Resilient UAV Swarms

Subjects: Engineering, Aerospace

Contributor: Abhishek Phadke , F. Antonio Medrano

UAVs have rapidly become prevalent in applications related to surveillance, military operations, and disaster relief. Their low cost, operational flexibility, and unmanned capabilities make them ideal for accomplishing tasks in areas deemed dangerous for humans to enter. They can also accomplish previous high-cost and labor-intensive tasks, such as land surveying, in a faster and cheaper manner. Researchers studying UAV applications have realized that a swarm of UAVs working collaboratively on tasks can achieve better results. The dynamic work environment of UAVs makes controlling the vehicles a challenge.

UAV swarm resiliency multi agent system

1. Introduction

Deploying multiple agents as part of a larger swarm has its advantages. Cooperative actions by several robots are a wide application domain [1]. Several possible advantages can be visualized particularly in the case of unmanned aerial vehicles (UAVs). A swarm of UAVs can search an area quicker than a single UAV making multiple passes over the same area. Higher-level approaches, such as search grid decomposition for individual agents, are more easily accomplished when multiple agents exist. Smaller size UAVs carry limited equipment to reduce equipment power consumption and reduce overall aircraft weight. It is possible to equip different agents in a swarm with different sensors. The result will be richer data streams that will be generated once the different sensor data is combined. Similar experiments can be envisioned where a swarm of UAV agents work at different altitudes in order to survey ground subjects, thereby providing multiple perspectives on the target. Such improvements in results by swarm agents are particularly useful considering the highly dynamic environments in which UAVs operate. Situations on the battlefield may already have changed by the time a single UAV makes a pass over the area and then moves on to cover other areas, and then returns. Similar effects are noticed while measuring large-scale phenomena such as red tide growth [2] or fish shoals [3]. Sensitive incidents such as a search-and-rescue (SAR) mission may require multiple agents to be deployed. An area may be too large for a single UAV to cover, and more agents improve the probability that a victim can be found quickly.

A multi-vehicle system can be described as effective, efficient, flexible, and exhibit higher tolerance to faults than a single agent [4]. This makes it more viable to have a swarm of UAVs attempt a particular task. However, the challenging environment they work in makes creating resilient UAV swarms a challenge. Successful UAV swarm implementations have demonstrated exceptional ability in performing tasks in various fields such as agriculture [5], natural resource surveying of water, soil, wildlife [6], search-and-rescue operations [7][8], and the military [9].

Unanticipated events such as inclement weather, intrusion from enemy agents, collision with foreign bodies or other swarm agents, loss of communication, or bugs in controlling schemes and software are just some of the events that may impede swarm function. Oftentimes, current multi-agent systems are interdependent to a high degree, making the loss of even a single agent disastrous for the swarm as a whole and its mission progress. However, failures can come in many different forms, both internal and external. Communication, Navigation, and Surveillance (CNS) failures [10] are categorized as internal failures, while weather and obstacles are external events. This entry is part of an ongoing effort to improve resiliency in UAV swarms. To implement resiliency in swarms, we first need to conceptualize it into behavior responses that can then be implemented. Most modern systems exist and work in dynamic environments that are unpredictable in terms of their properties, composition, or behavior. Moreover, they have dependencies on input streams, power sources, and networks. Woods D.D in [11] perfectly condenses resilient behavior into four concepts.

- Resilience as a rebound
- Resilience as robustness
- Resilience as graceful extensibility
- Resilience as sustained adaptability

2. Resilient UAV Swarm Components and Modules

2.1. Communication

The main modules of the communication component that need to be addressed are connectivity, network coverage, structure, and characteristics. Each is a vital part of the communication process required by agents in the swarm to maintain contact with the base and each other. Important functionalities such as data transfer and action control take place through the communication pipeline. Keeping complete communication is often the first step towards resilient systems. Communication issues include communication delays between swarm agents with one another or with external entities, such as ground control [12]. Swarm agents may fail to communicate with each other due to a variety of reasons. Some agents might stray out of the communication area as a result of path planning and navigational actions. In such cases, the swarm as a whole must be flexible enough to select the optimal agent deployment area by considering communication equipment limitations. Communication at some point might be disconnected completely. This can be due to failure of communication equipment or loss of critical swarm agents responsible for handling connections. Certain UAV task algorithms might overwhelm agent computational capacities to the point that they become unresponsive and reduce the system to a standstill. In-path obstacles might also result in a temporary loss of communication with the swarm. Ongoing research on communication such formation control using ad hoc networks [12] identifies issues and proposes solutions. If some agents in the swarm become disconnected, flexible formation control can restructure swarm positions to bring back agents within the connectivity sphere. Transmission delays can also be offset using formation switching to alternate topology to

position swarm agents closer to broadcast handling agents. Passive beacons installed on the ground can help recover agents from failure by guiding them to failsafe points. Section 2.6 discusses the addition of heterogeneous agents in a swarm as a means of increasing operational resiliency. Ground vehicles can assist in providing emergency communication to aerial swarms and vice versa, as well as perform functions such as visual detection of navigation beacons to coordinate transmission to aerial swarms.

2.1.1. Connectivity

This section deals with maintaining connectivity between the swarm and communication systems at ground control, ground beacons, and the user. The communication path from ground control to UAV agents in operational space has numerous vulnerabilities. Swarms might go off course due to winds or might have to change path due to sudden obstacles. This might affect the range of communication links used such as Wi-Fi and radio. Additionally, obstacles might also block transmissions resulting in delay or loss. Swarm agents with limited fuel capacity have additional problems. Attempts at reconnection and prolonged communication at low signal strength might deplete power reserves more quickly, reducing flight range on the mission. Adaptable connectivity protocols are needed in this scenario [13]. Here, a hierarchical topology is described, where a master drone controls a fleet of lower-level drones that can fly the search area. The master drone acts as a data pathway to the control center. Single link topological frameworks such as this always have the issue of data pathways failing. Since these are also agents that are exposed to operational space dynamics, there is a probability of them failing as well. By assigning different area restrictions to master and low-level drones, the study ensures that there is a persistent data pathway between ground control and end agents. **Figure 1** shows a high-altitude fixed wing aircraft that has the equipment necessary to connect it to a ground-based communication system. A low altitude leader-follower quadcopter swarm is connected to the fixed wing aircraft. The swarm is able to communicate with ground-based stations located at a considerable distance using the fixed wing. However, there is still a probability that the master drone fails due to obstacles. Relay based connectivity maintenance [14] uses a similar communication link that uses relay and articulation UAVs to connect surveillance UAV agents in the swarm to a ground station.

