Journal of Engineering Science *Vol. XXXIII, no. 2 (2026), pp. 20 - 31*
Fascicle *Industrial Engineering* *ISSN 2587-3474*
Topic *Applied Engineering Sciences and Management* *eISSN 2587-3482*

https://doi.org/10.52326/jes.utm.2026.33(2).02
UDC 629.7.05:681.5:004.8

ANALYSIS OF CONTROL ARCHITECTURES FOR UNMANNED AERIAL VEHICLES

Adriana Ursu*, ORCID: 0009-0003-4263-2444,
Igor Calmîcov, ORCID: 0000-0002-4017-3702,
Viorel Cărbune, ORCID: 0000-0002-1556-4453

Technical University of Moldova, Studentilor 9/7, Chișinău, 2045, Republic of Moldova
* Corresponding author: Adriana Ursu, *ursu.adriana@iis.utm.md*

Received: 05.14.2026
Accepted: 06.17.2026

Abstract. Unmanned aerial vehicles (UAVs) have become important components of modern systems for monitoring, inspection, mapping, and autonomous intervention. Their performance directly depends on the efficiency of the control systems and the architecture used for data processing, decision-making, and mission coordination. The paper presents a comparative analysis of the main control architectures used in UAV platforms: control with on-board processing (On-Board Control), hierarchical Master-Swarm architecture, and centralized systems based on a Ground Control System (GCS). It describes the operating principles of each architecture, the hardware and software components involved, advantages, limitations, and specific areas of application. The analysis highlights the impact of each solution on autonomy, scalability, resilience to communication loss, and real-time control performance. The results obtained show that On-Board Control systems offer superior autonomy and robustness, Master-Swarm architectures are intended particularly for the collaborative coordination of drone swarms, and GCS systems provide advanced capabilities for monitoring and centralized processing. Furthermore, current trends oriented towards hybrid architectures are highlighted, which combine the advantages of the three models to increase the level of autonomy, safety, and operational efficiency of modern UAV systems.

Keywords: *UAV, autonomous drone, control system, On-Board Control, Drone Swarm, Ground Control System, real-time control, autonomous navigation, drone swarm, operational autonomy, multi-agent systems, distributed control.*

Rezumat. Vehiculele aeriene fără pilot (UAV) au devenit componente importante ale sistemelor moderne de monitorizare, inspecție, cartografiere și intervenție autonomă. Performanța lor depinde direct de eficiența sistemelor de control și de arhitectura utilizată pentru procesarea datelor, luarea deciziilor și coordonarea misiunilor. Lucrarea prezintă o analiză comparativă a principalelor arhitecturi de control utilizate în platformele UAV: control cu procesare la bord (On-Board Control), arhitectura ierarhică Master-Swarm și sistemele centralizate bazate pe un Sistem de Control la Sol (GCS). Se descriu principiile de funcționare ale fiecărei arhitecturi, componentele hardware și software implicate, avantajele, limitele și domeniile specifice de aplicare. Analiza evidențiază impactul fiecărei soluții asupra autonomiei, scalabilității, rezistenței la pierderea comunicațiilor și performanței controlului în timp real. Rezultatele obținute arată că sistemele de Control la Bord oferă autonomie și robustețe superioare, arhitecturile Master-Swarm sunt destinate în special coordonării colaborative a roiurilor de drone, iar sistemele GCS oferă capabilități avansate de monitorizare și procesare centralizată. În plus, sunt evidențiate tendințele actuale orientate spre arhitecturi hibride, care combină avantajele celor trei modele pentru a crește nivelul de autonomie, siguranță și eficiență operațională a sistemelor UAV moderne.

Cuvinte cheie: *UAV, dronă autonomă, sistem de control, control la bord, roi de drone, sistem de control la sol, control în timp real, navigație autonomă, roi de drone, autonomie operațională, sisteme multi-agent, control distribuit.*

1. Introduction

Unmanned aerial vehicles (UAV), generically known as drones, represent aerial platforms capable of operating without a human crew on board, being controlled remotely or functioning autonomously through complex algorithms or artificial intelligence systems. Initially developed predominantly for entertainment, UAV technologies have experienced an exponential expansion in recent decades, particularly in the agricultural, commercial, research, and military sectors. Due to their versatility, low operational costs, and the ability to explore hostile or hard-to-reach environments without risking human lives, drones have become important tools in a wide range of applications. Currently, they are widely used in fields such as precision agriculture, critical infrastructure monitoring, emergency management, mapping and three-dimensional terrain modeling, environmental surveillance, and logistics [1].

Despite significant progress in the development of UAV platforms, designing control systems remains one of the most complex engineering challenges. The nonlinear character of flight dynamics, variations in aerodynamic parameters, uncertainties in mathematical models, and the influence of external disturbances require the development of robust and adaptive control strategies. Moreover, the integration of an increasing number of sensors, communication systems, and data processing algorithms leads to higher demands regarding the computational performance and reliability of on-board systems [2].

