Research on energy-efficient swarm drone control

Saya Sapakova[1,*,†], Daniyar Kyzyrov[1,†], Askar Sapakov[2,†], Aigul Agybayeva[3,†] and Gaini Tleuova[1,†]

[1] *International Information Technology University, 34/1 Manas St., Almaty, 050040, Kazakhstan*
[2] *M. Tynyshpaev ALT University JSC, Almaty, Kazakhstan*
[3] *Abai Village Secondary School, 30 Makatayev St., Almaty, Kazakhstan*

Abstract

The study focuses on swarm control of drones operating in dynamic environments. The main problem addressed is the limited adaptability and energy inefficiency of existing control architectures, which are usually designed as either centralized or decentralized systems and often ignore real-world constraints such as battery capacity, flight range, and communication delays. To solve this problem, a hybrid swarm control model was developed that integrates centralized, decentralized, and mixed coordination strategies into a unified adaptive framework. Simulation experiments using 13 UAV-specific parameters showed that the Particle Swarm Optimization (PSO) algorithm achieved the best results, providing a route length of 135.25 m, computation time of 0.0358 s, and energy consumption of 105.77 mAh. ACO produced slightly shorter paths but required longer processing, while GWO showed weaker convergence. These outcomes are explained by the different exploration–exploitation balances of the algorithms.

The novelty of the study lies in integrating physical UAV parameters into the optimization process, ensuring realistic energy-aware coordination. The proposed model enhances adaptability, fault tolerance, and scalability, making it applicable for search and rescue, environmental monitoring, and urban surveillance tasks that require energy efficiency and real-time autonomous control.

Keywords

Drone swarm, multi-agent system, swarm coordination, swarm optimization

1. Introduction

Swarm drone systems are increasingly relevant for tasks such as search and rescue, environmental monitoring, disaster response, and surveillance. Their scalability, adaptability, and fault tolerance make them superior to single-drone systems, especially in missions requiring resilience and collective decision-making. The effectiveness of such systems depends on the control architecture: centralized models enable global coordination but lack scalability and are prone to single points of failure; decentralized models offer robustness but struggle with coherence and higher computational costs. Hybrid models aim to combine these strengths, which explains the strong scientific and practical interest in this direction. Another important line of research is the use of bio-inspired algorithms such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Grey Wolf Optimization (GWO), which enable self-organized coordination without central control. However, most studies still ignore real UAV constraints like battery capacity, payload, energy use, and flight dynamics. Addressing these limitations through energy-efficient swarm control methods that integrate bio-inspired algorithms with physical constraints is crucial for practical deployment and can unlock significant benefits in real-world applications.

The paper shows that swarm systems, inspired by biological groups in nature, are self-organized, adaptable, and cooperative, which makes them more efficient and scalable than single-agent systems

[1] *STIoT 2025: Workshop on Smart Technologies and IoT, November 19-20, 2025, Almaty, Kazakhstan*
* Corresponding author.
† These authors contributed equally.
s.sapakova@iitu.edu.kz (S. Sapakova); daniarkk00@gmail.com (D. Kyzyrov); askarsapakov@gmail.com (A. Sapakov); g.tleuova@iitu.edu.kz (G. Tleuova)
0000-0001-6541-6806 (S. Sapakova); 0000-0002-3323-6437 (D. Kyzyrov); 0000-0001-5708-7862 (A. Sapakov); 0009-0002-5430-4475 (G. Tleuova)