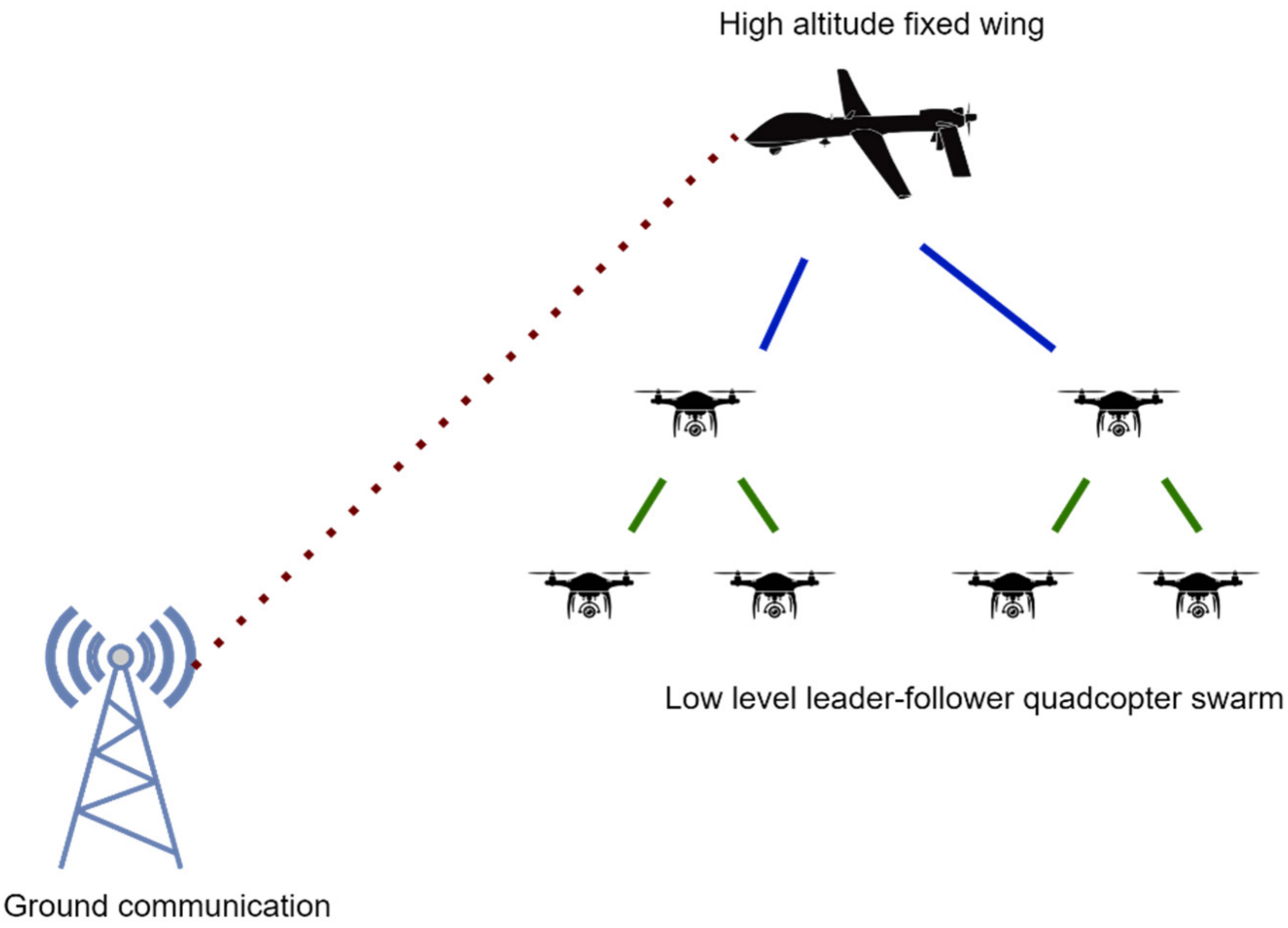

Figure 1. A high-altitude fixed-wing aircraft providing connectivity support to a low-altitude quadcopter swarm arranged in a multi-level hierarchical tree topology.

2.1.2. Network Coverage

Optimal coverage of an area by a swarm of UAVs falls under the connectivity domain. Network coverage involves using the agents in a swarm to blanket a particular area such that all of the area is covered during application-specific purposes such as surveillance, while at the same time maintaining a standard quality of connection with other agents in the swarm. There is a tradeoff between coverage and connectivity that is recognized, and metrics such as those proposed in [15] study it. Multiple UAV swarms have applications in delivering persistent surveillance. However, features such as dynamic target tracking can lead agents astray and out of connectivity limits, which can lead to the loss of data and agents. Simulations on the exploratory problem using MAS show that to accomplish covering a certain area, particularly in cases where high rewards are offered to explore it, an agent might stray from the swarm thereby sacrificing connection with other agents or the base while doing so. Optimized models designed to balance connectivity and coverage such as those designed by [16] study the effects of network density on coverage and performance. Swarms with a smaller number of agents thus need high-level models, balanced with increased computational power, to handle more frequent formation control than swarms with a higher number of agents. A model parameter that requires agents to remain near each other might prevent agents from spreading out thereby restricting coverage. Bio-inspired ant colony optimization algorithms try to imitate ant behavior for

communication and pathways based on pheromones. Chaotic ant colony optimization for coverage-connectivity [15] uses repulsive pheromones to visit unexplored areas. Alternatively, they were also used to avoid collisions between agents. With this approach, less work exists on balancing connectivity and coverage decision processes for swarms.

These forces can be calibrated to consider network strength. Thus, areas that might weaken connections between swarm components are assigned higher values to discourage single agents from attempting exploratory tasks. Intelligent algorithms can however assign a group of agents to visit high force areas such that the entire network shifts in that particular direction thereby maintaining network strength.

2.2. Movement

This component covers the decision process that involves the movement of the swarm agents in operational space, including flocking, optimized area coverage, path planning, and obstacle avoidance. As such, these are the physical behaviors that a swarm might exhibit during its operation. Major modules of the movement component are recognized to be area coverage, path planning, obstacle avoidance, collision avoidance, navigation, formation control, and flocking.