Current trends in UAV design aim to increase the degree of autonomy through the use of advanced control techniques and artificial intelligence algorithms. Thus, classical methods based on PID (Proportional-Integral-Derivative) control are supplemented or replaced by modern strategies, such as predictive control (MPC – Model Predictive Control), adaptive control, robust control, and machine learning methods [4, 5]. These approaches allow for improving navigation precision, optimizing energy consumption, and increasing the capacity to operate in complex and unstructured environments. They constitute, in essence, technological ecosystems of remarkable complexity, representing the convergence point of aerospace engineering, advanced electronics, systems theory, and applied computer science. Their fundamental purpose is to guarantee stable flight dynamics, high-precision navigation, and the autonomous or semi-autonomous execution of complex tasks, operating in dynamic and often unpredictable three-dimensional environments (subject to aerodynamic and meteorological disturbances).

In the context of modern applications, which include territory monitoring, mapping, precision agriculture, search and rescue operations, or autonomous deliveries, the requirements imposed on control systems are becoming increasingly complex. Depending on the operational requirements and the required degree of autonomy, the architecture of control systems can be implemented in different configurations. Thus, processing and decision-making can be performed directly on board the drone (On-Board Control) [4], can be coordinated via a main drone within multi-UAV systems (Master Drone) [7], or can be centrally managed by a Ground Control System (GCS) [8]. Each of these approaches presents specific advantages and limitations regarding communication latency, processing capacity, scalability, and the level of system autonomy. Analyzing these control architectures is very important for understanding how modern drones operate and for identifying optimal solutions based on the specific characteristics of each mission.

The problem addressed in this paper consists of the analysis and evaluation of control systems used for UAV platforms, focusing on the control mode, command architectures, and modern autonomous navigation methods [6]. The increasing complexity of missions performed by unmanned aerial vehicles requires the development of control systems capable of ensuring the safe, precise, and efficient operation of drones in dynamic and unpredictable environments. Traditional control systems, based exclusively on human operator intervention, are limited by communication latency, reduced real-time reaction capacity, and the difficulty of simultaneously managing a large number of aerial platforms. In this context, there is a need for advanced control architectures that allow for an increased degree of autonomy and adaptability for drones.

The main challenge lies in designing and integrating control architectures that ensure an optimal balance between system autonomy, communication reliability, processing capacity, and operational efficiency. The choice of how to organize local control on board the drone, coordinated via a main drone, or centrally managed via a ground control station directly influences the system's performance. Furthermore, the challenge is amplified by the need to maintain flight stability, synchronization among components, and resilience in the face of communication errors or unfavorable environmental conditions.

The objective of this paper is to analyze and compare the main control architectures used in UAV systems, with an emphasis on On-Board Control, hierarchical coordination through a Master Drone, and centralized management through a Ground Control System (GCS) [3, 6]. The paper aims to highlight the operating principles, advantages, and limitations of each approach, as well as to identify applicability domains based on operational requirements and the required level of autonomy. Through this analysis, the goal is to formulate conclusions regarding the efficiency and appropriateness of using different control models within modern drone systems.

2. Analysis of Control Architectures for UAV Platforms

The control system architecture represents one of the fundamental elements that determine the performance, degree of autonomy, and reliability of a UAV platform. The evolution of communication technologies, embedded systems, and artificial intelligence algorithms has led to the emergence of several models for organizing control, each optimized for specific operational scenarios. Among the most widely used architectures are On-Board Control, hierarchical coordination via a main drone (Master Drone Control), and centralized management through a Ground Control System (GCS).

2.1. On-Board Processing Control Architecture

The On-Board Control architecture is where all essential functions for flight control are executed by computing systems installed directly on the drone. This approach involves the integration of a flight control unit (Flight Controller), an on-board computer (Companion

Computer), and a set of sensors that provide information about the UAV's status and surrounding environment. Data from sensors are continuously analyzed to estimate the drone's position, speed, and orientation. Based on this information, control algorithms generate commands for the motors and control surfaces [4].

The diagram in Figure 1 presents the architecture of a closed-loop cascaded control system used for flight control of a drone or autonomous aircraft. Information flows from top to bottom for command execution, and measured data returns through a feedback loop to correct errors. The entire cycle repeats very rapidly (thousands of times per second) to maintain flight stability [5].

