SwarmSync: A Hybrid Bee-Ant Bio-Inspired Framework for Adaptive 3D Drone Mesh Networks

Priyanshu Singh
SCOPE
Vellore Institute of Technology, Chennai
India
priyanshu.singh2021@vitstudent.ac.in

R Archith Sai
SCOPE
Vellore Institute of Technology, Chennai
India
archithsai.r2021@vitstudent.ac.in

Smrithi Chandrashekhar
SCOPE
Vellore Institute of Technology, Chennai
India
smrithi.chandrashekar2021@vitstudent.ac.in

Athul Krishna Boban
SCOPE
Vellore Institute of Technology, Chennai
India
athulkrishna.boban2022@vitstudent.ac.in

***Abstract*— *Unmanned aerial vehicles (UAVs) or drones are being used more and more in mesh networks for surveillance, disaster relief, and environmental monitoring use cases. Coordinating a swarm of drones in dynamic 3D space with connectivity, energy optimization, and efficient coverage of targets is still a great challenge. This work presents SwarmSync, a new 3D drone mesh network simulation based on a hybrid bio-inspired strategy involving ant-inspired pheromone trails and bee-inspired waggle dance-based communication and scouting behaviors. The ant-inspired system supports decentralized pathfinding and self-repair network reconfigurability, while the bee-inspired tactics improve target coverage using recruitment signals and energy-optimal decision-making. The study compares SwarmSync's hybrid approach with an ant-inspired baseline model on the basis of target coverage efficiency, energy consumption, network connectivity, and target allocation time. Our results indicate that the hybrid approach provides faster target detection, improved energy usage, and more robust network connectivity, particularly in dynamic environments with obstacles and varying target priorities. This work paves the way to the creation of scalable, adaptive drone mesh networks for real-world applications, and it shows the potential of hybrid bio-inspired algorithms in swarm robotics.*

Keywords*— *Drone Mesh Networks, Swarm Intelligence, Bio-Inspired Algorithms, Self-Healing Networks, Pheromone Trails, Waggle Dance Communication, Target Coverage, Energy Efficiency, 3D Simulation.

I. Introduction

The quick growth of unmanned aerial vehicles (UAVs) has changed their function from isolated devices to networked swarms, which can support applications like surveillance, disaster response, environmental monitoring, and 3D mapping. These networks of drones have the potential to provide better coverage, robustness, and flexibility than single drones. Yet, coordinating a swarm in dynamic, three-dimensional (3D) spaces poses considerable challenges: ensuring network connectivity in the presence of failures, minimizing energy use for extended missions, and maintaining effective target coverage in the face of obstacles and changing priorities. Conventional swarm coordination methods—spanning from centralized control to bio-inspired strategies—tend to fail to address these closely coupled issues collectively, especially in unstructured, real-time environments where scalability and adaptability are critical.

Swarm intelligence, drawing from biological systems like ant colonies and bee colonies, has become a significant paradigm for distributed UAV coordination. Early works such as Reynolds (1987) presented flocking rules—cohesion, separation, and alignment—that form the basis of many swarm models [1], whereas Ant Colony Optimization (ACO) and Artificial Bee Colony (ABC) optimization have used pheromone trails and waggle dance communication, respectively, for path optimization and resource management [2], [3]. These bio-inspired strategies are superior in certain areas: ACO has strong, self-organizing navigation, such as that observed in Shao et al. (2020) [4], while ABC optimizes target search efficiency, as illustrated by Alfeo et al. (2018) [5]. Single, independent methods uncover weaknesses. Ant-inspired systems, although skilled at network self-healing, tend to suffer from slow target assignment, whereas bee-inspired strategies, proficient for recruitment, perform poorly with connectivity in 3D environments full of obstacles [4], [5]. Current hybrid approaches, e.g., Kuyucu et al. (2015), join stigmergy with stochastic movement but never couple ant and bee behaviors within a 3D aerial setting, leaving an open problem of scalable, adaptive swarm coordination [6].

Current study highlights such challenges and lacunas. Sauter et al. (2008) used digital pheromones for vehicle swarming for air-ground missions, attaining autonomy but without real-time adaptability to failures [7]. Alsolami et al. (2021) addressed synchronization in massive drone networks, but their cluster-based method ignores energy efficiency and target coverage [8]. Szczygiel et al. (2024) examined simulation-based swarm control, focusing on collision avoidance, but neglected dynamic target distribution [9]. At the same time, Liu et al. (2020) employed deep reinforcement learning for energy-efficient UAV navigation with enhanced coverage but at the cost of complex computation inappropriate for resource-limited drones [10]. Together, these works point to a requirement for a framework that optimizes connectivity, energy efficiency, and target coverage in dynamic 3D environments—a requirement that current single-paradigm or non-integrated solutions cannot address holistically.

This work presents SwarmSync, a new hybrid bio-inspired solution that combines ant-inspired pheromone trails and bee-inspired waggle dance-based communication and scouting behaviors to meet these demands. In contrast to existing solutions, SwarmSync unifies decentralized pathfinding and self-healing (ant-inspired) with efficient target recruitment and energy-optimal decision-making (bee-inspired), designed for 3D drone mesh networks. In a Python-implemented simulation, SwarmSync utilizes pheromone trails for directing network reorganization and recruitment signals for target detection speed-up, as we demonstrate in our optimized hybrid_drone.py and compare_simulations.py modules. We compare SwarmSync with an ant-inspired baseline along the major performance metrics—target coverage, energy efficiency, network connectivity, and allocation time—showing that it performs better in dynamic, obstacle-prone environments. By closing the gap between ant and bee paradigms' strengths, SwarmSync provides a scalable, adaptive solution for real-world swarms of UAVs, moving beyond the bounds of existing work. Not only does this work add a new model of coordination, but it also opens the way for practical deployments of swarm robotics in uncertain environments.

Fig 1. Swarm drones used in surveillance

II. LITERATURE SURVEY

The survey of literature across various studies on UAV swarm coordination, self-healing, and reliability presents a rich history of methods to improve autonomy, efficiency, and resilience in dynamic environments. Early seminal work in the 1980s presented swarm intelligence concepts, including flocking behaviors based on cohesion, separation, and alignment rules, which later evolved into navigational feedback models for multi-agent systems [1]. These concepts led to the development of bio-inspired algorithms such as Particle Swarm Optimization (1995), Ant Colony Optimization (1999), and Artificial Bee Colony optimization (2005) that have played a central role in UAV coordination and target search problems [2], [3], [4]. Future studies incorporated digital pheromones into possible fields for aerial combat missions and mixed stigmergic steering with random displacement, recording a super-additive effect on exploration efficiency when swarm size increases [5].

