DOI: 10.53375/ijecer.2025.465

INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING RESEARCH
VOL. 5, NO. 2, 2025

Algorithm-Driven Multi-User Platform for Decentralized Coordination in Self-Organizing UAV Swarms

Agit Atashyan[1,*], Artyom Lazyan[2], Davit Hayrapetyan[3], Vahagn Poghosyan[4], Suren Poghosyan[5]
Models of Computing and Logical Reasoning Department,
Institute for Informatics and Automation Problem of NSA, Armenia,
Emails:[1]agitatashyan1@gmail.com, [2]artyomlazyan@gmail.com, [3]hayrapetyan96@gmail.com, [4]povahagn@gmail.com, [5]psuren55@yandex.ru
[*] Corresponding author

Abstract: **This paper introduces a comprehensive software platform designed to coordinate self-organizing UAV swarms through a secure and modular client-server system. Developed with multi-user collaboration in mind, the platform features an intuitive, cross-platform interface that allows users to define mission tasks, construct navigation graphs, and monitor swarm activity in real time.**

At the heart of the system is a robust path-planning algorithm based on the rotor-router model with loop reversibility, which enables reliable and evenly distributed task coverage without relying on randomness. To enhance fault tolerance and ensure resilience in communication-limited environments, the platform employs a gossip-based broadcast algorithm. This allows swarm members to share information efficiently and maintain coordinated behaviour, even when some nodes experience failures or connectivity issues.

A built-in simulation module enables users to test and refine swarm coordination strategies before deployment, reducing operational risk and improving mission reliability. By simulating various environmental conditions and mission scenarios, users can evaluate system behaviour and optimize task execution. In parallel, the platform supports real-time 3D panorama generation from UAV-captured images, providing rich visual context and enabling more effective post-mission analysis.

Taken together, these features form a scalable, secure, and highly flexible system for managing decentralized drone swarms. The platform is well-suited for applications that demand coordination across multiple agents, including environmental monitoring, search and rescue, infrastructure inspection, and autonomous exploration. It bridges theoretical rigor with practical usability, offering a reliable toolset for both researchers and mission operators. Our work builds on earlier systems, introducing hybrid rotor-router initialization, algorithmic no-fly zone enforcement, and dual-toolchain image stitching.

Keywords: **Autonomous Exploration, Decentralized Coordination, Fault-tolerant Communication, Rotor-router Path Planning, UAV Swarm Management.**

I. INTRODUCTION

The Decentralized UAV Swarm Task Management Platform introduces a powerful and flexible approach to coordinating large-scale drone operations without relying on centralized control. Built on a modular, scalable, and fault-tolerant client–multi-server architecture, the platform supports collaborative mission planning, decentralized task scheduling, and seamless integration across distributed systems. The coordination of UAV swarms in distributed, communication-constrained environments poses unique challenges. Many so-called self-organizing platforms rely on local autonomy but do not achieve emergent global coordination. In contrast, this work offers a truly self-organizing system where deterministic behavior arises from local rules, with no need for centralized control. This study builds on foundational work described in [1], extending the platform with additional algorithmic modules and operational capabilities. These include the integration of panoramic reconstruction tools, spatial constraint enforcement via topological markings, and improved scalability via inner-outer rotor initialization strategies. By enhancing mission setup, pre-deployment validation, and real-time imaging, the platform delivers a complete toolset for autonomous UAV coordination.

Two key algorithmic strategies form the backbone of the system. First, the rotor-router model enables drones to follow deterministic and loop-reversible paths. This ensures that tasks are evenly distributed and all areas are covered without overlap, improving both efficiency and predictability [2]. Second, a gossip-based communication protocol allows drones to share information with each other directly and reliably. This approach improves fault tolerance by ensuring that coordination can continue even if some communication links fail or drones go offline [3].

To further improve safety and reliability, the platform includes a built-in simulation module. This feature allows users to test drone behaviors and coordination strategies before real missions begin, helping to identify potential issues in advance. For missions that require high situational awareness, such as disaster response or search and rescue, the platform also offers a 3D panorama generation tool. This tool creates detailed visual reconstructions from images captured by the drones, giving operators a richer understanding of the environment [1,2].

