased when the threshold increased, but with different slopes, since each drone will generate a random point around its current position, which is directly related to its formation and optimal path and choose the nearest point that maintains its formation, saves the safe distance between the UAVs and avoids any potential collisions. However, in all cases, the system shows high adaptability with an acceptable formation change. It shows that the approach effectively conserves the formation during missions.

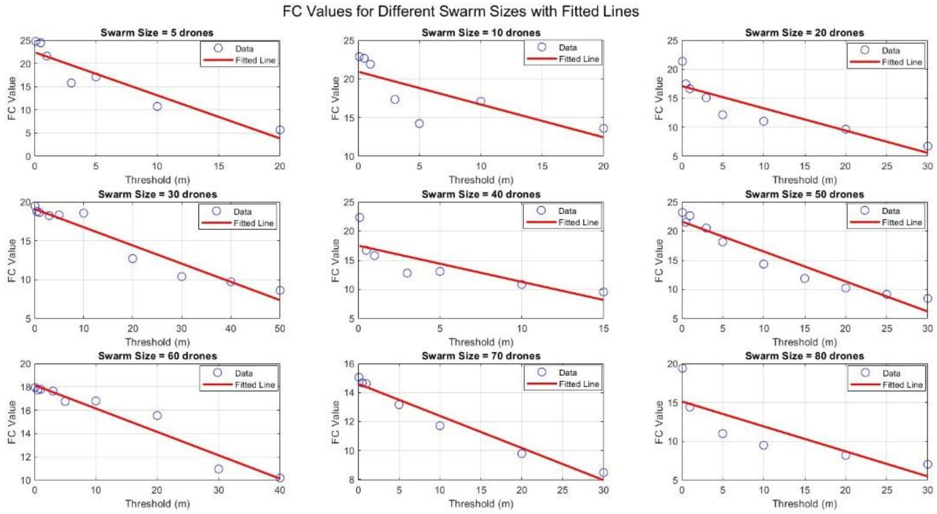

Figure 12. Formation change *vs.* the threshold values for different swarm sizes.

In summary, achieving robust solutions for complex tasks in dynamic and uncertain environments is a persistent challenge. In its integrated approach, the proposed system contributes to filling this gap by integrating ACO-based path planning, hybrid navigation and collision avoidance, enabling cooperative detection and avoidance in 3D dynamic environments with multiple objects, uncertainties and security restrictions. As demonstrated in the results, the system's performance is a direct consequence of this integration. ACO provides efficient path planning, while hybrid navigation and collision-avoidance algorithms work together to maintain formation and prevent collisions. The varying performance between swarms, where UAVs generate random points for collision avoidance, trading off mission objectives with safety directly related to these environments' dynamic and unpredictable nature is a key challenge identified in previous research. Although this variability is observed, the system consistently demonstrated high adaptability with acceptable formation changes, validating its robustness in complex scenarios.

4.5 Challenging Cases Evaluation

To further assess the system's performance, challenging cases were tested in which the swarm must preserve its formation with an allowable change distance of less than 0.1 m while flying in a dense-obstacle environment to assess how the system adapts to high levels of obstacle density while maintaining its formation. As shown in Table 2, cases with a threshold of 0.1 and many obstacles were tested when evaluating the hybrid navigation approach. This case's performance shows the hybrid approach's efficiency in achieving mission objectives while ensuring formation conservation.

Table 2. System's performance in challenging cases.

Number of Obstacles	Number of Drones	AD	FT	MC (100%)	PQ (100%)	FC (100%)	Cost Function
6	5	1	1.0969	100	97.7058	26.8118	34.3594
12	10	1	0.6097	100	99.7400	26.0080	35.0247
25	20	1.1	1.3182	100	99.1094	20.6193	35.8143
40	30	4.33	0.9135	100	94.1925	20.0458	34.3799
45	40	5.525	2.0233	100	94.5771	24.6229	33.0812
60	50	8.34	2.5419	100	97.6972	24.2275	33.1175
65	60	3.72	1.1032	100	98.9631	21.6300	35.1919
75	70	2.3571	1.5063	100	98.8808	31.3673	33.3299
85	80	4.19	1.8487	100	98.5522	28.7537	33.3524