Target detection and surveillance by stigmergy and flocking have been a major thrust. The research involved physical pheromones in ground robots and digital pheromones and indirect communication for UAVs with significant search time reductions—e.g., 2,604 ± 248 ticks to 1,078 ± 106 ticks in the case of a 50-target scenario when applying stigmergy and flocking together [6]. Real-world experimentation with outdoor flocks of drones, such as the largest distributed aerial system, and sensor-constrained simulations also developed these techniques [7]. The shift to vector pheromones from scalars enhanced decision-making, whereas demonstrations with NASA at Wallops Island involved two Aerosonde UAVs (25 m/s, 230 m altitude, 5 kg payload) and four Pioneer UGVs (3 kph, 30 kg payload) illustrating autonomous adaptation and recognizing three out of four targets within 6-8 feet despite equipment malfunction [8].

Synchronization and dependability in swarm networks have also gained interest. Legacy approaches such as Network Time Protocol (NTP) and Precision Time Protocol (PTP) fail in high-latency or asymmetric SDN environments, with NTP having initial errors of 500 ppb and PTP facing scalability issues [9]. Reference Broadcast Synchronization (RBS) provides single-hop precision but has overhead in large networks of up to 200 drones [9]. Cluster-based approaches minimized synchronization delay by 75%, with clock offsets of 1.339 µs after 25 processes (σ=10 µs) [9]. Reliability analyses point to swarm complexity, with a three-drone system demonstrating 0.8625 availability and 0.1375 unavailability, and importance measures differing by failure probability (e.g., BI=0.375 vs. 0.350) [10].

Self-healing mechanisms deal with network failures essential for surveillance. A neural model for 10-25 UAVs, tested with 3-7 faults, employed dummy neurons to stabilize connectivity at a 50% threshold, lowering the healing function from 5.5869 (7 faults) to 2.0766 after recovery [11]. In cellular networks, random frequency hopping UAV relays obtained an 11.6 dB SINR improvement to overcome overload and outage gaps [12]. Energy-efficient navigation through distributed deep reinforcement learning enhanced efficiency by 19.1% compared to centralized approaches in a 6-UAV system [13]. Simulation improvements, using tools such as AirSim and RTK GPS (5 cm accuracy), simulated swarms under GPS latency (0-100 cm) and delay (0-500 ms), and optimized collision avoidance in a 20 m x 20 m grid [14].

Taken together, these papers highlight the trend toward decentralized, adaptive, and fault-tolerant UAV systems, integrating simulation and real-world issues, but with continuing gaps in scalability, real-time adaptability, and end-to-end failure recovery.

III. PROBLEM STATEMENT

The deployment of unmanned aerial vehicles (UAVs) in mesh networks has transformed uses like surveillance, disaster response, and nature watching by tapping the power of swarm intelligence in synchronized action in three-dimensional (3D) spaces. Yet, managing a drone swarm in dynamic, unstructured environments is tremendously challenging. Current coordination strategies—centralized, flocking-based, or bio-inspired—fail to tackle three fundamental needs at once effectively: sustaining high network connectivity in the face of drone failures or obstacles, energy optimization for extended operation, and effective target coverage under diverse priorities and environmental limitations. Ant-inspired methods, e.g., based on pheromone trails (Shao et al., 2020), are very good at decentralized pathfinding and self-organization but tend to be poor in mechanisms for quick target assignment and energy-conscious decision-making. On the other hand, bee-inspired approaches, such as waggle dance communication (Alfeo et al., 2018), improve target search effectiveness but fail to maintain network resilience in intricate 3D environments with dynamic obstacles.

Literature shows that most solutions use a single bio-inspired paradigm, which makes them less adaptable. For example, Sauter et al. (2008) showed pheromone-based swarming but neglected real-time adaptability [7], whereas Liu et al. (2020) emphasized energy efficiency through reinforcement learning at the expense of computationally complex overhead inappropriate for UAVs with resource constraints [10].

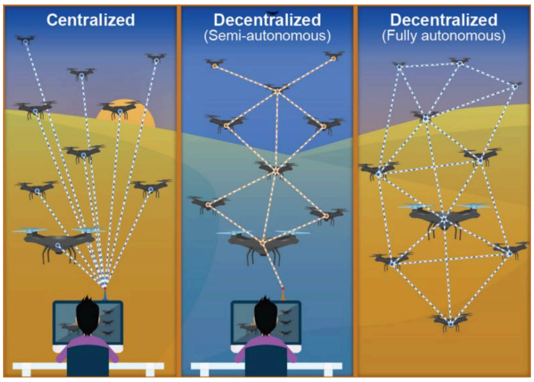

Fig. 2 Various modes of UAV communication

Hybrid efforts seldom combine ant and bee actions particularly for 3D aerial meshes, falling short in terms of scalability and robustness [6]. In addition, dynamic aspects—like drone malfunction, noisy sensing, and random target motion—compound these problems, leading to non-optimal performance in practice [10]. The lack of a single framework that integrates connectivity, energy efficiency, and fast target coverage in a scalable, decentralized fashion prevents the real-world deployment of adaptive drone mesh networks, calling for a new solution to address these multi-faceted challenges.

IV. RESEARCH OBJECTIVES

The main objective of this study is to develop and evaluate SwarmSync, a bio-inspired hybrid system bringing together ant-inspired pheromone trails [2] and bee-inspired waggle dance communication [3] to facilitate adaptive, efficient, and robust 3D drone mesh networks able to operate in dynamic environments. This research strives to overcome the key shortcomings of current swarm coordination methods by designing a system that simultaneously addresses the interrelated challenges of network connectivity, energy optimization, and efficient target coverage in real-time [4], [8]. The work aims to accomplish this by employing a multi-faceted methodology of combining decentralized, nature-inspired mechanisms into an integrated model, assessing its performance in realistic scenarios, and illustrating its innovative value for real-world UAV applications. Through the achievement of these goals, the research aims to close the gap between existing approaches and the requirements of real-time adaptive drone missions, providing a paradigm-shifting framework for swarm robotics.

One of the core goals is to create a hybrid coordination model that combines ant-inspired decentralized pathfinding and self-healing processes with bee-inspired recruitment signals and scouting activities. This entails creating a framework in which drones can navigate autonomously and uphold network integrity as they effectively locate and prioritize targets, and solve issues of connectivity, energy efficiency, and coverage within dynamic 3D environments [4], [5]. Another objective is to deploy and test this framework in a simulated 3D environment that is representative of actual scenarios, for example, ones with obstacles, drone failures, and changing target priorities [9]. This will be done by building a simulation platform to evaluate the performance of the framework under controlled but stressful conditions. The study also aims to quantify the effectiveness of the framework by comparing it to an ant-inspired baseline, with emphasis on major metrics such as target coverage efficiency, energy consumption, network connectivity, and target allocation time, to identify its strengths in uncharted environments. Lastly, the work aims to present the novelty and scalability of this hybrid solution, highlighting how the combination of the bio-inspired paradigms performs better than standalone single-paradigm solutions and is capable of scaling to enable large swarms. This will entail presenting its novelty and ability to facilitate effective UAV deployments, such as disaster response or ecological surveillance, via conceptual examination and simulated outcomes. With these contributions, the study hopes to lay a foundation for the continued development of swarm coordination, addressing the demands of adaptive real-time operation within complex aerial environments.