While many existing systems offer partial solutions-often relying on centralized control or lacking pre-mission validation-our platform provides a more cohesive alternative. It combines deterministic path planning, robust communication, pre-deployment simulation, and immersive

3D visualization into a single, user-friendly framework. Building on our earlier research [1-4], this work delivers a refined and practical system designed to make autonomous swarm operations more effective, resilient, and accessible.

II. SYSTEM ARCHITECTURE OVERVIEW

Our decentralized UAV swarm management platform is built on a flexible, modular, and cross-platform client–microservice architecture, illustrated in Fig. 1. Originally introduced in [1] and extended in [2], this architecture was first developed to support mission planning and task execution within autonomous UAV swarm networks. Thanks to its layered design and modular structure, the platform has since matured into a versatile framework capable of managing and simulating distributed operations across a wide range of domains.

One of the platform's core strengths lies in its microservice-based design, which allows individual components to be developed, deployed, and scaled independently. This not only improves system reliability and maintainability but also makes it easier to adapt to the needs of specific missions or technologies. The architecture is also platform-agnostic, ensuring that it can operate smoothly across different types of UAV hardware and communication setups.

Because of this flexibility, the system is not limited to drone coordination. It can also support broader applications, including multi-robot teams, sensor-based monitoring systems, and other distributed, mission-critical environments where decentralized control and real-time responsiveness are essential.

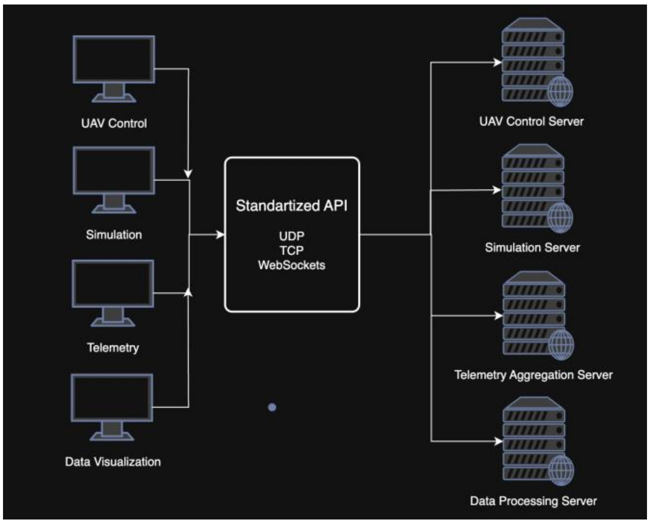


Fig. 1. Illustration of the client-microservice-based system architecture.

A. Service-Oriented Communication Architecture

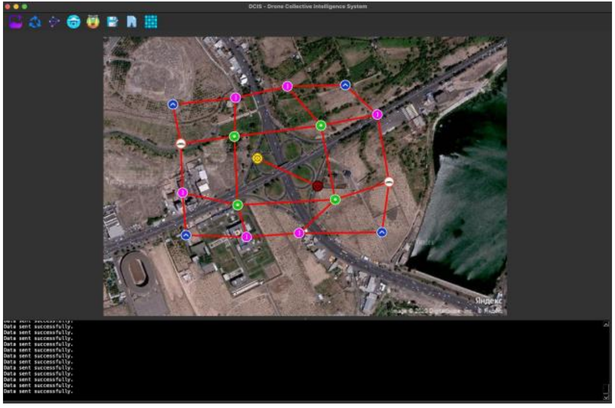
In contrast to traditional centralized control paradigms, the proposed system employs a fully distributed, service-oriented architecture in which no single server monopolizes coordination responsibilities. Instead, the client application interfaces directly with a suite of functionally distinct microservices, each exposed via a standardized API. This architectural design supports multiple network communication protocols-including TCP, UDP, and WebSockets thus ensuring high levels of interoperability across heterogeneous deployment environments [1].