2.2.1. Area Coverage

While Section 2.1.2 discussed tradeoffs between network coverage and connectivity, this section discusses area coverage as a part of the movement component. These problems are often intermixed with each other. The coverage problem determines the success and probability of when the area will be completely scanned. It is often defined as increasing coverage while managing trajectories and disruptions. Article [17] defines swarm coverage as a process to cover a selected region. Swarm area coverage is an important decision process for swarm systems. Area coverage is often utilized in application-specific scenarios to cover a region of interest (ROI) with a swarm of agents, particularly as a movement problem, with less regard for connectivity maintenance and more focus on completing a particular goal. A similar problem framework is discussed in [18]. “Chaos-enhanced mobility models for multilevel swarms of UAVs” mentions how area coverage is an original problem by describing it as “focusing on the mobility management of a swarm of autonomous UAVs to maximize the coverage of a squared geographical area”. The resilient component differentiates this from a simple path-planning objective. The two main objectives are for the swarm to cover a given area as well as counter any unpredictable disruptions that occur during the process. Considering the target area properties is an important input for decision-making models for swarm operations. Network coverage and area coverage differ in terms of how ROI is used. While network coverage focuses on making sure that agents in the area are always optimally connected, area coverage addresses the coverage problem and describes the actions taken by a swarm of agents to cover the ROI. The target space may differ in terrain and ground cover, as well as the presence of water bodies, wind channels, and stationary or dynamic obstacles. Current issues with area coverage lie mostly in the dimensional space in which they are tested. Most existing simulations portray coverage in a two-dimensional space. In [18], altitude information is not considered. Formation deployment controls may require agents flying through constricted spaces to fly in tighter formations

thus requiring agents to vary in flight altitude rather than sweep area. This is especially prevalent when reaching consensus after obstacle navigation and post-deployment primary formations. The control methodology in which agents cover an area can be categorized into different types and [19] offers a naming convention for them as follows. Static coverage is a standard agent deployment method in which an UAV examines a particular spot for its target. Several such agents examine individual spots. Barrier coverage forms a perimeter that can detect the entry of any object through the barrier. These are usually deployed on security-specific applications. Sweeping coverage is the name the authors have given to the dynamic deployments of agents which can change formation as they move through the area. This is the standard procedure followed across any SAR protocols. Several coverage models have been described such as imposing grid cells on the ROI to ensure that every cell is covered or dividing the area into small bits that are assigned to the agent's area decomposition [20][21] and sweep motions [22][23]. Article [17] mentions two primary methods for cooperating coverage, centralized, and distributed decision-making. They propose a self-organized decision-making approach for the problem modeled in velocity space. The approach is divided into perception, decision, and actions. Multiple UAVs coordinate with each other for sharing position, velocity, and obstacle information. Their decision uses a reciprocal coverage method that creates collision-free optimal spaces. The swarm coverage decision model considers the above parameters with collision avoidance with other agents, obstacle avoidance, and optimal velocity decision in each iteration. This is carried out using the Monte Carlo method for velocity-finding in confirmed space.

A different obstacle characteristic is considered by [24] while solving the area coverage problem. Not all obstacles have the same threat levels, nor are all of them equal in terms of dimensions and nature. Additionally, energy constraints on UAVs during coverage problems have often not been addressed in complex scenarios. They propose a two-step auction framework for energy-constrained UAVs in a given area. The agents evaluate the threat levels of each area cell referred to in their paper as a module and bid in an auction for the UAV to come to it. If two bidding prices match, energy loss is also considered. The UAV determines the winning module. The second step is the obstacle avoidance strategy for any obstacles that the UAV might face while traveling to the winning module. In the case that an obstacle is unreachable by flying over it, the second-best module is selected. Additional constraints for sleep mode and two UAV bid clashes are also designed. Such strategies are also viable alternatives to reward-based ones where agents are rewarded for considering a particular area. Additional parameters such as energy considerations can thus be programmed.

2.2.2. Path Planning

Efficient path planning for multi-agent systems is a prevalent challenge in swarm development. Algorithms such as Particle Swarm Optimization (PSO) are developed to find near-optimal solutions. Multiple iterations on a solution may provide better results. Several biologically inspired heuristic algorithms have been proposed for path planning. Bio-inspired algorithms have found remarkable success in the movement development of multi-agent systems as they exist predominately in the animal world. The development of such algorithms drew its inspiration from group behaviors in fish shoals, bees, and ants [25]. Article [26] proposes a modified fruit fly optimization algorithm. The original fruit fly swarming is inspired by fruit flies making their way to ripening fruit. The modified algorithm divides the swarm into smaller subswarms. By allowing flexible search parameters there is a shift from a global search to a

local search as the mission timeline progresses. This expands the search space considerably. The FOA (Fruit fly optimization algorithm) uses two search parameters, sight, and olfactory. While the visual search is a greedy search, the olfactory search is a directional search that examines the greatest concentration of the target in a cell and proceeds in areas with increased concentrations. Both processes are repeated until the termination stage is reached near the target destination. UAV inputs can be used to program essential paths for individual agents. Paths to the destination can be calculated by using terrain data, weather, and network signals as inputs. Based on pre-established network availability from base to destination, each agent can move on a path that has the best-preestablished parameters such as SINR. Secondary inputs such as terrain can be analyzed using the visual input.

Distributed path planning for multi-UAVs works similarly, ref. [25] where SAI (Surveillance Area Importance) values for each cell are analyzed and a connection is established that shares each UAV's location. The leader agent checks the area based on past SAI values, and uncovered areas are subdivided. Individual trajectories for each agent are generated. SAI is an intrinsic value generated and defined by [25] based on the probability of outside agents entering a restricted airspace. Since this is an application-specific development, such values are required. For general purpose use, however, the above-suggested values such as network strengths can be used to establish the grid importance. An alternative approach used by [27] uses external threat models to create channels through which agents can travel. The weather threat model measures wind and rain states and the transmission tower model calculates a safe path some distance away from transmission towers to prevent their electromagnetic waves from disrupting agent navigation systems. An upland threat model measures the terrain below the agents to maintain optimal distance between aircraft and the ground. An adaptive genetic algorithm controls the path generation schema.

Modeling individual threats as functions that act as inputs to learning algorithms is an efficient approach to creating low-cost shortest route solutions. By allowing models to scale as per a particular environment, threat functions that are not present can be eliminated thereby allowing quicker convergence on solutions. For example, the transmission tower threat function will not be used in an area that does not have them, thereby reducing overall model complexity.