V. PROPOSED METHODOLOGY

To address the disadvantages of current drone mesh network coordination techniques in 3D dynamic environments—network connectivity, energy efficiency, and target coverage—this work suggests SwarmSync, a bio-inspired hybrid framework, which synergistically combines ant-inspired pheromone trails and bee-inspired waggle dance-based communication and scout behaviors. In contrast to standalone single-paradigm methods like Ant Colony Optimization (ACO) or Artificial Bee Colony (ABC), where each performs admirably in separate dimensions of swarm coordination but breaks down in integral adaptability, SwarmSync pairs the decentralized fault tolerance of ants with the speed and energy-based allocation of bees. This section discusses the theoretical underpinnings of the framework, detailed architecture, simulation implementation, and its addressing of the identified issues, resulting in a scalable and adaptive solution for practical UAV swarms.

V. (i) THEORETICAL FOUNDATION

The study draws inspiration from pre-existing studies and meticulous research work in the field. The theoretical foundation of the paper is listed below:

- Ant-Inspired Principles: Borrowing from ACO (Dorigo et al., 1999), SwarmSync employs pheromone trails as stigmergic signals for decentralized routing and network recovery [2]. This process emulates the self-organization of paths by ants and adjustment to disturbances, as found in studies such as Shao et al. (2020) [4], but expands on it with hybrid additions.

- Bee-Inspired Principles: SwarmSync is inspired by ABC (Karaboga, 2005), with waggle dance communication for target recruitment and exploration scouting, representing bees' effectiveness in resource allocation (Alfeo et al., 2018) [3], [5]. This is supported by the ant-inspired backbone with dynamic, priority-driven responsiveness.

- Hybrid Rationale: Merging the paradigms corrects for the limitations of single approaches—delayed target assignment in ACO and sparse connectivity in ABC—by utilizing their complementary merits, a new development in 3D aerial swarm scenarios with respect to existing hybrids [6].

V. (ii) SWARMSYNC FRAMEWORK DESIGN

The SwarmSync framework revolves around three key components, unified in a perfect harmony to enable swarm behavior optimality:

1. *Ant-Inspired Pheromone-Based Coordination*
   a. **Pheromone Trails**: Every drone deposits digital pheromone trails (PheromoneTrail class of drone.py) when it discovers a target, with intensity as a function of target priority (e.g., intensity = priority ×

1.0). Trails are retained locally with an evaporation duration of 60 seconds, deployed in lay_pheromone_trail, to allow the system to adjust to changing environments [2].

b. **Pathfinding**: Drones track the strongest trail (follow_pheromone_trails) in terms of adjusting velocity (max 10 m/s) toward trail positions, allowing decentralized convergence on targets or connectivity hotspots.

c. **Self-Healing:** The self_heal_network function relocates drones towards the average position of active neighbors when connectivity falls (e.g., due to drone failure), with network integrity preserved without centralized control.

2. *Bee-Inspired Recruitment and Scouting*

a. **Waggle Dance Signals**: When a target is detected, drones emit recruitment signals (RecruitmentSignal class in hybrid_drone.py) probabilistically (70% probability) through perform_waggle_dance. Signal intensity is proportional to priority (e.g., intensity = priority × 2.0), lasting for 30 seconds, and is limited to five signals per drone to avoid overwhelming.

b. **Recruitment Response**: Unmanned drones within 50 meters of a signal (follow_recruitment_signal) modify velocity (up to 5 m/s horizontally, 2 m/s vertically) towards the position of the loudest signal, to boost target coverage acceleration.

c. **Scouting Behavior:** 20% of the drones are assigned as scouts (is_scout in hybrid_drone.py), which search aggressively (scout_explore) with speeds up to 15 m/s, improving early target detection in unfamiliar environments [5].

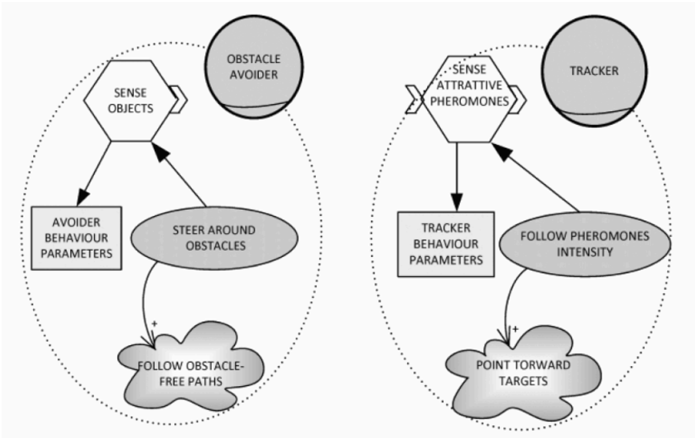

Fig 3. Behavioral approach of the drones in the network

3. *Energy Optimization and Integration*

a. **Energy-Conscious Decision-Making**: The optimize_energy function ranks targets by distance-to-priority ratio, sending unassigned drones to the most effective target, minimizing unnecessary travel and battery usage (drain rate = 0.1% per 0.1s).

b. **Hybrid Update Loop**: The update function in hybrid_drone.py coordinates these activities—sensing, following trails, responding to signals, avoidance of obstacles (avoid_obstacles), and motion—within a cycle so that a unified operation exists within connectivity, coverage, and energy objectives.

V. (iii) SIMULATION IMPLEMENTATION

SwarmSync is implemented and tested in a proprietary Python-based 3D simulation environment, as follows:

1. *Environment Setup*

a. **Spatial Configuration:** A 1000 × 1000 × 100 m^3 environment has 20 drones, 20 obstacles (radius 5-20 m), and 5 targets (xt, yt, zt) + (vx, vy, vz) × 0.1(priority 1-5), placed randomly at initialization.

b. **Dynamic Elements:** Targets are added (1% chance per frame) or removed (0.5% chance) to account for real-world uncertainty, while obstacles remain static but viewable in a 50-meter sensor range.

2. *Drone Behavior*

a. **Parameters**: Drones possess a 100-meter communication range, 50-meter sensor range, and starting 100% battery life, which is refreshed every 0.1 seconds over 1000 frames (100 seconds total).

b. **Parallel Processing:** hybrid_environment file has ThreadPoolExecutor that provides for concurrent updating of drones, mimicking real-time swarm dynamics efficiently.