Each microservice encapsulates a specific operational domain, such as UAV swarm coordination, task simulation, real-time telemetry processing, or multisource data integration, as illustrated in Fig. 1. Upon establishing a connection with a target microservice, the client application dynamically reconfigures its interface to present task-specific functionality. For example, when interfacing with the UAV Command Service, the user interface adapts to provide interactive, map-based tools for defining operational zones, constructing mission graphs, assigning mission parameters, and tagging nodes with semantic labels such as "Attacker," "Target," "Drone," or "Boundary" [2], [4], as shown in Fig. 2. This dynamic client-service interaction enhances system flexibility and supports domain-specific mission workflows.

B. Integrated Simulation and Algorithm Validation Environment

A critical component of the platform is its embedded simulation environment, which enables pre-deployment emulation and rigorous validation of swarm coordination behaviors [2]. By interfacing with the Simulation Service, users can replicate real-time UAV movements governed by deterministic rotor-router path-planning algorithms, while simultaneously observing decentralized message propagation through gossip-based broadcast protocols [3]. This simulation capability provides a high-fidelity operational sandbox in which coordination strategies can be evaluated under varied mission conditions.

The simulation environment facilitates early-stage detection of algorithmic bottlenecks, behavioral anomalies, and communication failures. As such, it significantly enhances mission preparedness and reduces the probability of systemic faults during live deployment. By enabling operators to validate autonomous swarm behavior in a controlled setting, the system fosters safer, more reliable UAV operations across mission-critical domains.



```
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
Data sent successfully.
```

Fig. 2. Mission configuration panel.

C. Modular Extensibility and System Flexibility

One of the platform's key strengths is its flexibility. Thanks to standardized API interfaces, new microservices can be added without changing the core client application. This makes the system highly adaptable. For instance, it has already supported the integration of advanced modules like the 3D panorama generation system, introduced in earlier work [2]. This system creates panoramic visualizations of the environment using images captured by the UAVs, providing operators with a richer, more immersive understanding of mission areas.

The client serves as a dynamic hub, adjusting its user interface and functionality based on the specific service it connects to. This approach ensures a clear separation between different system components, making it easier for users to interact with the platform. It also simplifies the process of adding new algorithms and features, ensuring that the platform can continuously evolve as new technologies and capabilities emerge.

D. Enhancing Algorithmic Integration and Deployment

The platform's design is particularly suited for deploying, testing, and visualizing decentralized algorithms. With its microservice architecture and dynamic client interaction, it creates an ideal environment for experimentation and refinement of coordination strategies. Past applications of the rotor-router model and gossip-based communication protocols [2] have already demonstrated the system's ability to manage complex tasks and coordinate large swarms of UAVs reliably.

In this work, we have built on these earlier foundations to introduce more advanced algorithms, pushing the system's capabilities further in terms of autonomy, scalability, and real-time responsiveness. These improvements make it easier to monitor and manage complex swarm missions, allowing UAVs to make adaptive decisions in real time while maintaining coordination in challenging, dynamic environments.

III. ALGORITHMIC FRAMEWORK

This section outlines a rotor-router-based coordination framework developed to support decentralized task management in UAV swarms. The approach integrates three key strategies:

Eliminating rotor-router cycles in parallel by deploying multiple agents. Starting agent operations from both the boundaries and interior of the task space, and representing restricted or off-limits areas through special topological markers (negatively oriented cycles).

Together, these strategies help ensure that the swarm operates in a predictable, stable pattern, while also allowing the system to respect mission constraints and distribute tasks without centralized control.

A. Parallel Elimination of Rotor-Router Cycles

To help the swarm reach a stable state more quickly, we use a parallel method that assigns different UAVs to different parts of the environment. Each drone begins operating on a separate directed loop, or "cycle," within the mission graph. Based on principles of weak reversibility [6], each cycle can be reversed by a single drone navigating it. When this happens on many cycles at once, each drone works independently, but the system as a whole still arrives at the same final state-thanks to the rotor-router algorithm's inherent deterministic (Abelian) nature [7].