2.3. Search-and-Rescue (SAR)

SAR missions are usually defined as an exploration problem. Exploration approaches can be used in a wide range of applications [28]. Target search applications can include searching for an intended target such as an entity in danger or need of medical attention or surveying the aftermath of an accident. The search function can also be expanded to include other agents in the UAV swarm that might have malfunctioned and crashed. Two major applications of the search function are discussed.

- Target search and tracking for entities that are not a part of the swarm.
- Track and search for agents of the swarm to open further conditional processes related to mission progress.

Rescue activity has additional decision parameters. If a crashed agent were located by the swarm, the cost of additional time and fuel that would be required to recover the agent should be incorporated into the model as a function that opposes the primary mission function. Additional conditional statements must be programmed to gauge agent failure in the first place. If the agent has failed due to a locally present disruption, deploying additional agents may result in their loss too.

The search function is an application of UAV where the aerial vehicle hunts for a particular target using vision ability or location information. Searching for a target using an UAV has various challenges. Depending on the hardware used, UAVs may have limited sensing and communication hardware onboard. Capturing raw footage and sending it to the ground station to be processed is computationally costly and may introduce delay. Two types of search algorithms are often used to enable autonomous search: (1) visual search using learning-based detection algorithms [29], and (2) location-based search using active or passive onboard sensor arrays. Search algorithms conventionally divide the area by using a probability index of where the target is most likely to be present [30][31].

2.4. Security

Swarm security is divided into two main categories. Physical agent security and protecting the swarm from cyber threats. Both are discussed in the following sections.

2.4.1. Physical Security

The physical security of agents deals with the detection of threats that might physically impede swarm progress. Additionally, defense or escape countermeasures should be designed as part of securing any multi-agent system. The counteraction from swarm agents is a response of the agents once a threat is detected and can include the following.

- Counteraction of UAV swarms against malicious agents trying to take down agents in the swarm
- Counterattack of UAV swarms against malicious agents trying to enter a restricted airspace

In either case, a threat classifier is needed for UAV swarms to detect and recognize potential threats. One defense approach to incoming malicious agents is to engage a swarm of counter-attack drones to intercept intruders. The approach by [32] follows a similar method by deploying a swarm of drones that approaches the intruder UAV. The deployed defense agents form a cluster around the intruder and restrict its movement while attempting to herd it to a non-threatening location. The assumption is that the enemy agent is aware of other agents surrounding it and will take steps to prevent collision with them. However, if an enemy agent is not equipped with such abilities, there is a high possibility that it may collide with one or more of the defender agents and result in damage or loss. UAV agents can also be used to jam network connections of enemy agents and stop their crucial operations. A GPS spoofing attack was proposed in [33] where it attempts to take control of an enemy agent. This is considered accomplished when it can successfully artificially specify the enemy agent's perceived position and velocity. By controlling the agent and providing false data it is possible to disable the enemy. For example, the spoofer used in

this study earlier demonstrated similar actions whereby it falsely produced ascending actions on a captured UAV that was hovering. To compensate, the agent started a descent and would have been catastrophic unless precautionary manual control took over [33][34]. However, such solutions which involve the deployment of additional drones for counterattack can be an expensive process considering the physical interactions that might take place among these agents. Replacement and repair of damaged defender aircraft can be costly. Such approaches should be deployed only, when necessary, when the main swarm is deemed incompetent to defend itself. Other solutions exist that can be accommodated onboard existing UAVs without the need for additional agents. Enemy agent ability jamming and evasive maneuvers are often advised in UAV swarm defense [35]. For recognition of enemy agents, detection methods are necessary, especially to differentiate between swarm agents and foreign agents. While this can be done using software in-loop, network recognition, and unique hardware identifiers, vision-based frameworks are also being studied. For example, ref. [36] uses a vision-based object detection method backed by deep learning to detect and track a potential enemy UAV. In addition to detection, a tracking system is implemented to keep the detected agent in a local bounding box and follow it.

2.4.2. Network Security

A large portion of control tasks for UAVs is dependent on a network structure. Any device, node, or link over a network is susceptible to cyber-attacks. With recent developments of malicious agents attempting to dissuade swarms from functioning, damaging the network capabilities of an UAV swarm is the primary method of executing attacks [37]. Moreover, the remote nature of UAVs, combined with limited battery power, fast switching routing, formation topologies, and small onboard computing power has made securing drone networks a challenge. As with any network, FANETs are susceptible to attacks as well, more so because of their high mobility and reduced computational powers. Onboard agents have reduced computational power, and most of the processor load might be dedicated to functions such as flight, navigation, and mission tasks. Any security measures thus need to consume as little energy and computation resources as possible [38].

Although attacks such as eavesdropping can be prevented using encryption for transmission, other attacks may still penetrate UAV networks. The encryption keys must be secure themselves to guarantee performance. Current security and management deployments assume that UAV swarms may accomplish tasks on a single charge. If refueling is needed, the subtraction of old agents and the addition of new swarm agents should be considered. Article [39] creates a swarm broadcast protocol that accounts for rapid changes in swarm numbers, such as the addition of new drones. It also accounts for agents leaving the swarm for activities such as fuel recharge. The addition of new agents may require them to use the encryption keys used to secure network transmissions. However, challenges to this approach include offset delays that may be caused by validating new agents, the transmission of keys, and unstable networks interrupting the key transfer process. A loss of key transmission could cause new agents to be unable to decrypt transmissions [39]. The requirements of the key management scheme proposed are also similar to the IDS requirements discussed in the next section. A lightweight management scheme that consumes low network and CPU resources is desirable. Secure broadcast protocols work whereby every time the swarm changes its agent composition a new secure key is created. This is done to prevent the old key from being used by any attackers. The new agents have separate identifiers that recognize them to be a part of

the swarm, this identifier unlocks the broadcast packet that contains the new key. The master maintains a list of all agent identifiers actively connected, and every time an agent sends a leave or join request the identifier table gets updated, and a new broadcast key is sent out. To prevent offset delays during key sharing, agents also receive an updated copy of the identifier table. They can then verify that the nearest neighbor is a verified agent and send the key packet. In this way, the master does not have to send the key to every individual agent. Intrusion detection systems are a reliable and efficient way of securing computer networks. Recent studies demonstrate they can be deployed on UAV networks as well [40].