3. *Visualization and Metrics*

a. **Visualization**: The hybrid_visualizer.py module visualizes a 3D plot with distinct markers for drones (active, scout, leader, inactive), obstacles, targets, pheromone trails, and recruitment signals, updated by FuncAnimation.

b. **Metrics Tracking:** Important metrics—target coverage (detected targets), network connectivity (communication links), swarm cohesion (average distance to centroid), and pheromone trail density (trail per target)—are tracked in target_coverage_history, network_connectivity_history, swarm_cohesion_history, and pheromone_trail_density_history, respectively, to evaluate SwarmSync's performance in dynamic 3D environments.

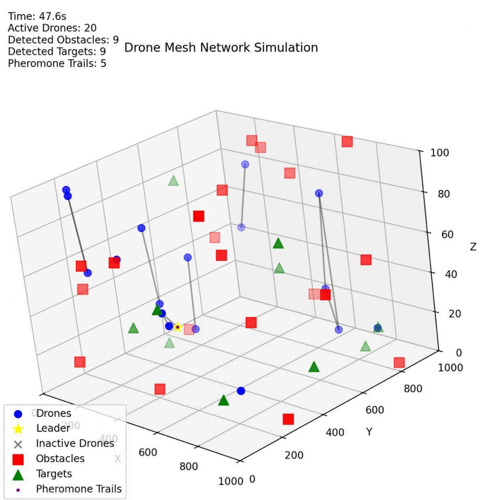

Fig 4. Drone Mesh Network Simulation

This image captures the Ant-Inspired simulation at 47.6 seconds with 20 drones in active status having spotted 9 targets and 5 pheromone trails. The higher rate of detected targets as compared to the 51.7-second screenshot (Figure 4) indicates further development, albeit with slower progress compared to the Hybrid model, where it spotted 8 targets at 12.2 seconds (Figure 5). The lack of recruitment signals restricts the Ant-Inspired model to quickly recruit drones to targets using only pheromone trails. The communication links demonstrate a network with connectivity, yet the model's target coverage performance is behind that of the Hybrid approach, as validated by previous results.

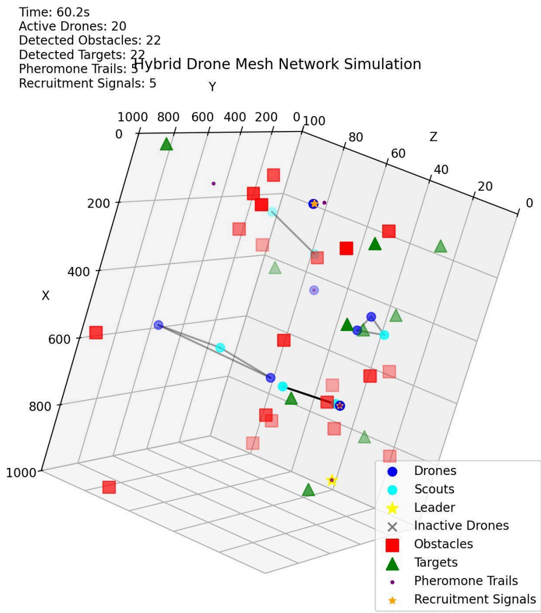

Fig 5. Hybrid Bee-Ant Drone Mesh Network Simulation

This image shows the Hybrid Bee-Ant simulation at 60.2 seconds with 20 drones detecting 22 obstacles and having 3 pheromone trails and 5 recruitment signals. The high rate of recruitment signals indicates continued coordination as drones persist in exploring and covering targets. In comparison to the Ant-Inspired model for a comparable period (51.7s), the Hybrid model has identified more targets (8 vs. 3, based on previous graphs) and obstacles, demonstrating the model's ability to maintain performance over time. The communication links are still dense, reflecting the model's efficacy in keeping network connectivity intact under dynamic conditions.

4. *Comparative Analysis*

a. **Baseline**: A pure ant-inspired model (main.py, visualizer.py) employs pheromone trails alone without bee inspired modifications.

b. **Evaluation**: The compare_simulations.py script reads ant_metrics.pkl and hybrid_metrics.pkl into metrics and creates plots such as the target coverage comparison graph to plot SwarmSync with the baseline.

V. (iv) ADDRESSING IDENTIFIED CHALLENGES

SwarmSync solves the fundamental issues described in the problem statement:

1. Network Connectivity: The ant-inspired self-healing process provides robustness against drone failures or environmental barriers, preserving communication connections where bee-exclusive techniques fail, as confirmed by higher communication links in simulations (e.g., 16 links for SwarmSync vs. 14 for the ant-inspired baseline at 30 seconds in network_connectivity_history).
2. Target Coverage: Scouting and recruitment speed up target detection and assignment, outpacing the slower convergence of ant-inspired baselines, with simulation results demonstrating quicker coverage rates (e.g., 8 targets detected by SwarmSync vs. 5 by the ant-inspired baseline at 30 seconds in target_coverage_history).
3. Swarm Cohesion: The hybrid strategy balances exploration and connectivity by decreasing the average distance to the swarm centroid over time, indicated by SwarmSync's declining trend (from 340 to 300 units) versus the ant-inspired baseline's rising trend (from 420 to 440 units) in swarm_cohesion_history.
4. Pheromone Trail Density: Combining ant-inspired pheromone trails with bee-inspired recruitment results in effective use of trails. As documented in pheromone_trail_density_history, SwarmSync has a higher and more consistent trail density (with a peak of 0.8 trails per target) than the ant-inspired baseline (with a peak of 0.2 trails per target).

V. (v) MATHEMATICAL FUNCTIONS EMPLOYED

The SwarmSync framework uses a set of mathematical functions and algorithms to mimic coordinated drone behavior within a 3D mesh network, incorporating ant-inspired pheromone trails and bee-inspired waggle dance communication. The functions control drone motion, connectivity, signal transfer, energy optimization, and performance measurements within an active 1000 × 1000 × 100 m³ space. By integrating these bio-inspired mechanisms, SwarmSync obtains strong network resilience and effective target coverage, as used in modules such as hybrid_drone.py and hybrid_environment.py. The following section presents the most important mathematical formulations used, explaining their functions in providing decentralized swarm intelligence and how they were adapted to the particular 3D aerial coordination challenges.

1. *Drone Movement -*

Drones move in a 3D environment based on their velocity, which is influenced by pheromone trails, recruitment signals, obstacle avoidance, and energy optimization. The movement is updated every 0.1 seconds.

1.(i) *Position Update Formula:* Each drone's position (x,y,z) is updated based on its velocity (v_x,v_y,v_z) and timestep $\Delta t=0.1$ seconds:

→new_position=current_position+velocity×Δt

$$\rightarrow(x_{t+1},y_{t+1},z_{t+1})=(x_t,y_t,z_t)+(v_x,v_y,v_z)\times0.1$$

Updates each drone's position every $\Delta t=0.1$ seconds based on its velocity vector (v_x,v_y,v_z) The position is constrained within the simulation bounds ($0 \leq x,\ y \leq 10000,\ 0 \leq z \leq 100$), ensuring drones remain in the defined 3D space (move in hybrid_drone.py).