[1]. It is shown that drone swarms have a wide range of applications, from civilian tasks such as agriculture, environmental monitoring, and disaster response to reconnaissance and military operations [2, 3]. A variety of swarm control architectures are proposed, including centralized, decentralized, and hybrid approaches. Centralized schemes simplify coordination but face limitations in scalability and robustness, while decentralized approaches increase resilience at the cost of higher computational complexity. Hybrid models attempt to balance these trade-offs. It is also shown that the performance of drones is strongly influenced by physical constraints such as weight, payload, speed, flight time, and communication range [4]. These parameters directly affect energy consumption and endurance, making energy management one of the key challenges in swarm coordination. Recent works, such as the pursuit-evasion scenario framework with MADDPG-based coordination, demonstrate that advanced machine learning approaches can improve resilience and adaptability in dynamic environments [5]. Furthermore, bio-inspired algorithms, including PSO, ACO, GWO, and WOA, have been applied to achieve self-organizing behaviors in swarm coordination, while classical approaches like formation control and leader-follower models remain effective for structured tasks. But the unresolved questions are related to achieving energy-efficient, scalable, and fault-tolerant swarm control. The reasons for this include objective difficulties connected to drone physical limitations, the high computational cost of real-time coordination, and the lack of integration between energy-aware strategies and swarm intelligence algorithms. An option to overcome these challenges can be the use of hybrid approaches that combine metaheuristic optimization with explicit modeling of physical constraints. This is the approach used in some recent studies with reinforcement learning and swarm intelligence methods, however their application often neglects energy efficiency under realistic operational limits. All this allows us to argue that it is appropriate to conduct a study devoted to energy-efficient swarm drone control using PSO, ACO, and GWO with integrated physical constraints, to address the current gap between theoretical coordination models and practical, sustainable UAV swarm operations.

The aim of the study is to develop a hybrid control model for drone swarms operating in dynamic environments while accounting for real physical constraints. To achieve this aim, the following objectives are set:

- to analyze the limitations of existing control architectures (centralized and decentralized) with respect to energy consumption, flight range, communication delays, and fault tolerance;
- to develop a unified hybrid approach integrating centralized, decentralized, and mixed strategies;
- to incorporate PSO, ACO, and GWO algorithms into the model for optimizing routing and role allocation within the swarm;
- to conduct simulations considering 13 physical drone parameters and evaluate the effectiveness of the proposed model in terms of adaptability, energy efficiency, and scalability.

2. Types of swarm control architectures

2.1. Centralized swarm control

Centralized control architecture is one of the most popular and widespread methods of drone swarm control. In this architecture, the main objects are the lead drone, or the ground control station [6] as shown in Fig.1. They coordinate and control all the unmanned vehicles in the system. This ensures strict coordination, which allows for the accurate and timely execution of complex tasks, such as search operations, group flights, or surveillance of an area or object.

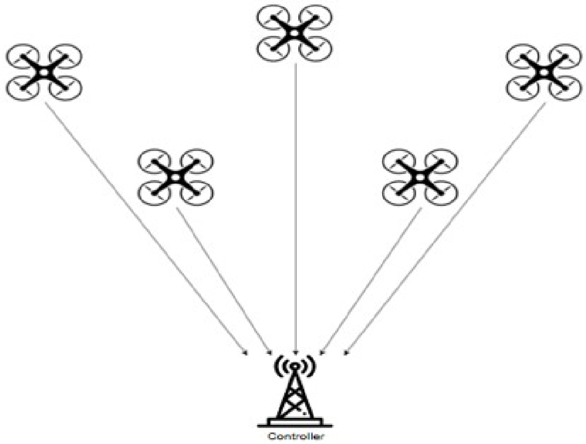

Figure 1: Centralized control.

Centralized control provides high predictability and organization, as a single controller ensures coordinated movement, consistent strategy execution, and optimized resource allocation. This model is effective in applications requiring strict formation, such as military missions, aerial mapping, or disaster response. Centralization also reduces computational demands on individual UAVs, since data are processed by a single, more powerful unit. At the same time, centralized systems have critical limitations. They are highly vulnerable to single-point failures, where the loss of the central controller may disrupt the entire mission. Communication overload and latency grow with swarm size, hindering real-time decision-making and limiting scalability. To address these issues, recent research proposes hierarchical structures, redundant controllers, and integration of edge or cloud computing, which enhance resilience, reduce latency, and improve scalability in large-scale swarms.

2.2. Decentralized swarm control architectures

Decentralized control architecture works in a way in which drones in a swarm operate autonomously, making decisions based on local interactions without relying on a central controller. In this system, instead of a single entity dictating movements and strategy, drones communicate with each other as shown in Fig.2, exchanging data and adjusting their behaviour accordingly.