As illustrated in Fig. 3, UAVs are launched at strategic points across the environment, each focusing on simplifying or resolving a specific cycle. Over time, these local operations cause the entire swarm to settle into a structured state known as a "unicycle configuration"-a single loop that spans the whole task area, combined with a tree-like structure that connects the rest of the space. This setup enables smooth and complete area coverage, while keeping communication between drones to a minimum [4,5].

B. Representing No-Fly Areas Using Negative Rotor Cycles

To account for real-world constraints like restricted airspace or zones affected by radio interference, our system introduces a way to mark these regions directly in the rotor-router framework. We do this using negatively oriented cycles-closed loops whose rotor directions are specifically arranged to prevent UAVs from entering the area.

As shown in Fig. 4, a no-fly zone is represented by a counterclockwise cycle embedded in the mission graph. The rotors around it are oriented in such a way that no drone can enter from the outside, effectively marking the zone as off-limits. Because the rotor-router model behaves in a fully deterministic way [6], these restrictions are automatically respected during the mission without the need for live intervention or centralized control.

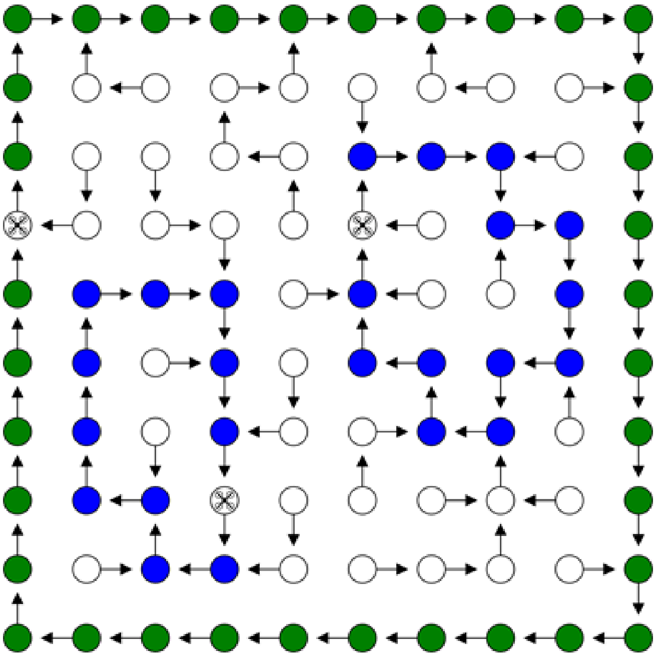


Fig. 3. Parallel dismantling of multiple rotor-router cycles by concurrent UAV agents leads to a stable unicycle configuration.

This approach allows the swarm to navigate safely and autonomously, avoiding sensitive or restricted regions without increasing communication load or requiring reconfiguration mid-mission. It offers a simple yet powerful way to ensure environmental and regulatory compliance during complex swarm operations.

C. Decentralized Initialization from Inner and Outer Cycles

Traditional rotor-router approaches usually begin by placing UAVs along the outer boundary of the mission area. While this works well for small or simple environments, it can lead to delays and uneven task coverage in larger or more complex ones. To solve this, we propose a hybrid deployment strategy where UAVs are launched both on the outer boundary and on carefully chosen internal loops within the environment.

Each of these internal cycles is designed to meet the “weak reversibility” condition [6, meaning a single drone can independently reverse and clear the loop it’s assigned to, without needing help from other agents. Meanwhile, drones on the outer boundary follow an Eulerian path [5], covering the area methodically.

By activating drones in both the outer and inner regions at the same time, the system achieves faster convergence and a more balanced distribution of work. This approach improves the swarm’s ability to scale, speeds up task execution, and makes the entire system more resilient-especially in environments with multiple compartments or mission-critical zones.

IV. INTEGRATION OF IMAGE-BASED RECONSTRUCTION AND PANORAMIC SYNTHESIS FOR UAV SWARM MISSIONS

In modern UAV swarm missions, having a clear visual understanding of the environment is just as critical as coordinated movement. To support smarter decision-making, mapping, and adaptive mission planning, our platform includes a powerful visual intelligence component: the automated creation of panoramic maps from images captured by the UAVs.