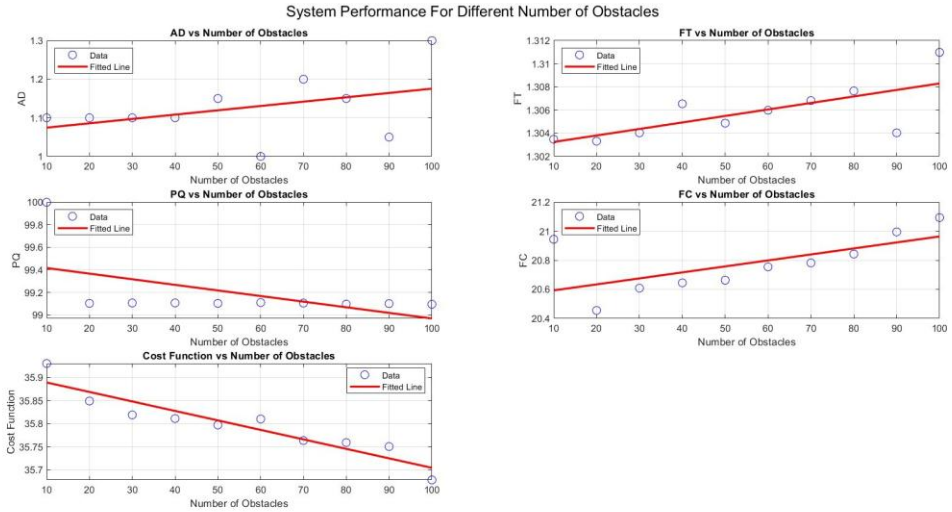

Figure 13. System performance for different numbers of obstacles.

To test the adaptability of the hybrid approach to the increased density of obstacles in the flight environment while maintaining the distance change to be less than 0.1 m, for a twenty-UAV swarm, the number of obstacles will start at 10 and then increase to 100, with a constant number of obstacles during each run session. As shown in Figure 13, increasing the number of obstacles will increase the possibility of collisions, since the operation space is so crowded, which increases the swarm flight time and the average number of divergences. With the increase in the number of obstacles, the UAVs need to increase the distance within the threshold to maintain their formation while saving their safe distance, which will normally affect path-quality and cost-function values. The tests are performed at a 1m safe distance.

5. CONCLUSION

This work presents an adaptable intelligent system for cooperative UAV swarm missions, integrating a path-planning algorithm based on the ACO algorithm, a collision-avoidance algorithm and a hybrid navigation system. The system was tested and evaluated in various scenarios, including different swarm sizes in dynamic 3D environments filled with moving and static obstacles while maintaining the desired formation. The simulation results demonstrate the system's outstanding performance, achieving a path quality of around 97% in most cases and never dropping below 90%, even in challenging scenarios. This reflects the high efficiency of the ACO module in finding optimal paths and the system's adaptability in consistently following them. The collision-avoidance module showed remarkable performance, ensuring that all missions remained collision-free, with a mission completeness rate of 100% in all testing scenarios. When the desired formation was necessary, the system showed its ability to maintain it even in dynamic environments within 30% of the allowable range in most cases. The system's success lies in its cooperative approach, in which all the modules work together smoothly. This collaborative and intelligent system illustrates its potential for real-world applications in various cooperative UAV-swarm missions.

REFERENCES

[1] S. Hayat et al., "Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint," IEEE Comm. Surveys and Tutorials, vol. 18, no. 4, pp. 2624–2661, 2016.

[2] M. Campion et al., "A Review and Future Directions of UAV Swarm Communication Architectures," Proc. of the 2018 IEEE Int. Conf. on Electro/Information Technology (EIT), pp. 0903–0908, 2018.

[3] R. Arnold, K. Carey, B. Abruzzo and C. Korpela, "What Is a Robot Swarm: A Definition for Swarming Robotics," Proc. of the 2019 IEEE 10th Annual Ubiquitous Computing, Electronics and Mobile Communication Conf. (UEMCON), pp. 0074–0081, New York, USA, 2019.

[4] M. Khelifi and I. Butun, "Swarm Unmanned Aerial Vehicles (SUAVs): A Comprehensive Analysis of Localization, Recent Aspects and Future Trends," J. of Sensors, vol. 2022, no. 1, p. 8600674, 2022.