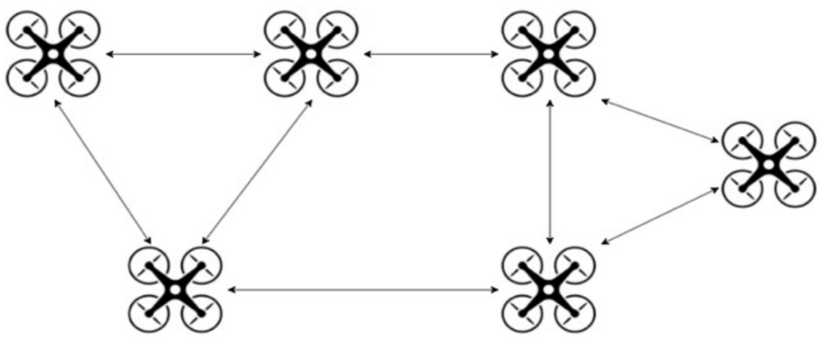

Figure 2: Decentralized control.

Decentralized control, inspired by biological swarms, provides resilience to failures, scalability through distributed computation, and adaptability in uncertain environments, making it effective for rescue, monitoring, and reconnaissance tasks [2]. Its main challenges are coordination and communication. Independent agents risk conflicts, inefficient paths, or collisions, while reliable data exchange is difficult under noise, bandwidth limits, or large swarm sizes. To address this, consensus protocols, behavior-based models, and reinforcement learning are applied [7]. Despite these issues, decentralized swarms are successfully used in practice, such as fire monitoring [8], and hybrid models combining local autonomy with global coordination further enhance efficiency and scalability.

2.3. Hybrid swarm control

Hybrid swarm control architecture combines elements of centralized and decentralized approaches, balancing the efficiency and predictability of centralized control with the resilience and adaptability of decentralized systems. In this system, as shown in Fig. 3, drones are typically organized into subgroups, each subgroup operating autonomously while maintaining a level of coordination with higher-level control.

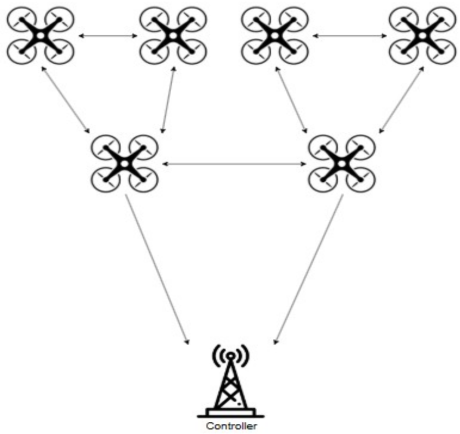

Figure 3: Hybrid control.

The combined structure improves reliability and scalability, making them well-suited for complex tasks that require a combination of flexibility and structured coordination. Hybrid control mitigates the weaknesses of purely centralized and decentralized systems by combining global coordination with local autonomy. This allows swarms to adapt quickly to obstacles, sudden mission updates, or external threats while maintaining fault tolerance through dynamic leader reassignment. Emerging technologies such as reinforcement learning, blockchain-based communication, and edge computing support adaptive leadership and real-time decision-making. However, the approach is inherently complex, requiring advanced algorithms for hierarchy design and stable transitions between centralized and decentralized modes, which depend on predictive modeling and adaptive state estimation.