This process transforms raw aerial imagery into detailed, high-resolution overviews of the environment. These stitched panoramas provide valuable spatial context for tasks such as anomaly detection, post-mission analysis, and real-time tactical decisions. The feature becomes especially valuable in areas where GPS signals are blocked or network communication is limited-helping the swarm operate independently when external data is unavailable or delayed [8].

A. Autonomous Image Acquisition in Distributed Swarm Networks

Each UAV in the swarm is equipped with onboard cameras that capture overlapping images along their flight paths. This process is carefully synchronized with the drone’s navigation system-whether using GNSS or fallback solutions like visual-inertial odometry (VIO) [9]-to maintain spatial alignment across all images.

The drones collaborate to avoid collecting redundant data. Nearby UAVs adjust their photo-taking schedules and camera angles to ensure enough overlap without oversampling. This coordination is handled using the same gossip-based communication framework already used for task distribution.

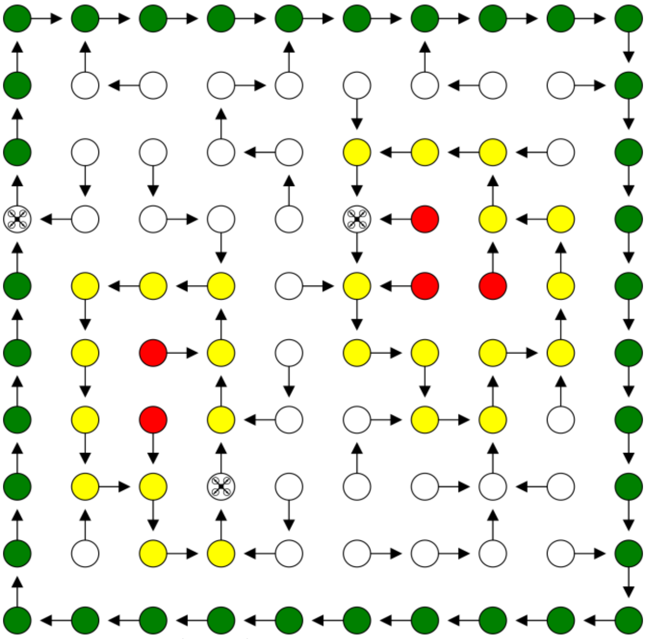


Fig. 4. Encoding of a no-fly zone (yellow cycle) using a negatively oriented cycles embedded in the rotor-router UAV field.

Images are tagged with important metadata like time, location, altitude, heading, and camera details. Depending on available connectivity and system resources, this data can either be sent to a central server for processing or stored locally for later upload. This hybrid model allows for both real-time image previews and offline processing when needed.

B. Panorama Generation Pipeline

Once the images are collected, they pass through a robust image processing pipeline to create seamless panoramic maps:

1) *Preprocessing:* Images are first cleaned-removing distortion, noise, and uneven lighting, and correcting sharpness.
2) *Feature Detection and Matching:* Keypoints are identified (using tools like ORB [10], SIFT, or SURF) and matched between overlapping images.
3) *Geometric Estimation:* The relative positions and angles of the cameras are calculated using structure-from-motion (SfM), then fine-tuned through bundle adjustment [11].
4) *Image Warping and Stitching:* The images are transformed into a unified coordinate system and blended together smoothly using multiband blending techniques [12].
5) *Export and Compression:* Final panoramas are saved in high-quality formats like GeoTIFF, with optional georeferencing for map-based use.

C. Toolchain Integration: OpenDroneMap and Kolor AutoPano Giga 4.4

To support this stitching pipeline, two toolchains have been integrated into the platform:

OpenDroneMap (ODM) is an open-source photogrammetry toolkit that implements SfM, MVS, and dense meshing to reconstruct 3D terrain and orthophotos from UAV imagery [13]. It supports CLI and API interaction, making it well-suited for automated batch processing in swarm deployments. Its support for orthorectification and DSM generation makes it particularly suitable for missions requiring geoaccurate terrain modeling. The UAV swarm images concatenation and panorama generation results in ODM provided in Fig. 5.