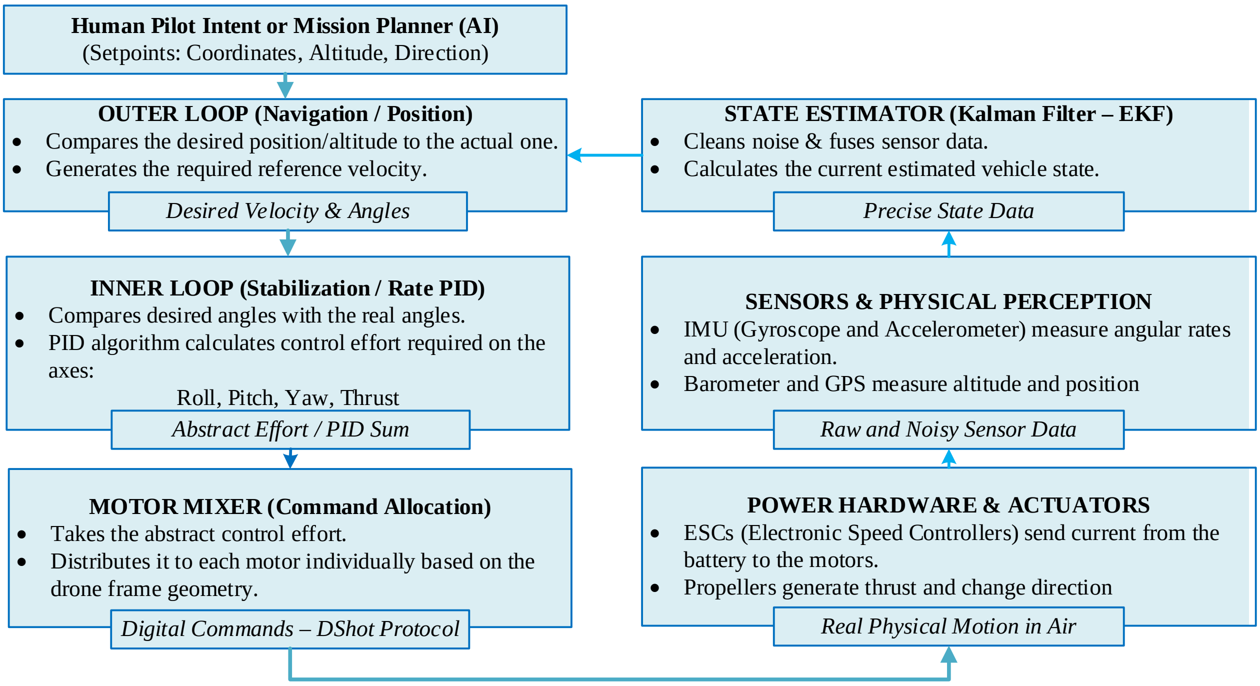

Figure 1. Architecture of an on-board processing control system for a drone.

1. The process begins with the actions of a human pilot or a mission planning application. This establishes the desired flight parameters, called setpoints, such as coordinates, altitude, and direction of travel. These values represent the objective that the control system must achieve.
2. The outer loop is responsible for navigation and positioning. It compares the desired position and altitude with the real position and altitude of the drone.
3. Based on this difference (control error), the system calculates the reference speed and angles needed to reach the desired position.
4. The output of the outer loop is transmitted to the inner loop, which handles the stabilization of the aircraft. This loop compares the desired angles with the actual angles of the drone. It uses PID (proportional-integral-derivative) control algorithms to calculate the required control action. This control is applied to the primary axes of motion, resulting in an abstract control action for each axis. This control action cannot be applied directly to the motors. Therefore, a motor mixer intervenes, transforming the abstract control commands and distributing them individually to each motor, taking into account the drone's frame geometry. Commands are sent as digital signals. Digital commands reach the ESCs (Electronic Speed Controllers), which control the current supplied by the battery to the motors and regulate the motors' rotation speed

[4, 5]. The motors rotate the propellers, which generate the necessary thrust for the drone to move through the air. In this stage, the actual physical motion of the drone occurs.

5. To understand the true state of the drone, the system uses multiple sensors such as a gyroscope, accelerometer, and barometer. Sensor data is raw and contains noise.
6. The sensory data is processed by the state estimator. It filters out noise from the data, combines information from different sensors, and calculates the drone's estimated state. The filtered data is transmitted back to the outer and inner control loops. The controller can compare the desired state with the real state, correct errors, and generate new commands. This process forms a closed control loop, which repeats continuously, usually between 1000 and 8000 cycles per second for drone control [4, 5]. Through this rapid updating, errors are corrected immediately, aircraft stability is maintained, and the vehicle can precisely execute navigation commands.

The on-board processing control architecture has several advantages. It does not need a continuous connection with a ground station for navigation, the drone is highly resistant to jamming attacks or accidental signal loss, decision-making time is minimal, and it is the only viable architecture for environments where GPS or radio signals are inaccessible. But it also has disadvantages, such as weight and energy consumption limiting available computing power; integrating high-performance processors and advanced sensors increases platform costs; and if the platform crashes and is destroyed, data processed and stored locally may be lost, not being transmitted in real-time to a secure database.