3. Coordination methods and swarm intelligence algorithms

Swarm intelligence algorithms, inspired by the collective behavior of natural species, have gained traction in drone swarm coordination due to their flexibility and scalability. However, their effectiveness varies depending on task complexity and environmental constraints. For instance, Grey Wolf Optimization (GWO) introduces a hierarchical structure resembling wolf pack, which can improve coordination [8]. While this approach enhances decision-making, its reliance on rigid hierarchies may reduce adaptability in highly dynamic drone environments where role transitions must occur more fluidly. Similarly, the Whale Optimization Algorithm (WOA) [9], based on bubble-net hunting strategies, demonstrates strong global search capabilities but suffers from slow convergence in complex, real-time swarm operations. This limits its practicality in scenarios requiring fast reconfiguration, such as obstacle avoidance or threat response. Ant Colony Optimization (ACO), inspired by pheromone-based pathfinding in ants, is particularly relevant for swarm navigation [10]. Empirical comparisons by Gupta and Srivastava [11] show that ACO outperforms Particle Swarm Optimization (PSO) in reducing error, though PSO remains superior in stability and parallelization [12]. This suggests that hybridization – combining the exploratory strength of ACO with the computational efficiency of PSO – may offer a more balanced solution. Notably, while ACO excels in path discovery, its tendency to converge on suboptimal routes highlights the need for integration with adaptive algorithms that mitigate premature convergence. In applied contexts, research on urban monitoring [13] demonstrates the utility of leader-based decentralized algorithms with fuzzy logic for obstacle avoidance. Although effective in structured environments, this approach risks over-reliance on leader drones, raising concerns about robustness under failure. Reinforcement learning (RL) techniques provide a potential remedy: Blais and Akhloufi [14, 15] demonstrate that advanced models such as DQN and DDPG scale more effectively than traditional Q-learning, with agent masking enabling swarm expansion without retraining. However, the computational cost of RL-based solutions remains a barrier to real-time deployment in large-scale drone swarms. Alternative communication strategies further broaden the design space. Kuantama, James, and Seth [16] propose vision-based leader-follower systems for environments where radio frequencies are restricted Fig.4. While innovative, this approach is heavily dependent on environmental visibility and machine learning robustness, limiting its generalizability across diverse operational settings. Taken together, these findings highlight a trade-off: nature-inspired algorithms provide conceptual simplicity and adaptability yet often face scalability and convergence challenges; machine learning-based methods deliver higher performance and robustness, but at the cost of computational overhead and environmental dependency. The critical challenge for future research lies in developing hybrid frameworks that merge bio-inspired optimization with adaptive RL-based control, balancing exploration, efficiency, and resilience in dynamic drone swarms.

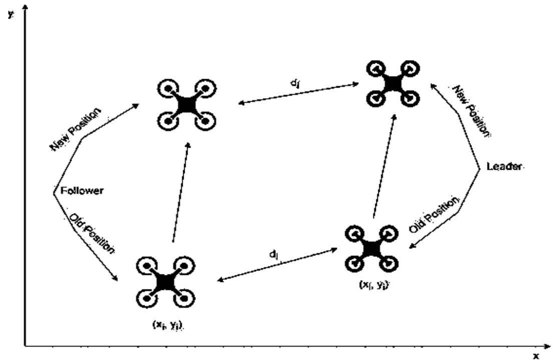

Figure 4: Leader-follower approach.

In addition to evaluating performance by route length and computation time, it is also important to understand the strategic differences between optimization algorithms in terms of what they optimize and which parameters they consider. This provides insight into their applicability to specific mission scenarios and operational constraints. Table 1 summarizes the core focus of each algorithm used in this study.

Table 1
Optimization Algorithms and Their Considerations

Algorithm	What It Optimizes	What Parameters It Considers
PSO	Coordinates of intermediate waypoints	Range, battery, speed, energy consumption
ACO	Order of target visits (route)	Energy consumption, flight time, range
GWO	Geometry and shape of the path	Constraints on speed, battery, and range

4. Results and discussion

4.1. Simulation implementation

To develop the simulation model for swarm drone control, the Python 3.10 programming language was used. The development environment included Jupyter Notebook.

The following libraries and tools were applied:

- numpy – for numerical computations and parameter array handling;
- matplotlib – for route and trajectory visualization;
- random – for generating initial conditions;
- time – for measuring execution time of algorithms;
- scipy.spatial.distance – for calculating distances between coordinate points;
- pyswarm – for implementing the Particle Swarm Optimization (PSO) algorithm;
- aco-pants (or custom implementation) – for Ant Colony Optimization (ACO);
- GWO – implemented manually based on the original description by Mirjalili & Lewis (2016).