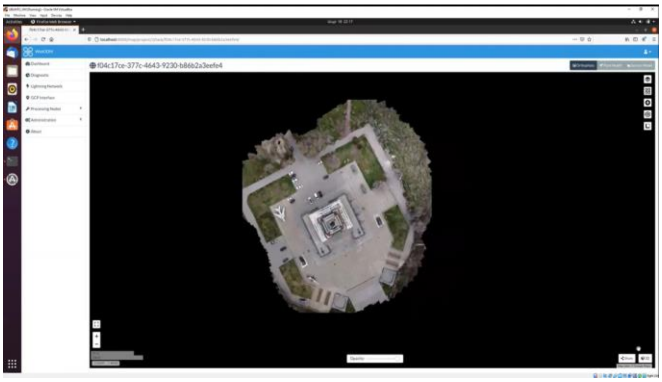


Fig. 5. Example orthomosaic stitched with OpenDroneMap showing stitched aerial coverage over structured environment with GNSS-based georeferencing.

Kolor AutoPano Giga 4.4, a commercial panoramic stitching tool, offers superior alignment and blending capabilities for unordered image sets. Known for its use in panoramic photography and 360-degree video, AutoPano Giga excels in maintaining consistent color, minimizing ghosting artifacts, and stitching images in visually complex environments such as interiors or forested zones [14]. Although it lacks full 3D reconstruction support, its simplicity and speed make it a valuable asset in real-time mapping and low-altitude missions. The UAV swarm images concatenation and panorama generation results in Kolor AutoPano Giga 4.4 provided in Fig. 6

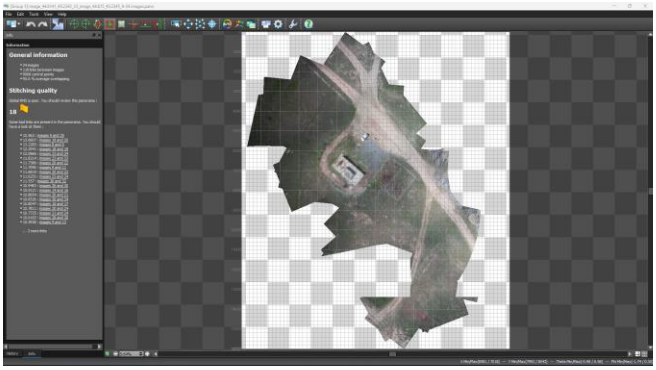


Fig. 6. Example panoramic output generated using Kolor AutoPano Giga 4.4. Image demonstrates exposure blending and minimal ghosting in urban overlap zones.

D. Integration Challenges and System Adaptation

Bringing both OpenDroneMap (ODM) and Kolor AutoPano Giga into a single UAV swarm platform was not without its difficulties. Each tool has its strengths, but also unique requirements that needed to be addressed at the system level:

- ✓ *Different Output Styles:* ODM produces georeferenced, map-accurate outputs, while AutoPano focuses on visually aligned panoramas without spatial metadata.
- ✓ *Hardware Requirements:* ODM's processes like photogrammetry and 3D reconstruction demand significant computing power, often needing GPUs or high-performance CPUs. In contrast, AutoPano works well on standard machines, offering faster results with less hardware.
- ✓ *Metadata Expectations:* ODM requires consistent GNSS metadata for accurate output, while AutoPano can function with just image content-making it useful when GPS signals are unavailable.

To accommodate these differences, the platform includes a smart middleware layer. This layer automatically chooses which tool to use based on factors like the mission’s goals, the availability of metadata, and system resources. This way, the platform stays flexible and efficient, no matter the context or constraints.

E. Comparative Evaluation and Performance Metrics

To understand how well each tool performs, we tested ODM and AutoPano on identical sets of drone images captured in urban, semi-structured, and natural environments. We evaluated the results using several practical and technical criteria:

- ✓ *Accuracy of Stitching:* We checked how well keypoints aligned across images and how seamless the final outputs looked.
- ✓ *Image Quality:* We looked for issues like ghosting, inconsistent brightness, and blurring at the seams.
- ✓ *Processing Time:* We measured how long each tool took to process the same dataset on identical hardware.
- ✓ *Scene Coverage:* We examined how many of the input images were successfully included in the final panorama.
- ✓ *Performance in Challenging Conditions:* We tested how each tool handled difficult lighting and image sets with high perspective shifts.