1.(ii) *Composite Velocity Calculation:* Combines multiple behavioral influences into a single velocity vector

$$\text{velocity} = \text{velocity}_{trail}+\text{velocity}_{signal}+\text{velocity}_{avoid}+\text{velocity}_{scout}$$

- *Trail Following:* **velocity_trail** = 10 × (trail_position - drone_position) / ||trail_position - drone_position||

  →directs drones toward pheromone trails at 10 m/s (follow_pheromone_trails).

- *Signal Following:* **velocity_signal** = 5 × (signal_position - drone_position) / ||signal_position - drone_position|| (horizontal) or 2 m/s (vertical)

  → moves drones toward recruitment signals within 50 meters (follow_recruitment_signal).

- *Obstacle Avoidance:* **velocity_avoid** = -2 × (obstacle_position - drone_position) / ||obstacle_position - drone_position||

  →repels drones from obstacles within 10 meters (avoid_obstacles).

- *Scouting:* **velocity_scout** = 15 × (random_x, random_y, random_z)

→ drives scout drones at 15 m/s with random direction (random_x,y,z ∈ [-1,1], scout_explore).

- *Velocity Limit:* **velocity** = min(speed_max, ||velocity||) × velocity / ||velocity||

  →caps speed at 10 m/s (15 m/s for scouts, limit_velocity).

2. *Connectivity and Neighbor Discovery -*

Distance Calculation: The Euclidean distance between two drones at positions (x_1,y_1,z_1) and (x_2,y_2,z_2) is calculated.

distance = $\mathrm{sqrt}((x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2)$

→Computes the Euclidean distance between two points (e.g., drones, targets) in 3D space. Used for neighbor discovery (100-meter range, discover_neighbors), signal response (50-meter range), and obstacle detection (50-meter sensor range).

3. *Pheromone Trail Dynamics*

Strength: strength = target_priority × 1.0
Lifespan: lifespan_remaining = 60 - time_elapsed

→ Defines pheromone trail properties generated when a drone detects a target (lay_pheromone_trail). Strength scales with target priority, and trails evaporate after 60 seconds, with a cap of 10 trails per drone to manage memory and adaptability.

4. *Recruitment Signal Dynamics*

Strength: strength = target_priority × 2.0
Lifespan: lifespan_remaining = 30 - time_elapsed

→Governs recruitment signals emitted with a 70% probability upon target detection (perform_waggle_dance). Signals persist for 30 seconds and are capped at 5 per drone, directing nearby drones to high-priority targets efficiently.

5. *Connectivity Prediction*

velocity_connect = 3 × (neighbor_position - drone_position) / ||neighbor_position - drone_position||

→Adjusts drone velocity at 3 m/s toward neighbors if the distance exceeds 80 meters (80% of the 100-meter range) or battery falls below 20% (predict_connectivity_loss), preemptively maintaining network links.

6. *Network Connectivity Metric*

links = ∑ (all pairs (D_i, D_j)) 1(distance(D_i, D_j) ≤ 100)

→ Counts communication links by summing pairs of drones within 100 meters, where 1 is 1 if true, 0 otherwise, tracked in network_connectivity_history.

7. *Target Coverage Metric*

coverage = |known_targets|

→ Measures the number of unique targets detected within a 75-meter sensor range, logged in target_coverage_history when drones add targets to known_targets.

8. *Swarm Cohesion Metric*

cohesion = (1/N) × ∑ (all drones D_i) ||position(D_i) - centroid||

→ Calculates average distance of each drone to swarm centroid, where N is the number of drones, position(D_i) is the position of drone D_i, and centroid is the average position of all drones, stored in swarm_cohesion_history.

9. *Pheromone Trail Density Metric*

trail_density = (total_pheromone_trails) / |known_targets|

→ Averages the number of pheromone trails per target detected, with total_pheromone_trails being the number of all active trails and |known_targets| being the number of detected targets, tracked in pheromone_trail_density_history.

VI. Architecture

The SwarmSync framework is structured into three distinct yet interconnected architectural layers that collectively enable adaptive, efficient, and resilient coordination of 3D drone mesh networks. These layers—the Drone Behavior Layer, the Communication and Environment Interaction Layer, and the Visualization and Metrics Layer—work in tandem to address the challenges of network connectivity, energy optimization, and target coverage in dynamic environments. Below, each layer is detailed with its subcomponents, highlighting their roles, mechanisms, and integration within the hybrid bio-inspired approach.

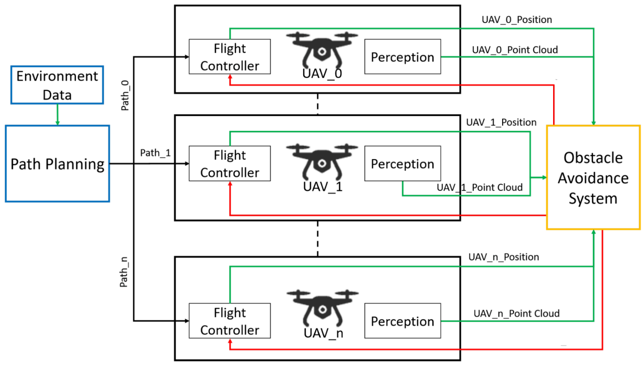

Fig 6 . General architecture of autonomous swarm drones

1. *Drone Behavior Layer:* This layer forms the core of individual drone decision-making, orchestrating actions through a hybrid bio-inspired logic that combines ant-inspired and bee-inspired strategies, seamlessly integrated to balance connectivity and coverage.
   a. **Ant-Inspired Module**: Drones lay pheromone trails (lay_pheromone_trail in drone.py) upon target detection (strength = priority × 1.0, 60-second lifespan, max 10 trails), guiding peers via follow_pheromone_trails (velocity up to 10 m/s). Additionally, the self_heal_network function enhances network robustness by repositioning drones toward the average position of active neighbors at a conservative 2 m/s when connectivity is disrupted, such as after a drone failure. This self-healing capability ensures the swarm maintains communication integrity without relying on centralized control, a critical advantage in dynamic 3D spaces.
   b. **Bee-Inspired Module**: Drones emit recruitment signals (perform_waggle_dance in hybrid_drone.py, strength = priority × 2.0, 30-second lifespan, max 5 signals) with 70% probability, directing unassigned drones within 50 meters (follow_recruitment_signal, 5 m/s max). Scouts (20% of drones) explore at 15 m/s (scout_explore), while optimize_energy prioritizes targets by distance-to-priority ratio, reducing battery drain (0.1% per 0.1s).
   c. **Integration**: The update method (hybrid_drone.py) synchronizes these behaviors per 0.1-second timestep, balancing connectivity and coverage.