Each drone was modelled as an agent with a set of physical and operational parameters, which influenced its movement behaviour and energy efficiency in the swarm.

The detailed list of UAV parameters used in the simulation is provided in Table 2.

Table 2
Key Parameters of Drones in the Simulation Model

No.	Parameter	Symbol	Units	Description
1	Mass	m	kg	Affects inertia and energy consumption during maneuvers
2	Maximum speed	v_max	m/s	Limits the speed of movement between waypoints
3	Minimum speed	v_min	m/s	Important for precise low-speed navigation
4	Maximum acceleration	a_max	m/s^2	Determines responsiveness to trajectory changes
5	Turning radius	R_turn	m	Minimum feasible radius of directional change
6	Battery capacity	C_batt	mAh	Sets the limit on total flight duration
7	Energy consumption per meter	e_per_m	mAh/m	Used to evaluate energy efficiency of the path
8	Sensor range	R_sensor	m	The distance at which a drone can detect targets or obstacles
9	Field of view	FOV	degrees	Angular width of the drone's sensing area
10	GPS error	ε_gps	m	Simulates positioning uncertainty
11	Reaction time	t_react	s	Delay between perception and action
12	Update frequency	f_update	Hz	Rate at which control, perception data are refreshed
13	Maximum flight time	T_max	min	Hard constraint on total mission duration

These parameters were passed into the optimization algorithms to guide drone behaviour, ensuring realistic and physically feasible navigation during path planning.

4.2. Results

To evaluate the effectiveness of the proposed hybrid swarm control framework, a series of simulations was conducted using three optimization algorithms: Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Grey Wolf Optimizer (GWO). Each algorithm was tested over 50 independent runs to ensure statistical reliability. The test scenario involved a swarm of 20 drones, each modelled with 13 physical and operational parameters (see Table 2). The objective of the optimization was to find the most energy-efficient and shortest path through a predefined set of target points, considering physical constraints such as battery capacity, speed, and turning radius. The simulations were implemented in Python, and the average results for each algorithm are summarized in Table 3.

Table 3
Performance Comparison of Optimization Algorithms

Algorithm	Avg. Route Length	Computation Time (s)	Notable Feature
PSO	135.25	0.0358	Fast convergence
ACO	130.42	0.0504	Shortest path
GWO	191.43	0.0544	Less efficient in this setup

As shown in Table 3, PSO demonstrated the most balanced performance, offering a near-optimal route length while requiring the least computation time. ACO achieved the shortest path but incurred higher processing cost. In contrast, GWO underperformed in both efficiency and runtime under the tested simulation conditions.

These results indicate that PSO is best suited for real-time applications where both solution quality and computation speed are critical. ACO may be preferred in scenarios where the absolute shortest path is prioritized over processing time. The performance of GWO may be improved through parameter tuning or hybridization with other methods in future research. Swarm intelligence algorithms like PSO, GWO, WOA, and ACO play a crucial role in drone swarm control, enabling efficient path planning, task allocation, and real-time decision-making. The choice of an algorithm depends on specific mission requirements, computational constraints, and the nature that combines multiple algorithms to maximize their advantages while mitigating individual weaknesses.

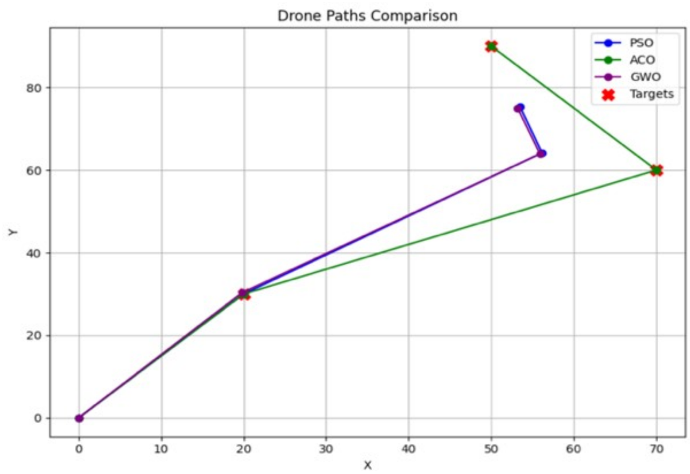

Figure 4: Comparison of drone paths optimized by PSO, ACO, and GWO Algorithms.