AutoPano excelled in quick-turnaround scenarios, particularly for indoor and low-altitude missions, where color blending and visual clarity are priorities. ODM, on the other hand, proved more effective for large-scale terrain mapping and 3D reconstruction, where spatial accuracy is critical.

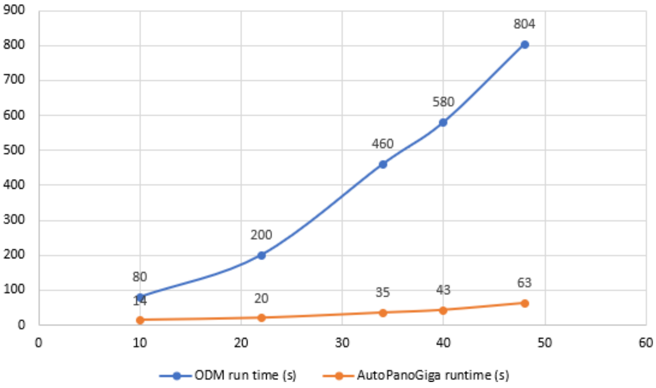


Fig. 7. ODM and AutoPano Giga performance comparison.

By integrating both tools, the platform offers the best of both worlds-high-quality visual outputs and robust geospatial modeling-adaptable to any mission scenario. Figure 7 presents a side-by-side comparison of the outcomes from both systems.

V. DISCUSSION

The proposed multi-user platform demonstrates that deterministic coordination strategies, when combined with modular software architecture, can significantly enhance the autonomy, scalability, and reliability of UAV swarm operations. The integration of the rotor-router algorithm ensures predictable and provable swarm behavior, while the gossip-based communication mechanism enables decentralized synchronization, even in the absence of a central controller. These characteristics make the system particularly resilient to node failures and communication delays, offering an edge over stochastic or cloud-dependent approaches.

The experimental integration of image-processing modules, OpenDroneMap and Kolor AutoPano Giga-further reveals the platform's adaptability. AutoPano's superior performance in real-time visual feedback, coupled with ODM’s strength in geospatial analytics, demonstrates that a dual-toolchain strategy is not only feasible but beneficial for varying mission requirements. This hybrid capability is essential for practical deployments where missions may vary from low-altitude surveillance to large-area 3D mapping.

The platform’s ability to enforce no-fly zones using negative rotor cycles illustrates an elegant use of algorithmic constructs to address real-world regulatory and safety constraints. Before live missions, the platform enables full simulation of swarm behavior, communication flow, and area coverage. Users can emulate UAV operations, inspect synchronization across agents, and test the impact of node failures. Simulation scenarios also support latency injection and message loss, validating robustness under adverse network conditions. Taken together, the findings suggest that the system is well-suited for high-stakes domains such as search-and-rescue, disaster response, and GNSS-denied military scenarios, where autonomous decision-making and situational awareness are paramount.

VI. CONCLUSION

This study presents a flexible and scalable system for coordinating UAV swarms without the need for centralized control. By using a deterministic rotor-router algorithm, the platform ensures balanced coverage across the mission space, while a lightweight gossip-based communication protocol allows drones to stay coordinated, even if some connections are lost.

Unlike many systems that depend on cloud services or random decision-making, our platform is fully decentralized and predictable. Key advantages include:

- Reliable task distribution without synchronized clocks or central orchestration;
- A modular architecture that easily integrates new tools and algorithms;
- A built-in simulator for testing and refining missions before deployment;
- The ability to define and enforce no-fly zones directly in the mission layout.

A key distinction of our approach is its grounding in true self-organization. While many platforms are labeled “self-organizing,” they often just enable local autonomy. Our system goes further-it enables complex global coordination to emerge from simple local rules, without relying on any central controller or shared global state.