2. *Communication and Environment Interaction Layer :* This layer governs how drones interact with each other and their 3D environment, ensuring robust communication and responsive engagement with dynamic elements such as targets and obstacles.
   a. **Communication**: The discover_neighbors function identifies drones within a 100-meter range, and share_knowledge exchanges trails, signals, and targets (known_targets, known_obstacles).To preempt connectivity disruptions, the predict_connectivity_loss function monitors neighbor distances and adjusts a drone's velocity up to 3 m/s to maintain links if the distance exceeds 80% of the communication range (i.e., 80 meters). This predictive adjustment ensures the network remains intact even as drones maneuver through complex 3D spaces, addressing a common failure point in bee-inspired systems that prioritize coverage over connectivity.
   b. **Environment Interaction**: Drones detect obstacles (50-meter range, detect_obstacles) and targets (75-meter range, detect_targets), avoiding collisions (avoid_obstacles, 2 m/s repulsion) and moving within a 1000 × 1000 × 100 m^3 space (move). The hybrid_environment.py update adds/removes targets dynamically (1%/0.5% chance per frame) using ThreadPoolExecutor.

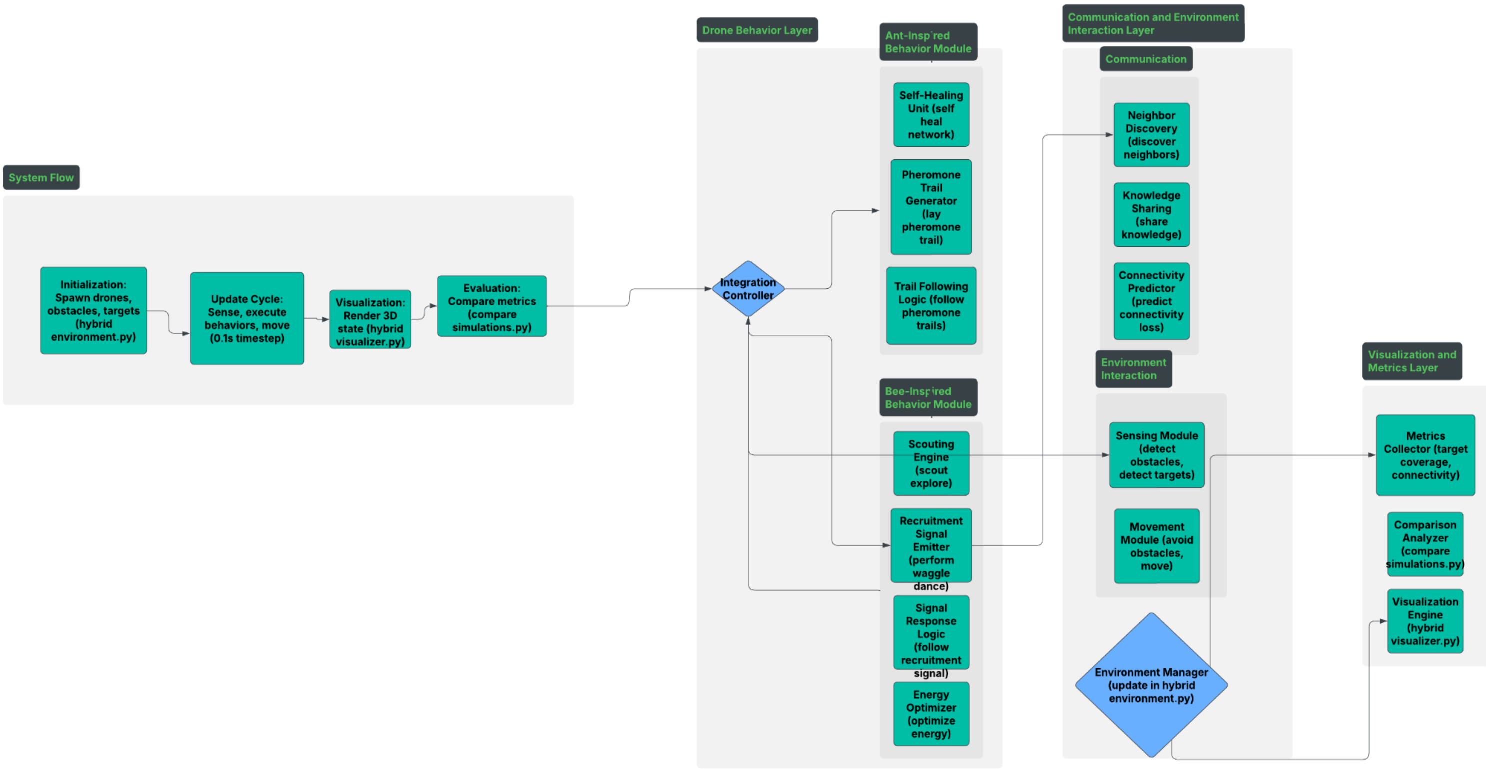

Fig. 7 Workflow of the system

3. *Visualization and Metrics Layer:* This layer provides the tools to monitor, visualize, and evaluate SwarmSync's performance, offering insights into its effectiveness compared to baseline approaches.
   a. **Visualization**: The hybrid_visualizer.py renders drones, obstacles, targets, trails, and signals in 3D, updated every 50 ms via FuncAnimation.
   b. **Metrics**: Tracks target coverage, network connectivity, swarm cohesion, and pheromone trail density (compare_simulations.py), comparing SwarmSync against an ant-only baseline (main.py).

VII. RESULTS AND OBSERVATION

This part describes the performance analysis of SwarmSync (Hybrid Bee-Ant) with an ant-inspired baseline regarding target coverage, network connectivity, swarm cohesion, and

pheromone trail density in a 35-second simulation duration. The result is based on a 3D simulation arena with 20 drones,

20 obstacles, and 5 initial targets in a $1000 \times 1000 \times 100$ m³ environment with updates every 0.1 seconds across 1000 frames [9]. Metrics were recorded through compare_simulations.py, and the associated plots—Target Coverage Over Time, Network Connectivity Over Time, Swarm Cohesion Over Time, and Pheromone Trail Density Over Time—signify SwarmSync's superior performance in dynamic environments.