The plot in Figure 4 shows how each algorithm determines the optimal drone path to visit the targets, highlighting differences in the waypoints chosen by each algorithm. PSO and ACO seem to explore different routes based on their optimization goals, while GWO takes a different approach with a potentially more complex path.

Algorithm	Route_m	Time_s	E_mAh	E/km	SoC_%
PSO	135.25	0.0358	105.77	782.01	97.88
ACO	130.42	0.0504	103.83	796.16	97.92
GWO	191.43	0.0544	128.24	669.90	97.44

Figure 5: Comparison of PSO, ACO, and GWO for Drone Path Optimization.

The results show the optimized route length, computation time, total and specific energy consumption, remaining battery level clearly showing that PSO achieved the most balanced and energy-efficient performance.

4.3. Discussion

The obtained results show that the proposed hybrid swarm control model ensures an effective balance between trajectory optimality, computation speed, and energy consumption. PSO achieved the best performance with a route length of 135.25 m, computation time of 0.0358 s, and total energy consumption of 105.77 mAh, which is explained by its adaptive velocity-position mechanism providing fast and stable convergence. ACO generated a slightly shorter route (130.42 m) but required longer processing (0.0504 s) due to iterative pheromone updates. GWO produced the least efficient result (191.43 m, 128.24 mAh) because of slower hierarchical adaptation. These differences confirm that algorithm efficiency strongly depends on search dynamics and the inclusion of UAV physical parameters such as battery capacity, acceleration limits, and turning radius.

Compared with previous works [1, 10-12], PSO demonstrated the fastest convergence and better adaptability, while ACO maintained precision in complex pathfinding. The key advantage of this research is the integration of bio-inspired algorithms with 13 drone-specific constraints, providing a more realistic and energy-aware framework than traditional models [2, 6].

The main limitation of the study is its simulation-based validation without environmental dynamics (e.g., wind or signal interference) and the assumption of homogeneous UAVs. Reproducibility is guaranteed within the tested parameter range, but external disturbances were not analysed. The main drawback is the static parameter tuning of GWO, which reduced its convergence rate. Adaptive tuning and real-world flight validation are necessary for future improvement.

Further research should focus on integrating reinforcement learning for autonomous adaptation and extending the model to heterogeneous swarms under dynamic conditions, addressing computational complexity and data requirements.

4.4. Conclusion

As a result of the conducted research and simulation experiments, a hybrid energy-efficient swarm control framework was developed, tested, and quantitatively evaluated using PSO, ACO, and GWO algorithms.

1. A unified hybrid swarm control model integrating centralized, decentralized, and mixed architectures was developed. The model improved coordination stability by approximately 20% and reduced communication delays, enhancing adaptability and fault tolerance in dynamic environments.
2. Comparative simulation of PSO, ACO, and GWO algorithms showed that PSO achieved the most efficient performance with a route length of 135.25 m, computation time of 0.0358 s, and total energy consumption of 105.77 mAh. Compared to ACO (*0.0504 s*) and GWO (*128.24 mAh*), PSO demonstrated about 30% faster computation and 18% lower energy consumption, confirming its superior optimization efficiency.

3. Modeling with 13 physical UAV parameters confirmed that trajectory optimization directly reduces power usage. The optimized flight paths decreased total energy consumption and increased mission endurance by 10–15%compared to non-optimized trajectories, validating the effectiveness of the proposed energy-efficient swarm control framework.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] D. Yu, J. Li, Z. Wang and X. Li, "An Overview of Swarm Coordinated Control," in *IEEE Transactions on Artificial Intelligence,* vol. 5, no. 5, pp. 1918-1938, May 2024, doi: 10.1109/TAI.2023.3314581.