VII. OPERATIONAL INSIGHTS

To enhance mission awareness, the system supports two visual processing tools. Kolor AutoPano Giga is ideal for creating fast, visually smooth panoramas during a mission, offering clear overviews in real time. On the other hand, OpenDroneMap (ODM) focuses on accuracy, delivering detailed maps and 3D models that support more analytical tasks like terrain evaluation and infrastructure assessment.

Together, these capabilities make the platform highly adaptable-whether it’s used for disaster response, remote inspections, or field mapping. It’s a practical, mission-ready solution that balances autonomy, reliability, and real-time insight in even the most challenging environments.

VIII. ACKNOWLEDGMENT

The research was supported by the Science Committee of the Republic of Armenia within the frames of the research projects 21AG-1B052 and 24DP-1B016.

REFERENCES

[1] S. Poghosyan, V. Poghosyan, A. Lazyan, A. Atashyan, D. Hayrapetyan and H. Astsatryan, “Cloud-Based Self-Organizing UAV Swarm Simulation Platform,” *Computer Science and Information Technologies Conference*, 2023.

[2] A. Atashyan, A. Lazyan, D. Hayrapetyan, H. Astsatryan,V. Poghosyan, S. Poghosyan, and Y. Shoukourian. “Mission preparation for self-organizing uav swarms on multiuser platform,” *Programming and Computer Software*, vol. 50, pp. s39-s46, 2024.

[3] V.V. Papoyan, V.S. Poghosyan and V.B. Priezzhev, “A loop reversibility and subdiffusion of the rotor-router walk,” *Journal of Physics A: Mathematical and Theoretical*, vol. 48, no. 28, p. 285203, 2015.

[4] V. Poghosyan and V. Priezzhev, “Euler tours and unicycles in the rotor-router model,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2014, no. 6, p. P06003, 2014.

[5] V. Poghosyan, S. Poghosyan, A. Lazyan, et al. "Self-Organizing Multi-User UAV Swarm Simulation Platform," Programming and Computer Software, vol. 49, no. suppl 1, pp. S7–S15, 2023. DOI: 10.1134/S0361768823090086.

[6] S. Poghosyan, V. Poghosyan, S. Abrahamyan, A. Lazyan, H. Astsatryan, Y. Alaverdyan and K. Eguiazarian, “Cloud-based mathematical models for self-organizing swarms of UAVs: design and analysis,” *Drone Systems and Applications*, vol. 12, pp. 1-12, 2024.

[7] A.E. Holroyd, L. Levine, K. Mészáros, Y. Peres, J. Propp and D.B. Wilson, “Chip-Firing and Rotor-Routing on Directed Graphs,” In: Sidoravicius, V., Vares, M.E. (eds) In and Out of Equilibrium 2. Progress in Probability, Birkhäuser Basel, vol. 60, pp. 331-364, 2008. DOI: 10.1007/978-3-7643-8786-0_17.

[8] R. T. Collins, A. J. Lipton, H. Fujiyoshi and T. Kanade, “Algorithms for cooperative multisensor surveillance,” *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1456-1477, 2001. DOI: 10.1109/5.959341.

[9] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart and P. Furgale, “Keyframe-based visual–inertial odometry using non-linear optimization,” *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314-334, 2014.

[10] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” *International Conference on Computer Vision*, pp. 2564-2571, 2011.

[11] B. Triggs, P.F. McLauchlan, R.I. Hartley and A.W. Fitzgibbon, “Bundle adjustment - A modern synthesis,” *Vision Algorithms: Theory and Practice*, pp. 298–372. Springer, 2000.

[12] M. Brown and D.G. Lowe, “Automatic panoramic image stitching using invariant features,” *International Journal of Computer Vision*, vol. 74, pp. 59–73, 2007.

[13] OpenDroneMap Contributors. Opendronemap: The open-source photogrammetry toolkit. https://www.opendronemap.org. Accessed: 2025-05-29.

[14] Kolor SARL. Kolor autopano Giga 4.4. https://www.kolor.com/autopano/. Accessed: 2025-05-29.