[5] Q. Li et al., "A Review of Unmanned Aerial Vehicle Swarm Task Assignment," Proc. of the Int. Conf. on Guidance, Navigation and Control (ICGNC 2022), Springer, pp. 6469–6479, 2022.

[6] Y. Alqudsi, A. Kassem and G. El-Bayoumi, "A Robust Hybrid Control for Autonomous Flying Robots

in an Uncertain and Disturbed Environment," INCAS Bulletin, vol. 13, no. 2, pp. 187 – 204, 2021.
[7] S. A. H. Mohsan et al., "Unmanned Aerial Vehicles (UAVs): Practical Aspects, Applications, Open Challenges, Security Issues and Future Trends," Intelligent Service Robotics, vol. 16, no. 1, pp. 109–137, 2023.
[8] M. Abdelkader, S. Güler, H. Jaleel and J. S. Shamma, "Aerial Swarms: Recent Applications and Challenges," Current Robotics Reports, vol. 2, pp. 309–320, 2021.
[9] M. Cummings, "Operator Interaction with Centralized *versus* Decentralized UAV Architectures," Handbook of Unmanned Aerial Vehicles, pp. 977–992, DOI: 10.1007/978-90-481-9707-1_117, 2015.
[10] S. S. Ponda et al., "Cooperative Mission Planning for Multi-UAV Teams," Handbook of Unmanned Aerial Vehicles, vol. 2, pp. 1447–1490, DOI: 10.1007/978-90-481-9707-1_16, 2015.
[11] M.-H. Kim, H. Baik and S. Lee, "Response Threshold Model Based UAV Search Planning and Task Allocation," Journal of Intelligent and Robotic Systems, vol. 75, pp. 625–640, 2014.
[12] P. O. Pettersson and P. Doherty, "Probabilistic Roadmap Based Path Planning for an Autonomous Unmanned Helicopter," Journal of Intelligent and Fuzzy Systems, vol. 17, no. 4, pp. 395–405, 2006.
[13] L. De Filippis, G. Guglieri and F. Quagliotti, "Path Planning Strategies for UAVs in 3D Environments," Journal of Intelligent and Robotic Systems, vol. 65, pp. 247–264, 2012.
[14] O. Cetin, I. Zagli and G. Yilmaz, "Establishing Obstacle and Collision Free Communication Relay for UAVs with Artificial Potential Fields," J. of Intell. and Robotic Systems, vol. 69, pp. 361–372, 2013.
[15] S. Hacohen, S. Shoval and N. Shvalb, "Applying Probability Navigation Function in Dynamic Uncertain Environments," Robotics and Autonomous Systems, vol. 87, pp. 237–246, 2017.
[16] K. S. Camilus and V. Govindan, "A Review on Graph Based Segmentation," International Journal of Image, Graphics and Signal Processing, vol. 4, no. 5, p. 1, 2012.
[17] S. M. Persson and I. Sharf, "Sampling-based A* Algorithm for Robot Path-planning," The International Journal of Robotics Research, vol. 33, no. 13, pp. 1683–1708, 2014.
[18] H. Cartwright, "Swarm Intelligence by James Kennedy and Russell Ceberhart with Yuhui Shi. Morgan Kaufmann Publishers: San Francisco, 2001. £43.95.xxvii+512pp. ISBN: 1-55860-595-9," The Chemical Educator, vol. 7, pp. 123–124, 2002.
[19] A. Slowik and H. Kwasnicka, "Nature Inspired Methods and Their Industry Applications—Swarm Intelligence Algorithms," IEEE Trans. on Industrial Informatics, vol. 14, no. 3, pp. 1004–1015, 2017.