As with a regular IDS, an UAV-IDS is created to detect any suspicious activity on the network and prevent its execution. Intrusion detection systems can be developed based on behaviors or anomalies. Bio-inspired particle detection IDS can also be used [41]. An updated taxonomy of intrusion detection systems has been conducted by the authors in previous research [42]. The IDS monitors network traffic such as data packets, transmission power, and routes, as well as new incoming requests from nodes.

While traditional attacks for computer networks may not directly apply here, IDS protocols can be easily modified to suit FANET requirements. An integration with Hyperledger Fabric, a distributed ledger platform, is possible to maintain unique identifiers for all acceptable nodes thereby blocking any unknown nodes from gaining access [43]. An IDS functionality can also be influenced by its deployment location. Typically, an IDS located at ground control may have access to higher resources than on-board implementations. The placement can be determined depending on factors such as the level of protection required, the type of IDS being used, as well as resource requirements of the IDS itself. Multi-layered IDS are possible but require additional considerations of network and node capacity during system development. A policy-based IDS can be created that sets distinct patterns of behavior that are allowed whereas all others are flagged and sent to a higher-level operation for additional examination. Several research challenges as mentioned by [29] discuss detection latency, IDS computational costs, and implementation overheads. A solution is required that is multi-positioned (agents and ground control both), multi-layered, and comprehensive in terms of attack detection and mitigation. Add to that the challenges faced by current IDS implementations, such as bottlenecks due to higher bandwidth and a lack of concrete defense policies [44], and attack classifiers contribute further challenges to the problem. Articles [45][46] have a rule-based mechanism for the IDS. Detection rules for some of the most notorious attack types are predefined. The location is at the ground station and categorizes each agent in the swarm by its alleged threat. An intrusion report message is generated by each agent and sent to the ground station, which assists in verifying agent status as well as any foreign agents that might seek to infiltrate the swarm. A verification check is conducted by using anomaly detection backed by learning algorithms trained to identify and label behaviors.

The addition of learning algorithms significantly expands the capability of intrusion detection systems, and blockchain-supported network security has also been recently explored as a way of securing UAV swarms. Blockchain is a peer-to-peer (P2P) distributed ledger that is secure, transparent, and flexible. It can store data in a chain of blocks that are tamper-proof. Smart contracts can be developed on the chain to execute specific predefined actions. Blockchain has widespread use in security, finance, and government applications [47]. Blockchain technology is a viable solution to issues with UAV network security.

Current UAV softwarization techniques are summarized in [43] such as the wide range of attacks on control, application, and data plane. Several proposed blockchain-based architectures are a comprehensive solution to these issues. **Figure 2** demonstrates a simplified blockchain based method for securing low level swarms. A blockchain instance runs on a cloud and ground-based station. Every agent in the swarm has a hardcoded unique identifier. A state table with the agents and their identifiers is maintained by the blockchain and synced across the station nodes. Agents validate their existence at every iteration and the blockchain is updated accordingly. A spoofed agent with an identifier that is not present on the blockchain is unable to join the swarm, is denied communication services and is recognized immediately.

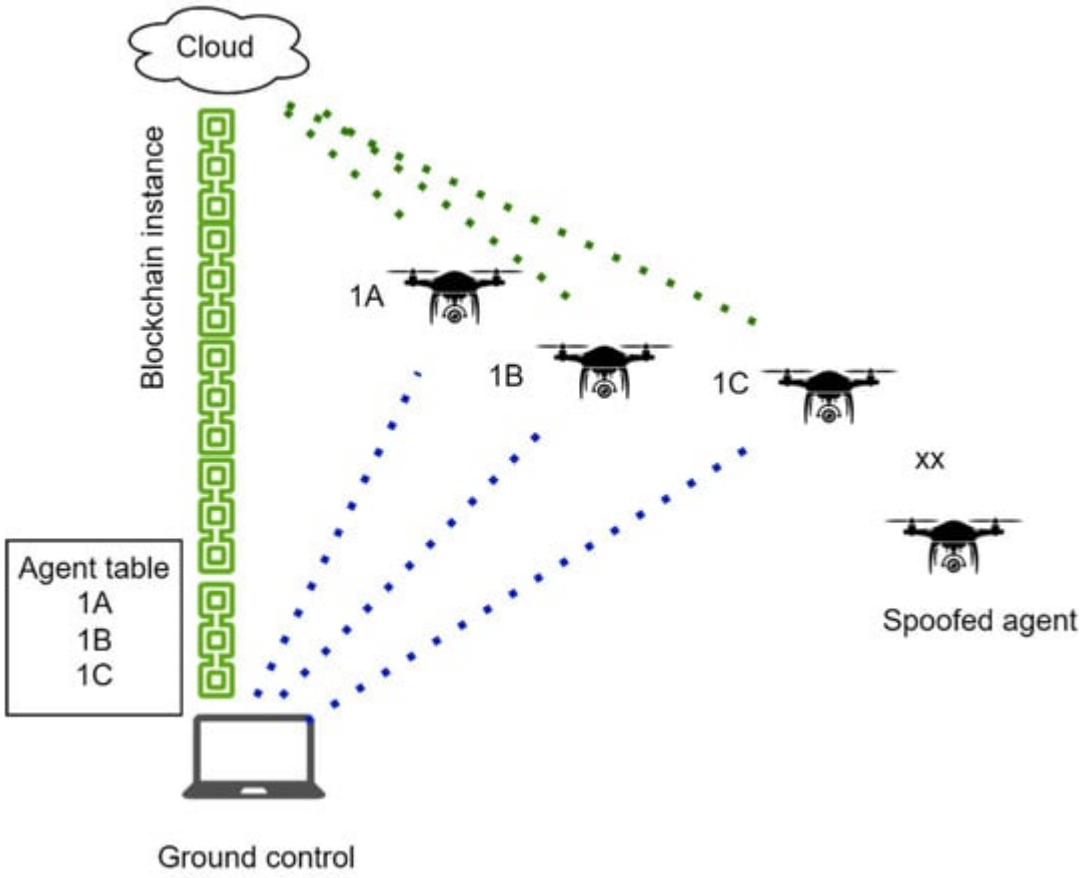

Figure 2. A blockchain based methodology for securing UAV swarms.