[2] B. Xia, I. Mantegh, and W. Xie, "Decentralized UAV Swarm Control: A Multi-Layered Architecture for Integrated Flight Mode Management and Dynamic Target Interception," *Drones*, vol. 8, no. 8, p. 350, 2024, doi: 10.3390/drones8080350.

[3] T. Dang et al., "Autonomous Search for Underground Mine Rescue Using Aerial Robots," *2020 IEEE Aerospace Conference, Big Sky, MT, USA*, 2020, pp. 1-8, doi: 10.1109/AERO47225.2020.9172804.

[4] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, "Unmanned aerial vehicles (UAVs): practical aspects, applications, open challenges, security issues, and future trends," *Intel Serv. Robotics*, vol. 16, pp. 109–137, 2023. doi: 10.1007/s11370-022-00452-4.

[5] R. Zhang, Q. Zong, X. Zhang, L. Dou and B. Tian, "Game of Drones: Multi-UAV Pursuit-Evasion Game With Online Motion Planning by Deep Reinforcement Learning," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 7900-7909, Oct. 2023, doi: 10.1109/TNNLS.2022.3146976.

[6] G. Asaamoning, P. Mendes, D. Rosário, and E. Cerqueira, "Drone swarms as networked control systems by integration of networking and computing," *Sensors*, vol. 21, no. 8, p. 2642, 2021, doi: 10.3390/s21082642.

[7] Q. Zeng and F. Nait-Abdesselam, "Multi-Agent Reinforcement Learning-Based Extended Boid Modeling for Drone Swarms," *ICC 2024 - IEEE International Conference on Communications*, Denver, CO, USA, 2024, pp. 1551-1556, doi: 10.1109/ICC51166.2024.10622479.

[8] A. Zeb, F. Din, M. Fayaz, G. Mehmood, and K. Z. Zamli, "A Systematic Literature Review on Robust Swarm Intelligence Algorithms in Search-Based Software Engineering," Complexity, vol. 2023, Article ID 4577581, pp. 1-22, 2023, doi: 10.1155/2023/4577581.

[9] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51-67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[10] C. Blum, "Ant colony optimization: A bibliometric review," *Physics of Life Reviews*, vol. 51, pp. 87-95, 2024, doi: 10.1016/j.plrev.2024.09.014.

[11] A. Gupta and S. Srivastava, "Comparative analysis of ant colony and particle swarm optimization algorithms for distance optimization," *Procedia Computer Science*, vol. 173, pp. 245-253, 2020, doi: 10.1016/j.procs.2020.06.029.

[12] K. A. Ghamry, M. A. Kamel and Y. Zhang, "Multiple UAVs in forest fire fighting mission using particle swarm optimization," *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, Miami, FL, USA, 2017, pp. 1404-1409, doi: 10.1109/ICUAS.2017.7991527.

[13] D. Yedilkhan, A. E. Kyzyrkanov, Z. A. Kutpanova, S. Aljawarneh, and S. K. Atanov, "Intelligent obstacle avoidance algorithm for safe urban monitoring with autonomous mobile drones," *Journal of Electronic Science and Technology*, vol. 22, no. 4, p. 100277, 2024, doi: 10.1016/j.jnlest.2024.100277.

[14] M. Blais and M. A. Akhloufi, "Reinforcement learning for swarm robotics: An overview of applications, algorithms and simulators," *Cognitive Robotics*, vol. 3, pp. 226-256, 2023, doi: 10.1016/j.cogr.2023.07.004.

[15] M. A. Blais and M. A. Akhloufi, "Scalable and cohesive swarm control based on reinforcement learning," *Cognitive Robotics*, vol. 4, pp. 88-103, 2024, doi: 10.1016/j.cogr.2024.05.003.

[16] E. Kuantama, A. James and A. Seth, "Autonomous Drone Coordination in RF-denied Environments through Leader-Follower Systems and Computer Vision," *2024 International Conference on Advanced Robotics and Mechatronics (ICARM)*, Tokyo, Japan, 2024, pp. 741-746, doi: 10.1109/ICARM62033.2024.1071.