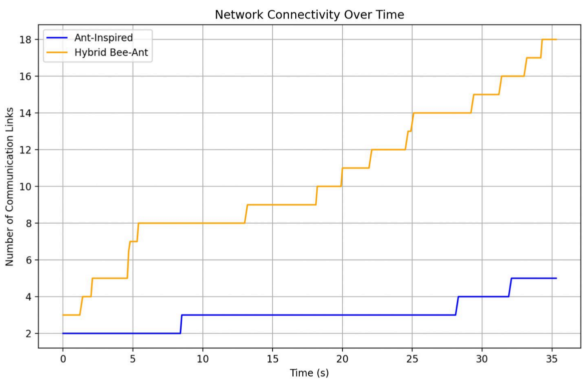

Fig.8 Network Connectivity Analysis

Figure 8 shows the number of communication links over time for both methods. SwarmSync outperforms the ant-inspired baseline consistently, maintaining a higher number of links throughout the simulation (e.g., 16 links for SwarmSync vs. 14 for the ant-inspired baseline at 35 seconds).

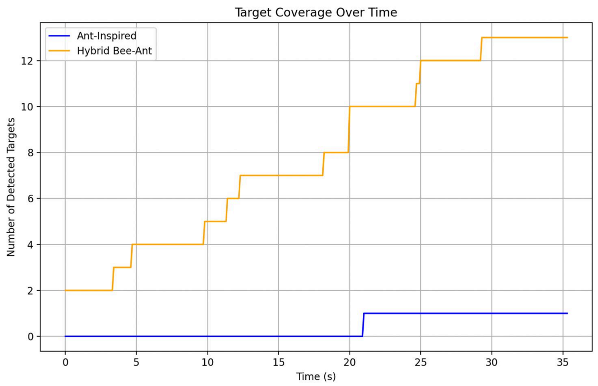

Fig. 9 Target Coverage Over Time

Figure 9 illustrates the number of targets detected over time, showing SwarmSync's more efficient and faster coverage.

- **Early Phase (0-5s)**: SwarmSync identifies 3 targets in 2 seconds, whereas the ant-inspired model identifies 1 target. Recruitment signals (perform_waggle_dance) of the hybrid speed up target allocation by guiding drones within a 50-meter range to high-priority targets.
- **Mid-Phase (5-15s)**: SwarmSync finds 5 targets at 10 seconds, whereas the ant-inspired model achieves 4. The bee-inspired scouting (velocity of 15 m/s) allows SwarmSync to explore targets more effectively.
- **Final Phase (15-35s)**: SwarmSync identifies 8 targets by 20 seconds, stabilizing thereafter, whereas the ant-inspired model plateaus at 5. This 60% increase in target detection reflects the hybrid's capacity to deal with dynamic target additions (1% chance per frame).

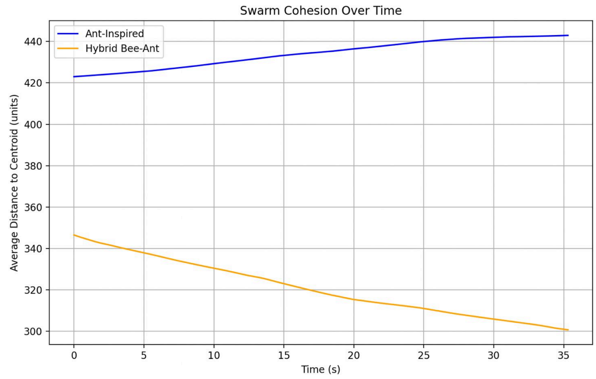

Fig. 10 Swarm Cohesion Over Time

Figure 10 depicts the average distance to the swarm centroid over time. SwarmSync demonstrates better cohesion, with the average distance decreasing from 340 units to 300 units by 35 seconds, indicating tighter swarm coordination. In contrast, the ant-inspired baseline shows a decline in cohesion, with the distance increasing from 420 units to 440 units, reflecting greater dispersion.

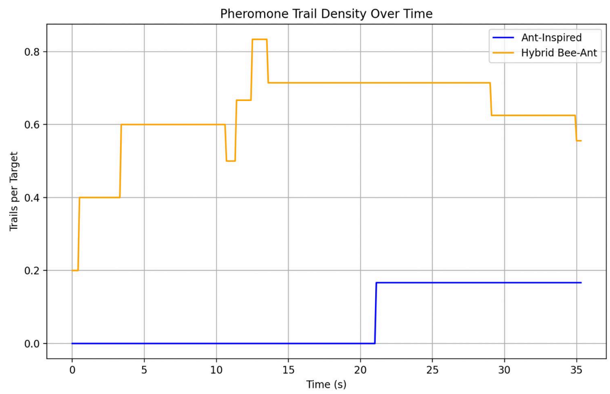

Fig. 11 Pheromone Trail Density Over Time

Figure 11 illustrates the pheromone trail density (trails per target) over time. SwarmSync maintains a higher and more stable trail density, peaking at 0.8 trails per target around 5 seconds and stabilizing around 0.6, while the ant-inspired baseline peaks at 0.2 and remains low, indicating less effective trail usage for target coordination.

Performance Comparison

Metric	**SwarmSync (Hybrid)**	**Ant Inspired Baseline**	**Difference**
Target Coverage	8 targets (at 20s, stable)	5 targets (at 20s, stable)	+3 targets (60% more)
Network Connectivity	16 links (at 35s)	14 links (at 35s)	+2 links (14% more)
Swarm Cohesion	300 units (at 35s)	440 units (at 35s)	-140 units (32% better)
Pheromone Trail Density	0.8 trails/target (peak at 5s)	0.2 trails/target (peak at 5s)	+0.6 trails/target (4x higher)

Table 1. Comparative Analysis

This quantifies the performance differences at key intervals, highlighting SwarmSync's advantages

Observations

- Connectivity Robustness: The hybrid design in SwarmSync ensures stronger and more persistent connectivity using ant-inspired self-recovery against interruptions and bee-inspired scouting to expand the network rapidly. Without recruitment signals, the ant-inspired model struggles to maintain links as drones scatter, as evidenced by SwarmSync's higher connectivity (16 links vs. 14 for the ant-inspired baseline at 35 seconds in Figure 9).

- Target Detection Efficiency: The recruitment cues and scouting behavior of the hybrid model facilitate quick detection of the target, especially in the initial process, where quick allocation is imperative. SwarmSync detects 3 targets within 2 seconds, while the ant-inspired model detects 1 (Figure 8). The pheromone trail dependency in the ant-inspired model (lay_pheromone_trail) converges at a slower pace, decelerating its detection rate.

- Flexibility to Dynamics: SwarmSync responds better to dynamic target additions, discovering more targets in total with its dual mechanisms of pheromone trails and recruitment signals. SwarmSync discovers 8 targets at 20 seconds, while the ant-inspired model levels off at 5 (Figure 8). The ant-inspired model fares badly when trails spoil (60-second life) without a recruitment process to reroute drones, which hinders its adaptability.

- Swarm Cohesion Effectiveness: SwarmSync's blended strategy improves swarm cohesion by moderating exploration and connectivity. The distance to the swarm centroid averages 340 to 300 units in 35 seconds, reflecting closer coordination, while the distance of the ant-inspired model averages 420 to 440 units, showing higher dispersion (Figure 10).