Due to its independence from external communication networks, this architecture dominates in applications deployed in isolated or heavily shielded environments. Areas of application for On-Board Control include autonomous navigation inside mines, sewer networks, or complex logistics warehouses, where a GPS signal is non-existent and maintaining a Data Link with the exterior is impossible. The drone relies entirely on local data processing and inertial positioning methods. Use in last-mile urban delivery scenarios, where zero latency in dynamic obstacle avoidance decisions (Sense and Avoid) is critical for safety. Space platforms sent to other planets, where the enormous latency of interplanetary communications makes real-time remote control impossible.

2.2. Master-Swarm Hierarchical Control Architecture

For missions requiring vast spatial coverage or operational redundancy, coordinated multi-agent systems are used. The Master-Swarm architecture reduces the individual complexity of drones by centralizing tactical intelligence at the level of a single aerial node. The Main Drone (Master) functions as a mobile aerial server. It has superior computing power and a more complex sensor suite. It handles generating global trajectories, obstacle avoidance, and dynamic target allocation.

The Subordinate Fleet (Swarm) are the component drones, which operate exclusively at the hardware level with simplified control loops. Their main task is maintaining a relative position compared to the Master drone or tracking velocity vectors transmitted by it. Communication is achieved via an ad-hoc network (FANET - Flying Ad-Hoc Network) [9]. The Master drone issues setpoint commands, and the local control loops of the swarm drones execute necessary corrections. This topology prevents radio frequency spectrum congestion and significantly reduces the fleet's production cost.

Figure 2 presents the architecture of a drone swarm system (master–swarm) used for coordination and autonomous execution of distributed missions. The system is controlled by the Ground Control System (GCS), which transmits mission objectives and monitors overall operation status through a long-range communication link. At the center of the architecture is the Master drone, responsible for mission planning, dynamic task allocation, global route calculation, and swarm health monitoring. It processes data received from drones in the network and transmits global commands to them.