[20] R. D. Arnold and J. P. Wade, "A Definition of Systems Thinking: A Systems Approach," Procedia Computer Science, vol. 44, pp. 669–678, 2015.
[21] R. Austin, Unmanned Aircraft Systems: UAVs Design, Development and Deployment, John Wiley & Sons, vol. 54, ISBN: 978-0-470-05819-0, 2011.
[22] M. A. Akhloufi, S. Arola and A. Bonnet, "Drones Chasing Drones: Reinforcement Learning and Deep Search Area Proposal," Drones, vol. 3, no. 3, p. 58, 2019.
[23] R. J. Bachmann et al., "A biologically Inspired Micro-vehicle Capable of Aerial and Terrestrial Locomotion," Mechanism and Machine Theory, vol. 44, no. 3, pp. 513–526, 2009.
[24] M. Hassanalian et al., "A New Method for Design of Fixed Wing Micro Air Vehicle," Proc. of the Institution of Mechanical Engineers, Part G: J. of Aerospace Eng., vol. 229, no. 5, pp. 837–850, 2015.
[25] S. Roy, S. Biswas and S. S. Chaudhuri, "Nature-inspired Swarm Intelligence and Its Applications," Int. Journal of Modern Education and Computer Science, vol. 6, no. 12, p. 55, 2014.
[26] F. Glover, "Future Paths for Integer Programming and Links to Artificial Intelligence," Computers & Operations Research, vol. 13, no. 5, pp. 533–549, 1986.
[27] Y. Chen et al., "Delivery Path Planning of Heterogeneous Robot System under Road Network Constraints," Computers and Electrical Engineering, vol. 92, p. 107197, 2021.
[28] N. A. Kyriakakis et al., "Moving Peak Drone Search Problem: An Online Multi-Swarm Intelligence Approach for UAV Search Operations," Swarm and Evolutionary Comput., vol. 66, p. 100956, 2021.
[29] X. Yu, C. Li and J. Zhou, "A Constrained Differential Evolution Algorithm to Solve UAV Path Planning in Disaster Scenarios," Knowledge-based Systems, vol. 204, p. 106209, 2020.
[30] V. Gonzalez et al., "Coverage Mission for UAVs Using Differential Evolution and Fast Marching Square Methods," IEEE Aerospace and Electronic Systems Magazine, vol. 35, no. 2, pp. 18–29, 2020.
[31] C. Wu, X. Huang, Y. Luo and S. Leng, "An Improved Fast Convergent Artificial Bee Colony Algorithm for Unmanned Aerial Vehicle Path Planning in Battlefield Environment," Proc. of the 2020 IEEE 16th Int. Conf. on Control & Automation (ICCA), pp. 360–365, Singapore, 2020.
[32] X. Zhen et al., "Rotary Unmanned Aerial Vehicles Path Planning in Rough Terrain Based on Multi-objective Particle Swarm Optimization," J. of Sys. Eng. and Electr., vol. 31, no. 1, pp. 130–141, 2020.
[33] M. D. Phung and Q. P. Ha, "Safety-enhanced UAV Path Planning with Spherical Vector-based Particle Swarm Optimization," Applied Soft Computing, vol. 107, p. 107376, 2021.
[34] B. Tong et al., "A Path Planning Method for UAVs Based on Multi-objective Pigeon-inspired Optimisation and Differential Evolution," Int. J. of Bio-inspired Computation, vol. 17, no. 2, pp. 105–112, 2021.
[35] C. Qu et al., "A Novel Hybrid Grey Wolf Optimizer Algorithm for Unmanned Aerial Vehicle (UAV)