Integration benefits include communication data security and transparency in node transactions. However current blockchain technology has scaling and latency issues [48]. A proper framework choice is needed that fits the stochastic requirements of UAV swarms. The first step is the selection of an appropriate consensus algorithm for the blockchain itself. Fast consensus algorithms instead of proof of work are required. Article [49] recommends PBFT (practical byzantine fault tolerance) consensus that has a frequency of 500 Hz. Such higher frequencies can satisfy the intensive demands of routing policies, resource allocation, and mobility management in addition to deploying secure swarms. Article [43] in particular focuses on the security and privacy of communication links for UAVs via blockchain-supported softwarization architecture. By ensuring the authenticity of virtual machines in the virtual infrastructure, it is also capable of protecting against various security threats.

2.5. Resource and Task Handling

Resource allocation and task assignment are terms used by some researchers interchangeably. This is based on the premise that once a task is assigned to an agent or a group of agents, their resources will be locked during the duration of task completion or until a dynamic change is required in the decision-making capacity of the task assignment module. However, not all studies incorporate higher levels into task assignments. The assumption is that each agent in the swarm is a resource that is assigned to complete a particular task. The problem is an interaction between resources and the environment, whereas the allocation scheme is an incentivized function for the agents.

2.6. Agent Property

The agent property component focuses on individual agents in the swarm. It is possible to increase overall swarm performance by modifying individual agent capabilities [50][51]. The easiest way to do this is to introduce swarm heterogeneity. Heterogeneity is defined as components of a system that are of dissimilar composition or properties. Heterogeneity may be imbibed in a multi-agent system by using a variety of features. The following studies focus on performance effects on a mission by the inclusion of heterogenous agents in a previously homogenous swarm [50][51][52]. There is a marked increase in performance observed by the introduction of varied agents. This performance might be in terms of time taken to complete tasks or an increase in another measurement metric used to measure swarm resilience. Our previous research provides a classification system for swarms labeled as heterogeneous:

- By operational space of agents
- By nature of agents
- By hardware of agents

In addition to aerial spaces, other varied operational spaces for the unmanned vehicle in a swarm may include ground surfaces, water surfaces, underwater, or even underground. Article [4] deploys a heterogenous team in an enclosed environment. The team comprises robots classified as heterogenous by their operational space—an UGV and an UAV. The authors mention how each vehicle brings different capabilities to the swarm. The UGV is autonomous whereas the UAV is fast and provides greater motion flexibility. The UAV can fly over obstacles that the UGV cannot cross. Challenges to the deployment of such hybrid swarms are also mentioned. Air ground coordination and navigation are the primary challenges. Deployment of such swarms is not limited to enclosed environments only. Indeed, a wider operational space is available by the inclusion of agents on the ground, water surfaces, or even at different altitudes in the air. Current deployments of heterogenous agents have shown a marked increase in resilient behavior [50][51][52]. Such swarms can be single or multi-space or heterogonous by nature or have hardware capability. In pursuit–evader situations, the pursuit UAV often needs to be agile and lightweight to be capable of faster flight speeds than the target UAV. However, a swarm that features a mix of lightweight pursuit-capable agents along with heavier support agents that carry additional equipment needed to

support swarm operations is ideal. A military purpose swarm such as the one in [53] where a mix of differently capable agents is used can be used. Moreover, experiments by [51] observed a marked increase in swarm responsiveness to external stimuli due to the inclusion of fast agents in swarms. Benefits are observed across the whole range of UAV swarm components. A hierarchical structure of mixed agents described by [54] includes high-altitude fixed-wing aircraft providing communication and sensing support to a group of low-altitude quadcopters. They also note a difference in computational resources, energy consumption, and communication systems among the agents. The possibility of evolving capabilities among heterogeneous swarms has also been observed. Implementing a different characteristic trait by modifying operational parameters can change the way homogenous agents behave thereby allowing them to function better. A change in the allowable range that some swarm agents can move from a fixed ground target can allow them to become less cautious of external disruptions, thereby exhibiting different behavior. A coordination scheme by other non-modified agents to balance tasks and resources is observed in such scenarios. The addition of differently capable agents can also provide complementary capabilities [55]. Homogenous agents in a swarm may have a particular weakness that may easily bring the whole swarm down. Such issues can be varied such as bugs in the control system or routing protocols, detection and tracking, or communication range. Such issues can be solved by introducing agents that might fix such gaps in the overall swarm properties. For example, a relay drone that possesses extra hardware to support multiple communication bands can aid in communication gaps and delays. A similar deployment is described in [56], where some robots in the group have higher sensor payload, processing power, and memory capacity. These are labelled as leader robots, whereas child robots have more limited resources. These child robots rely on leader robots for tasks such as localization. Data from the child robots can be used for sensor fusion, global pose, and location estimation, which can then be used to modify the swarm movement. The proposed distributed leader-assisted localization algorithm can provide accurate localizations for child robots even when they are beyond the sensing range of leader agents.

Application-specific improvements such as those in [55] show how a richer dataset is created when a micro aerial vehicle (MAV) collaborates with an UGV in the creation of detailed maps. The MAV recorded the top view dimensions whereas the UGV accounted for the side view spread. Added considerations for implementation need to be accounted for in this type of scenario though. **Figure 3** shows how an UAV-UGV coordination can produce higher detailed maps. The UGV maps the ground level dimensions while the UAV maps the top view.