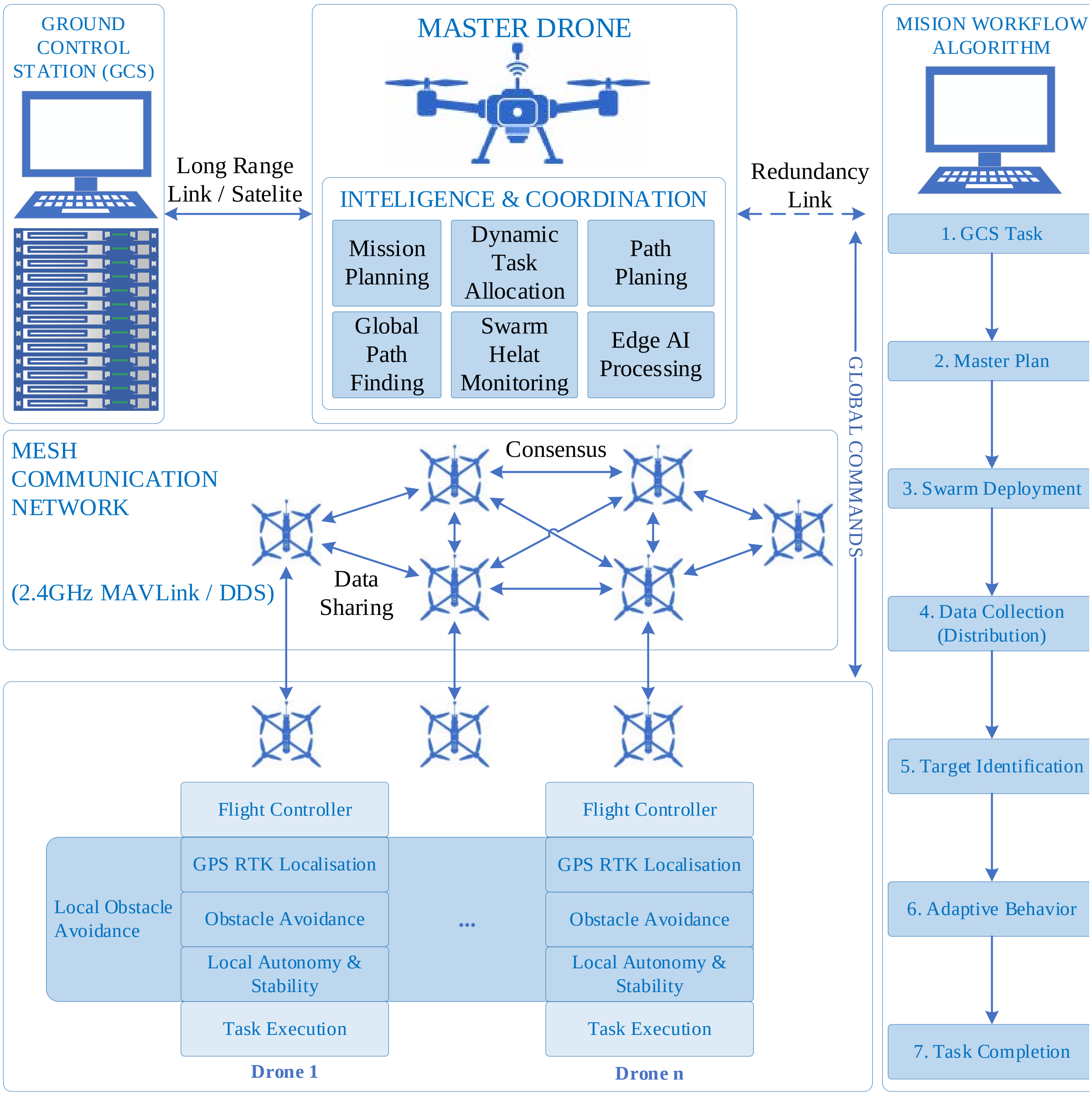

Figure 2. Architecture of a master-swarm drone system.

Drones in the swarm communicate with each other through a communication mesh network based on protocols such as MAVLink (Micro Air Vehicle Link) or DDS (Data Distribution Service) [7], which allows data sharing, routing through intermediate nodes, and achieving consensus between units. Each drone features a flight controller, RTK GPS localization system, obstacle avoidance sensors, and local autonomy modules, allowing it to execute tasks independently. The mission workflow includes stages such as task planning, swarm deployment, distributed data collection, target identification, and adaptive behavior depending on the situation [2]. This architecture allows for distributed coordination, scalability, and operational robustness in complex autonomous applications [1].

The Hierarchical Master-Swarm architecture, because it distributes tasks between an aerial command center and multiple executors, optimizing spatial coverage, has several advantages, such as:

- simultaneous monitoring of extensive areas;
- adding new drones to the swarm can be done with minimal software reconfiguration of the network topology;
- subordinate drones can be extremely simple and cheap, possessing only the hardware strictly necessary for flying and maintaining formation;
- the loss or failure of a drone in the swarm does not compromise the overall objective. The network automatically reconfigures to compensate for the missing node;

However, there are also disadvantages, such as:

- if the Master drone is destroyed or loses power, the entire swarm becomes inoperable or enters fail-safe mode, being unable to complete the mission;
- the system has high computational and algorithmic complexity;
- communication between drones is sensitive to radio spectrum congestion and requires considerable bandwidth over an area.

The ability to cover vast areas through efficient load balancing makes this architecture preferable for large-scale monitoring systems. Using swarms for simultaneous multispectral mapping of thousands of hectares of agricultural land. While the main drone creates a global map of water deficits or pest infections, the swarm drones perform parallel tasks, spraying pesticides only in affected areas. It is used for the rapid scanning of vast mountainous or maritime areas. Subordinate drones can scan the given area, and upon detecting a thermal anomaly (such as a lost person), the main drone would reorganize the swarm topology to focus monitoring efforts on that point. Another scenario could be analyzing the spread of forest fires or toxic chemical spills in real-time, minimizing human impact and reducing operating costs (losing a drone in a swarm does not stop the mission).

2.3 Ground Control Architecture

The Ground Control System (GCS) is a system where the control component is located on the ground, and the drone acts as a strict executor of commands received remotely. Communications management (Data Link) takes place through a bidirectional radio link, consisting of an uplink channel (navigation commands, parameter adjustments) and a downlink channel (detailed telemetry, real-time video stream). The human operator or the software installed on the GCS defines the route via waypoints, leaving the Flight Controller to manage exclusively the stabilization on the determined trajectory. An inherent vulnerability of architectures based on ground control stations (GCS) is the loss of the data link. For this reason, the architecture must include algorithms (safety protocols) to automatically initiate a return-to-home function. Upon detecting signal loss for a critical period (approximately 3 seconds), the drone automatically switches to strict On-Board mode to return to the launch point based on GPS coordinates stored at takeoff [8].

The diagram in Figure 3 presents the complete control cycle of a UAV, highlighting the information flow between the ground control station and the aerial vehicle, as well as the role of the drone's internal components. The system operates through two main types of communication links: control link (uplink) and telemetry and video (downlink). Initially, the operator uses a signal transmitter from the ground control station (GCS) to input flight control commands. These include parameters such as pitch, roll, yaw, throttle, or flight modes. Commands are transmitted to the drone via uplink.