Path Planning," Knowledge-based Systems, vol. 194, p. 105530, 2020.

[36] Y. Alqudsi and M. Makaraci, "UAV Swarms: Research, Challenges and Future Directions," Journal of Engineering and Applied Science, vol. 72, no. 1, p. 12, 2025.

[37] Y. Alqudsi and M. Makaraci, "Exploring Advancements and Emerging Trends in Robotic Swarm Coordination and Control of Swarm Flying Robots: A Review," Proc. of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, vol. 239, no. 1, pp. 180–204, 2025.

[38] S. Alqefari and M. E. B. Menai, "Multi-UAV Task Assignment in Dynamic Environments: Current Trends and Future Directions," Drones, vol. 9, no. 1, p. 75, 2025.

[39] M. Dorigo, G. Di Caro and L. M. Gambardella, "Ant Algorithms for Discrete Optimization," Artificial Life, vol. 5, no. 2, pp. 137–172, 1999.

[40] D. Corne, M. Dorigo, F. Glover, D. Dasgupta, P. Moscato, R. Poli and K. V. Price, New Ideas in Optimization, ISBN: 0077095065, McGraw-Hill Ltd., UK, 1999.

[41] H.-B. Duan, "Ant Colony Algorithms: Theory and Applications," Chinese Science, 2005.

[42] B. H. Sababha et al., "Sampling-based Unmanned Aerial Vehicle Air Traffic Integration, Path Planning and Collision Avoidance," Int. Journal of Advanced Robotic Systems, vol. 19, no. 2, 2022.

[43] A. Al-Mousa, B. H. Sababha, N. Al-Madi, A. Barghouthi and R. Younisse, "Utsim: A Framework and Simulator for UAV Air Traffic Integration, Control and Communication," Int. Journal of Advanced Robotic Systems, vol. 16, no. 5, p. 1729881419870937, 2019.

ملخص البحث:

أصبحت الطّائرات غير المأهولة تلعب دوراً حاسماً في العديد من العمليات المتنوّعة، وبخاصّةٍ عندما يتعين الحفاظ على حياة البشر. ويُعدّ التّخطيط الفعال للمسار والتنسيق الذّاتي أمرين مهمين لأسراب الطّائرات غير المأهولة في المهمّات الدّيناميكية التّعاونية ثلاثية الأبعاد حيث يكون التّكيُّف في الزّمن الحقيقي أمراً أساسياّ. ويتناول هذا البحث التّحدّي المتمثّل في تحسين عمليات الطّائرات غير المأهولة الّتي تعمل في أسراب عن طريق اقتراح نظام مبتكر هجين للمِلاحة الجوّية يرتكز إلى التّحسين القائم على الطّريقة المتّبعة في مستعمرة النّمل. ويوازن النّظام المقترح بين إيجاد المسار الأمثل والتّحكم الدّيناميكي بتشكيل سِرب الطّائرات بناءً على متطلبات المهمّة التي يقوم بها السّرب. وتتمثّل إحدى المساهمات الأساسية لهذا البحث في نظام المِلاحة الجوّية الهجين المستخدَم الّذي يُعطي الأولوية للتّشكيل المرغوب للسّرب أو طول المسار وزمن الطّيران عبر آليةٍ ذات عتبة، الأمر الّذي يمكّن من التّكيُّف في الزّمن الحقيقي للبيئات المتغيرة. كذلك يوفّر النّظام المقترح دالّةً شاملةً للتّكلفة تعمل على تقييم جودة المسار واستهلاك الوقت واستكمال المهمّة والانحراف عن المسار.

لقد أثبتت التّجارب أنّ النّظام المقترح يُنتج مساراتٍ ذات جودة عالية تصل إلى 97% في معظم الحالات ولا تهبط إلى ما دون 90% حتى في السيناريوهات الّتي تتّسمٍ بالتّحدّيات. وتضمن وحدة تجنُّب الاصطدام استكمال المهمّات بنسبة 100%، سامحةً للطائرات دون طيار بالالتفاف حول العوائق مع الحفاظ على المسار الأمثل. ومن ناحية أخرى، تعمل آلية الحفاظ على التّشكيل بفعاليةٍ على المحافظة على التّشكيل المرغوب لسرب الطّائرات مع التّكيُّف مع العوائق، بحيث يبقى التّغيُّر في التّشكيل في حدود 30% من المدى المسموح به في معظم السيناريوهات، الأمر الّذي يدلّ على قدرة النّظام على الحفاظ على التّشكيل حتّى في البيئات الدّيناميكية. ويُعد هذا البحث إسهاماً في تقدُّم الذّكاء المتعلّق بأسراب الطّائرات غير المأهولة؛ فهو يمكّن من إنجاز العمليات –على نحو فعّال ومستقلّ- في البيئات المعقّدة ثلاثية الأبعاد للحصول على حُلولٍ تعاونية متنوّعة لإكمال المهمّات. وتفتح قابلية النّظام للاستجابة لمتطلبات تشكيلات أسراب الطّائرات بدون طيار الباب أمام إمكانياتٍ جديدةٍ للتطبيقات المرتبطة بأسراب الطّائرات غير المأهولة، مطوِّراً بذلك فعالية المِلاحة الجوّية ومحسِّناً التّحكُّم بالتّشكيلات.