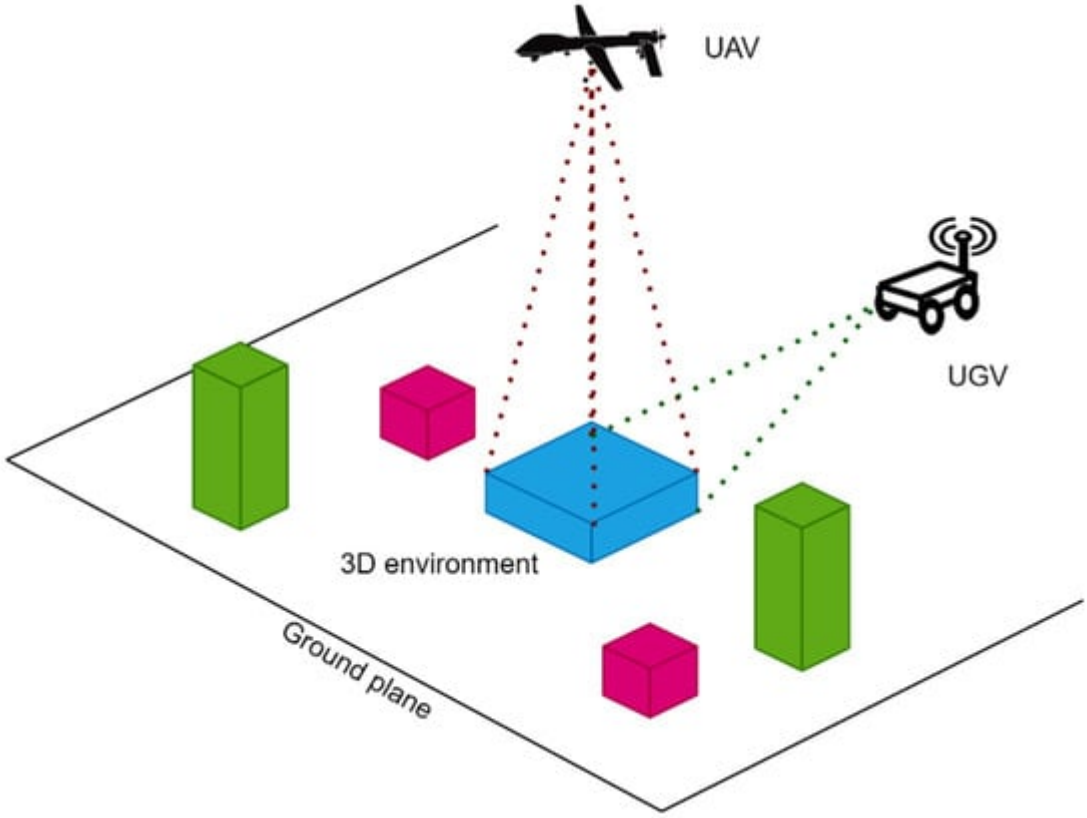

Figure 3. A multi space unmanned vehicle swarm working a mapping mission in tandem.

2.7. Resiliency Evaluation

Cyber-physical systems need testbeds for simulations before real-world deployments. However, due to the extreme software-hardware interdependence, designing them is a challenging task. UAV swarms are among those systems that require thorough testing but are difficult to test in controlled environments. Simulations cannot mimic the varied and extreme variables they might face [57]. Resiliency assessment is a process of observing and evaluating the resilient behavior of UAV swarms to disruptions. An evaluation metric system is needed to accurately recognize performance and its lack thereof. It is important to devise feedback systems and metrics to measure resilience performance in the above modules. Real-time feedback loops may allow decision-making models to follow an iterative process in creating better outputs. Metrics created to evaluate system components such as signal strength, target detection times based on varied inputs, and fuel saved based on optimized path planning can provide insights and recognize tradeoffs.

Evaluation metric design is a less explored branch. It is a strategic process for evaluating multi-pronged decision models by measuring individual decision branches and quantifying their outputs into a single value. Article [58] cites a flaw in the creation of resilience evaluation metrics. Metrics often use the initial swarm performance as a baseline. This may not be accurate, as certain onboard systems may need more time to exhibit full performance capacity. The approach by [58] introduces variables that provide free space for missions. By relaxing the condition that the swarm has to return to performance before it is attacked, the authors argue that a flexible baseline that determines whether a system is performing is needed. The swarm may have to take certain actions during the

attack process such as sacrificing a certain agent or reducing the number of agents assigned to cover an area after an attack takes place. Article [58] lays out preliminary dynamic evaluation parameters.

The second problem is that these metrics often use network connectivity as a basis of evaluation. As long as the swarm maintains a connected state in terms of its required signal strength or coverage, the swarm is deemed resilient. However, the presence or absence of network and communication capabilities alone cannot determine the robustness of swarms. Time values for the recovery process should also be measured. For example, in the case of an attack scenario on a swarm described by [58], there are a few issues that the article does not take into account:

- The swarm does not check for agent wellbeing after it determines that the attack has ended.
- In the case that an agent is lost, there are no search and recovery procedures.
- Mission progress may be lost when swarm control completes task re-assignment. In this case, depending upon the scenario tasks such as localization, area decomposition and data collection may need to be restarted after the loss of data is examined.

A basic evaluation metric is the one proposed by [59] which measures performance loss after a disruption with the time it takes for the system to return to normal levels. This creates a baseline through which preliminary system performance can be maintained. Additional application-specific constraints can be established. On the occasion that an agent is lost, the time taken for the mayday signal of the agent to reach central command can be measured along with the time it takes for other agents to locate the lost agent and deploy rescue procedures. A secondary decision process is started where probability models determine if it is safe to try and locate the lost agent or proceed with the task. If the decision to proceed is made, the model then decides if additional agents should be deployed to make up for lost agents or if task reassignment should distribute the workload to other agents. The fundamental metric here is the time value which is measured for every decision to be executed. Multiple time values can be fed into a measurement model to produce one time-based value of system performance.

References

1. Rubio, F.; Valero, F.; Llopis-Albert, C. A review of mobile robots: Concepts, methods, theoretical framework, and applications. Int. J. Adv. Robot. Syst. 2019, 16, 172988141983959.
2. Santa Ana, R. Drones Survey Waning Red Tide. Available online: https://agrilifetoday.tamu.edu/2015/10/22/drones-survey-waning-red-tide-at-south-padre-island/ (accessed on 8 October 2022).
3. Rieucau, G.; Kiszka, J.J.; Castillo, J.C.; Mourier, J.; Boswell, K.M.; Heithaus, M.R. Using unmanned aerial vehicle (UAV) surveys and image analysis in the study of large surface-

associated marine species: A case study on reef sharks Carcharhinus melanopterus shoaling behaviour. J. Fish Biol. 2018, 93, 119–127.