The signal is received by the transceiver on board the drone, which transmits the data to the Flight Controller (FC), the central component responsible for stabilizing and coordinating all systems. The flight controller processes the received commands and uses information from the drone's sensors, such as the IMU (gyroscope and accelerometer), GPS, and barometer, to determine the vehicle's orientation and position [10]. Simultaneously, the drone transmits telemetry data to the control station via the MAVLink protocol, including information on battery level, GPS position, altitude, and drone orientation. Also, a video camera with a VTX (Video Transmitter) broadcasts the real-time video stream. This information forms the downlink, through which the operator receives visual and technical feedback about the drone's status [8]. Thus, a continuous cycle of control and monitoring is created, where commands are sent from the ground to the drone, and status and video data are returned to the operator for adjusting piloting and supervising the mission.

In the ground control system, the drone acts as a sensory relay and maneuver executor, but decision-making is outsourced to terrestrial servers or human operators. This allows running massive Machine Learning models on the Ground Control System or performing real-time photogrammetric processing, which is not possible to integrate onto a mobile platform [10]. This is essential in critical or emergency situations where final decision-making requires human judgment and adherence to strict flight safety regulations.

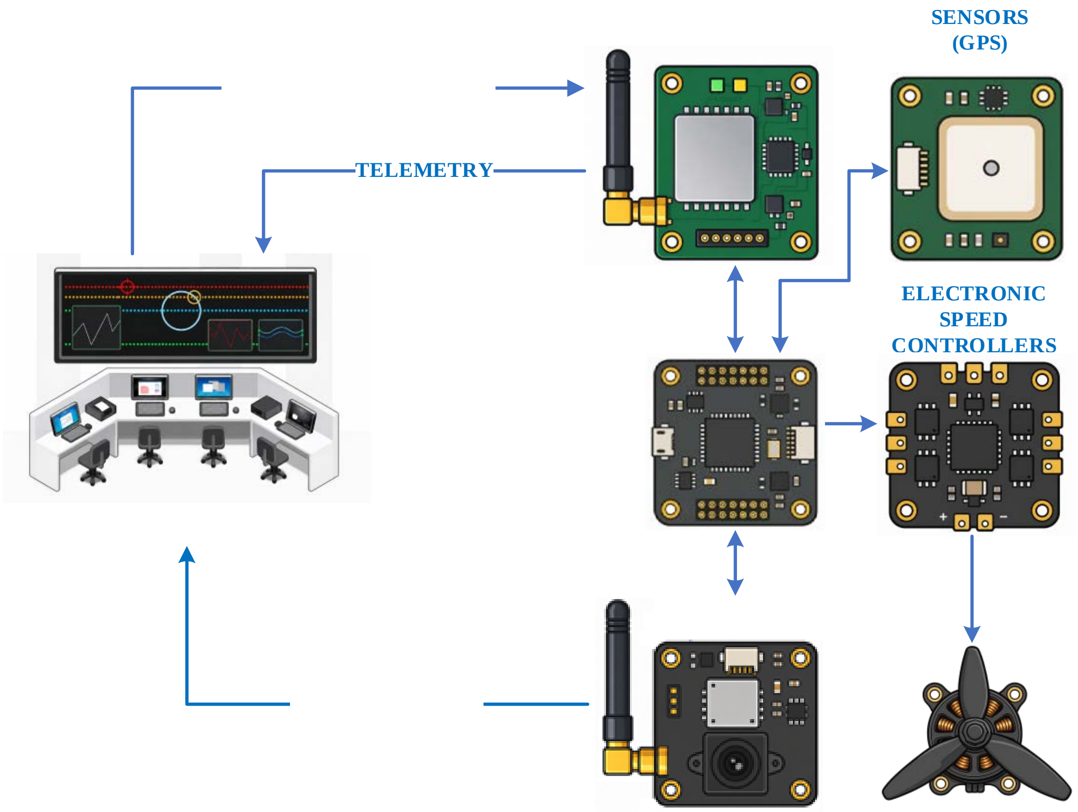

Figure 3. The control cycle between a drone and the ground station.

Shifting the computing load to the ground allows the drone to be physically lighter, and its computational and electronic characteristics to be simplified, resulting in enhanced maneuverability and longer flight times. However, this architecture also has several disadvantages. Any communication interruption, atmospheric interference, topographic shielding, or cyberattack on the communication channel can have a critical negative impact on the drone. The range is strictly dictated by the performance of ground antennas, except when using satellite communication systems, which introduce increased costs. The time required for data to be transmitted to the ground, processed, and a return command sent to the drone can cause instability in situations that require prompt actions.

When decision-making and big data analysis capabilities are more important than the mobility offered by remote UAV piloting architecture, the GCS architecture becomes the appropriate standard. This is used for monitoring oil and gas pipelines or high-voltage networks. Platforms navigate based on pre-established waypoints, transmitting terabytes of data (high-resolution photos, thermography) directly to terrestrial servers for immediate structural defect analysis. Another usage scenario for this architecture is high-precision terrain mapping with complex flight paths that require precise sensor calibration using the ground station control interface.