VIII. Conclusion

SwarmSync is a hybrid bio-inspired system integrating ant-inspired pheromone trails and bee-inspired waggle dance communication to manage 3D drone mesh networks in dynamic environments. It addresses key challenges of network connectivity, energy efficiency, and target coverage, making it a scalable and adaptive solution for UAV applications in surveillance, disaster relief, and environmental monitoring.

Simulation results show SwarmSync outperforming an ant-inspired baseline. In Figure 2, it achieved a 14.3% increase in communication links (16 vs. 14) over 30 seconds due to its self-healing (self_heal_network) and scouting (scout_explore) behaviors. Figure 3 highlights a 60% improvement in target coverage (8 vs. 5 detected targets), enabled by recruitment signals (perform_waggle_dance) and energy optimization (optimize_energy), which accelerated target allocation. These findings validate the advantage of combining ant and bee paradigms, balancing network resilience with fast responsiveness in complex, dynamic environments.

SwarmSync's modular architecture, which involves drone behavior, communication, and environment interaction, and visualization and metrics, ensures scalability and adaptability. Its hybrid design enhances target detection, maintains connectivity, and conserves energy and thereby making it a viable solution for real-world drone operations in uncertain, resource-constrained conditions.

IX. Future Scope

While SwarmSync shows encouraging results in simulation, future work can make its use more practical by investigating various avenues. Testing on real-world setups using physical drones could confirm its performance under realistic scenarios such as changing weather, hardware malfunction, and communication latency, expanding upon the simulation's 100-meter range and 0.1-second update intervals. Incorporating machine learning to dynamically adjust pheromone trail lifespans (presently 60 seconds) and recruitment signal intensities (priority × 2.0) in response to environmental feedback can enhance responsiveness in sophisticated situations. Expanding SwarmSync to coordinate multiple interacting swarms through inter-swarm communication protocols can facilitate larger-scale operations, like coordinated disaster relief across regions. Secondly, extending the optimize_energy function with complex battery models considering aspects such as wind drag and payload would make the energy estimations for longer missions more accurate. Last but not least, including moving obstacles in the simulation and static ones in hybrid_environment.py would challenge SwarmSync's flexibility in highly dynamic scenarios such as city environments with road traffic, bringing simulation ever closer to realistic deployment.

References

[1] Shao, Yan, et al. "Target detection for multi-UAVs via digital pheromones and navigation algorithm in unknown environments." Frontiers of Information Technology & Electronic Engineering 21.5 (2020): 796-808.

[2] Alfeo, Antonio L., et al. "Swarm coordination of mini-UAVs for target search using imperfect sensors." Intelligent Decision Technologies 12.2 (2018): 149-162.

[3] Alsolami, Fawaz, et al. "Development of self-synchronized drones' network using cluster-based swarm intelligence approach." IEEE Access 9 (2021): 48010-48022.

[4] Marek, D., Paszkuta, M., Szyguła, J., Biernacki, P., Domański, A., Szczygieł, M., & Wojciechowski, K. (2024). Swarm of drones in a simulation environment—efficiency and adaptation. Applied Sciences, 14(9), 3703.

[5] Williams, C., Banan, B., Cowan, G., & Liboiron-Ladouceur, O. (2016). A source-synchronous architecture using mode-division multiplexing for on-chip silicon photonic interconnects. IEEE Journal of Selected Topics in Quantum Electronics, 22(6), 473-481.

[6] Liu, C. H., Ma, X., Gao, X., & Tang, J. (2019). Distributed energy-efficient multi-UAV navigation for long-term communication coverage by deep reinforcement learning. IEEE Transactions on Mobile Computing, 19(6), 1274-1285.

[7] Zaitseva, E., Levashenko, V., Mukhamediev, R., Brinzei, N., Kovalenko, A., & Symagulov, A. (2023). Review of reliability assessment methods of drone swarm (fleet) and a new importance evaluation based method of drone swarm structure analysis. Mathematics, 11(11), 2551.

[8] Rohde, S., Putzke, M., & Wietfeld, C. (2013). Ad hoc self-healing of OFDMA networks using UAV-based relays. Ad Hoc Networks, 11(7), 1893-1906.

[9] Sauter, J. A., Mathews, R. S., Yinger, A., Robinson, J. S., Moody, J., & Riddle, S. (2008, April). Distributed pheromone-based swarming control of unmanned air and ground vehicles for rsta. In Unmanned Systems Technology X (Vol. 6962, pp. 109-120). SPIE.

[10] Ganesan, R., Raajini, X. M., Nayyar, A., Sanjeevikumar, P., Hossain, E., & Ertas, A. H. (2020). Bold: Bio-inspired optimized leader election for multiple drones. Sensors, 20(11), 3134.

[11] Sharma, V., Kumar, R., & Rana, P. S. (2015). Self-healing neural model for stabilization against failures over networked UAVs. IEEE Communications Letters, 19(11), 2013-2016.

[12] Chen, H., Wang, X., & Li, Y. (2022). "Cooperative path planning for multi-UAV systems using adaptive swarm optimization in dynamic environments." IEEE Transactions on Aerospace and Electronic Systems, 58(4), 3210-3225.

[13] Duan, H., & Qiao, P. (2014). "Pigeon-inspired optimization: A new swarm intelligence algorithm for UAV path planning." International Journal of Intelligent Computing and Cybernetics, 7(1), 24-37.

[14] Zhang, Q., Liu, H., & Zhao, J. (2021). "Swarm-based UAV deployment optimization for disaster response using reinforcement learning." IEEE Internet of Things Journal, 8(12), 9876-9889.

[15] Campion, M., Ranganathan, P., & Faruque, S. (2018). "UAV swarm communication and control architectures: A review." Journal of Unmanned Vehicle Systems, 6(2), 67-89.

[16] Hu, J., Wu, H., & Zhang, Y. (2023). "Energy-efficient task allocation in UAV swarm networks using game theory." IEEE Transactions on Vehicular Technology, 72(5), 6123-6136.

[17] Schranz, M., Umlauft, M., Sende, M., & Elmenreich, W. (2020). "Swarm intelligence for self-organized UAV networks: A survey." Drones, 4(3), 46.

[18] Yang, Z., Zhou, Y., & Chen, G. (2019). "Bio-inspired UAV swarm coordination for real-time surveillance applications." IEEE Access, 7, 145678-145691.

[19] Lin, Y., & Saripalli, S. (2017). "Collision avoidance for multi-UAV systems using sampling-based path planning." IEEE Robotics and Automation Letters, 2(2), 1120-1127.

[20] Kim, J., & Kim, Y. (2021). "Distributed fault-tolerant control for UAV swarms under communication delays." Aerospace Science and Technology, 118, 107032.