3. Comparative Analysis of UAV Control Architecture Performance

The comparative analysis of On-Board Control, Master Drone, and Ground Control System (GCS) architectures highlights that each control model is designed to meet different operational requirements and presents specific advantages and limitations, as seen in Table 1 [6]. From an autonomy perspective, the architecture based on on-board processing offers the best results. Because all critical decisions are made locally, the drone can continue the mission without relying on external communications. Master Drone type systems also offer a high level of autonomy, but their functioning depends on coordination between participating drones. Conversely, architectures based on a Ground Control System present a medium level of autonomy, as a significant part of the decision-making process is transferred to the ground control station.

Table 1

Comparative analysis of performances

Criterion	On-Board Control	Master Drone	GCS
Autonomy	Very high	High	Medium
Scalability	Low	High	Very high
Communication necessity	Minimal	Medium	Very high
Resilience to link loss	Very high	Medium	Low
Available computing power	Limited	Medium	Very high
Infrastructure cost	Medium	High	Very high
Real-time control	Excellent	Very good	Good

Regarding scalability, centralized systems and those based on hierarchical coordination are superior to local control. A single drone with on-board processing can efficiently execute individual missions, but simultaneously managing a large number of UAVs becomes difficult. The Master Drone architecture allows efficient coordination of drone groups by distributing tasks and organizing activities in a hierarchical structure. However, the highest scalability is offered by GCS systems, which can simultaneously monitor and control vast fleets of UAVs using high-performance IT infrastructures [3].

Analyzing communication dependency, it is observed that the On-Board Control architecture is the least dependent on the existence of a permanent communication link. This aspect represents a major advantage in applications deployed in isolated areas or in environments with electromagnetic interference. For the Master Drone architecture, communications are necessary for the continuous exchange of information between the leader and subordinate drones. GCS systems are the most dependent on communications, as the monitoring and command process relies on continuous data transfer between aerial platforms and the control center [3].

A criterion closely related to communications is resilience to communication link loss. Due to local processing, On-Board Control architectures are the most resistant to communication interruption. They can continue the mission or execute automatic fail-safe procedures without external support. In the Master Drone architecture, losing communication with the master drone can affect the coordination of the entire group, although modern redundancy mechanisms can mitigate this risk. GCS systems are the most vulnerable, as losing connection with the control station can significantly limit mission management capacity [6].

Regarding available computing power, GCS systems hold a dominant position. They benefit from high-performance servers and workstations that allow the implementation of complex artificial intelligence, video analysis, and mission optimization algorithms. Master Drone architectures offer intermediate computing capacity, as part of the processing is handled by the leader drone. For on-board control, computing power is limited by the dimensions, weight, and energy consumption of the equipment installed on the aerial platform [6].

Analysis of infrastructure costs indicates that GCS systems require the largest investments. Their implementation involves developing a complex infrastructure of communications, servers, management software, and operating stations. Master Drone-based architectures also involve high costs due to the complexity of coordination and redundancy mechanisms. On-board control presents moderate costs, as most functions are integrated directly into the UAV [3].

In terms of real-time control, the On-Board Control architecture provides the best performance. Local processing eliminates delays associated with data transmission and allows rapid reactions to environmental changes. Master Drone systems also offer very good performance, but response time can be influenced by information exchange between drones. In the case of GCS systems, performance depends on the quality and latency of communication channels.

Overall, the analysis highlights that the On-Board Control architecture is optimal for missions requiring high autonomy, rapid reaction, and independence from communications. The Master Drone architecture is best suited for the collaborative coordination of multiple UAVs and implementing smart swarm concepts. The Ground Control System architecture offers the best results in situations requiring centralized surveillance, simultaneous management of a large number of platforms, and the use of advanced computational resources. In practice, the current trend leans towards hybrid architectures that combine the advantages of all three models to achieve a superior level of autonomy, flexibility, and operational efficiency.

5. Conclusions

The analysis of control architectures for UAV platforms highlights that selecting a control solution depends directly on the mission's operational requirements, the required degree of autonomy, and available resources at the communication and processing infrastructure level. The architecture based on on-board processing (On-Board Control) offers the best performance in terms of autonomy, resilience to communication loss, and real-time control, being recommended for missions carried out in isolated environments or characterized by difficult communication conditions. The hierarchical Master-Swarm architecture allows the efficient coordination of a large number of drones and provides a high level of scalability, making it suitable for applications requiring distributed cooperation, monitoring over extensive areas, and simultaneous execution of multiple tasks. However, its functioning depends on maintaining the main node's integrity and the existence of efficient coordination and redundancy mechanisms. Systems based on a Ground Control System (GCS) have the most vastest computing resources and allow centralized monitoring of UAV operations, being appropriate for applications involving large data volumes, advanced processing, and in specific cases, can minimize or even completely reduce the need for human intervention in the decision-making process. The primary limitation of this approach is its high dependence on communication infrastructure and vulnerability to data link loss.

The comparative analysis performed demonstrates that there is no universal architecture for all use scenarios. The current trend in UAV system development is geared towards hybrid architectures integrating local processing, distributed coordination, and centralized management. This approach allows exploiting each model's advantages and contributes to increasing autonomy, operational safety, flexibility, and energy efficiency. Future research directions include the integration of distributed artificial intelligence technologies, use of 5G/6G communications and FANET (Flying Ad-Hoc Network) networks, developing adaptive mechanisms for switching between different control modes, and implementing collaborative swarm-type systems capable of operating autonomously in complex and dynamic environments [9].

Acknowledgments: This article was funded by the project: "Research on intelligent methods and effective means for monitoring forest, perennial and aquatic areas with the application of drones", 25.80012.5007.71SE at the stage of analysis of the field of study.

During the preparation of this article, the author Viorel Cărbune used Gemini3.1 Pro for translation into English. The authors reviewed and revised the result and assume full responsibility for the content of this publication.

Contribution of authors

Adriana Ursu: Conceptualization, Methodology, Investigation, Writing – original draft.
Igor Calmîcov: Investigation, Writing – editing.
Viorel Cărbune: Supervision, Investigation, Writing – editing.

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

References

1. Lizzio, F.F.; Capello, E.; Guglieri, G. (2022). 'A Review of Consensus-Based Multi-Agent UAV Implementations', *Journal of Intelligent & Robotic Systems*, 106, Article 43. DOI: 10.1007/s10846-022-01743-9;
2. Alenezi, M.; et al. (2024). 'Advancement Challenges in UAV Swarm Formation Control: A Comprehensive Review', *Drones*, 8(7), Article 320. DOI: 10.3390/drones8070320;

3. Cetinsaya, B.; Reiners, D.; Cruz-Neira, C. (2024). 'From PID to Swarms: A Decade of Advancements in Drone Control and Path Planning – A Systematic Review (2013–2023)', *Swarm and Evolutionary Computation*, 89, Article 101626. DOI: 10.1016/j.swevo.2024.101626;
4. Abbas, N.; Abbas, Z.; Zafar, S.; et al. (2024). 'Survey of Advanced Nonlinear Control Strategies for UAVs: Integration of Sensors and Hybrid Techniques', *Sensors*, 24(11), Article 3286. DOI: 10.3390/s24113286;
5. Wang, G.; Gu, C.; Li, J.; et al. (2023). 'Heterogeneous Flight Management System (FMS) Design for Unmanned Aerial Vehicles: Current Stages, Challenges, and Opportunities', *Drones*, 7(6), Article 380. DOI: 10.3390/drones7060380;
6. Zhang, Y.; Wang, X.; Li, H.; et al. (2024). 'Cooperative Motion Planning and Control for Aerial-Ground Autonomous Systems: Methods and Applications', *Progress in Aerospace Sciences*, 146, Article 101005. DOI: 10.1016/j.paerosci.2024.101005;
7. Xu, Z.; Liu, Y.; Wang, S.; et al. (2022). 'Algorithms and Applications of Intelligent Swarm Cooperative Control: A Comprehensive Survey', *Progress in Aerospace Sciences*, 135, Article 100869. DOI: 10.1016/j.paerosci.2022.100869;
8. Javaid, S.; Saeed, N.; Qadir, Z.; et al. (2023). 'Communication and Control in Collaborative UAVs: Recent Advances and Future Trends', *IEEE Transactions on Intelligent Transportation Systems*, 24, 5719–5739. DOI: 10.1109/TITS.2023.3248841;
9. Bertizzolo, L.; D'Oro, S.; Ferranti, L.; et al. (2020). 'SwarmControl: An Automated Distributed Control Framework for Self-Optimizing Drone Networks', *arXiv Preprint*, arXiv:2005.09781. DOI: 10.48550/arXiv.2005.09781;
10. Khan, M.A.; Kumar, N.; Mohsan, S.A.H.; et al. (2022). 'Swarm of UAVs for Network Management in 6G: A Technical Review', *arXiv Preprint*, arXiv:2210.03234. DOI: 10.48550/arXiv.2210.03234.

Citation: Ursu, A., Calmîcov, I., Cărbune, V. (2026). Analysis of control architectures for unmanned aerial vehicles. *Journal of Engineering Science*. 2026, 33 (2), pp. 20-31. https://doi.org/10.52326/jes.utm.2026.33(2).02.

Submission of manuscripts: jes@meridian.utm.md