4. Roldán, J.; Garcia-Aunon, P.; Garzón, M.; de León, J.; del Cerro, J.; Barrientos, A. Heterogeneous Multi-Robot System for Mapping Environmental Variables of Greenhouses. Sensors 2016, 16, 1018.
5. Tsouros, D.C.; Bibi, S.; Sarigiannidis, P.G. A Review on UAV-Based Applications for Precision Agriculture. Information 2019, 10, 349.
6. Anderson, K.; Gaston, K.J. Lightweight unmanned aerial vehicles will revolutionize spatial ecology. Front. Ecol. Environ. 2013, 11, 138–146.
7. Vincent, P.; Rubin, I. A Framework and Analysis for Cooperative Search Using UAV Swarms. In Proceedings of the 2004 ACM Symposium on Applied Computing (SAC ’04), Nicosia, Cyprus, 14–17 March 2004; Association for Computing Machinery: New York, NY, USA, 2004; pp. 79–86.
8. Aljehani, M.; Inoue, M. Multi-UAV tracking and scanning systems in M2M communication for disaster response. In Proceedings of the 2016 IEEE 5th Global Conference on Consumer Electronics, Kyoto, Japan, 11–14 October 2016; IEEE: New York, NY, USA, 2016; pp. 1–2.
9. Arafat, M.Y.; Moh, S. Routing Protocols for Unmanned Aerial Vehicle Networks: A Survey. IEEE Access 2019, 7, 99694–99720.
10. Isufaj, R.; Omeri, M.; Piera, M.A. Multi-UAV Conflict Resolution with Graph Convolutional Reinforcement Learning. Appl. Sci. 2022, 12, 610.
11. Woods, D.D. Four concepts for resilience and the implications for the future of resilience engineering. Spec. Issue Resil. Eng. 2015, 141, 5–9.
12. Suo, W.; Wang, M.; Zhang, D.; Qu, Z.; Yu, L. Formation Control Technology of Fixed-Wing UAV Swarm Based on Distributed Ad Hoc Network. Appl. Sci. 2022, 12, 535.
13. Celtek, S.A.; Durdu, A.; Kurnaz, E. Design and Simulation of the Hierarchical Tree Topology Based Wireless Drone Networks. In Proceedings of the 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), Malatya, Turkey, 28–30 September 2018; IEEE: New York, NY, USA, 2018; pp. 1–5.
14. Zhu, Q.; Zhou, R.; Zhang, J. Connectivity Maintenance Based on Multiple Relay UAVs Selection Scheme in Cooperative Surveillance. Appl. Sci. 2016, 7, 8.
15. Rosalie, M.; Brust, M.R.; Danoy, G.; Chaumette, S.; Bouvry, P. Coverage Optimization with Connectivity Preservation for UAV Swarms Applying Chaotic Dynamics. In Proceedings of the 2017 IEEE International Conference on Autonomic Computing (ICAC), Columbus, OH, USA, 17–21 July 2017; IEEE: New York, NY, USA, 2017; pp. 113–118.

16. Yanmaz, E. Connectivity versus area coverage in unmanned aerial vehicle networks. In Proceedings of the 2012 IEEE International Conference on Communications (ICC), Ottawa, ON, Canada, 10–15 June 2012; IEEE: New York, NY, USA, 2012; pp. 719–723.
17. Chen, R.; Xu, N.; Li, J. A Self-Organized Reciprocal Decision Approach for Sensing Coverage with Multi-UAV Swarms. Sensors 2018, 18, 1864.
18. Elmokadem, T.; Savkin, A.V. Computationally-Efficient Distributed Algorithms of Navigation of Teams of Autonomous UAVs for 3D Coverage and Flocking. Drones 2021, 5, 124.
19. Reynolds, C.W. Flocks, Herds and Schools: A Distributed Behavioral Model. SIGGRAPH Comput. Graph. 1987, 21, 25–34.
20. Acar, E.U.; Choset, H.; Rizzi, A.A.; Atkar, P.N.; Hull, D. Morse Decompositions for Coverage Tasks. Int. J. Robot. Res. 2002, 21, 331–344.
21. Choset, H.; Pignon, P. Coverage Path Planning: The Boustrophedon Cellular Decomposition. In Field and Service Robotics; Zelinsky, A., Ed.; Springer: London, UK, 1998; pp. 203–209.
22. Huang, W.H. Optimal line-sweep-based decompositions for coverage algorithms. In Proceedings of the 2001 ICRA, IEEE International Conference on Robotics and Automation (Cat. No.01CH37164), Seoul, Korea, 21–26 May 2001; IEEE: New York, NY, USA, 2001; Volume 21, pp. 27–32.
23. Gonzalez, E.; Alvarez, O.; Diaz, Y.; Parra, C.; Bustacara, C. BSA: A Complete Coverage Algorithm. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; IEEE: New York, NY, USA, 2005; pp. 2040–2044.
24. Sun, Y.; Tan, Q.; Yan, C.; Chang, Y.; Xiang, X.; Zhou, H. Multi-UAV Coverage through Two-Step Auction in Dynamic Environments. Drones 2022, 6, 153.
25. Ahmed, N.; Pawase, C.J.; Chang, K. Distributed 3-D Path Planning for Multi-UAVs with Full Area Surveillance Based on Particle Swarm Optimization. Appl. Sci. 2021, 11, 3417.
26. Shi, K.; Zhang, X.; Xia, S. Multiple Swarm Fruit Fly Optimization Algorithm Based Path Planning Method for Multi-UAVs. Appl. Sci. 2020, 10, 2822.
27. Liu, H.; Ge, J.; Wang, Y.; Li, J.; Ding, K.; Zhang, Z.; Guo, Z.; Li, W.; Lan, J. Multi-UAV Optimal Mission Assignment and Path Planning for Disaster Rescue Using Adaptive Genetic Algorithm and Improved Artificial Bee Colony Method. Actuators 2021, 11, 4.
28. Andrade, F.A.A.; Hovenburg, A.; de Lima, L.N.d.; Rodin, C.D.; Johansen, T.A.; Storvold, R.; Correia, C.A.M.; Haddad, D.B. Autonomous Unmanned Aerial Vehicles in Search and Rescue Missions Using Real-Time Cooperative Model Predictive Control. Sensors 2019, 19, 4067.

29. Opromolla, R.; Inchingolo, G.; Fasano, G. Airborne Visual Detection and Tracking of Cooperative UAVs Exploiting Deep Learning. Sensors 2019, 19, 4332.
30. Bertuccelli, L.F.; How, J.P. Robust UAV search for environments with imprecise probability maps. In Proceedings of the 44th IEEE Conference on Decision and Control, Seville, Spain, 15 December 2005; IEEE: New York, NY, USA, 2005; pp. 5680–5685.
31. Yang, Y.; Minai, A.A.; Polycarpou, M.M. Decentralized cooperative search by networked UAVs in an uncertain environment. In Proceedings of the 2004 American Control Conference, Boston, MA, USA, 30 June–2 July 2004; IEEE: New York, NY, USA, 2004; pp. 5558–5563.
32. Brust, M.R.; Danoy, G.; Stolfi, D.H.; Bouvry, P. Swarm-based counter UAV defense system. Discov. Internet Things 2021, 1, 2.
33. Kerns, A.J.; Shepard, D.P.; Bhatti, J.A.; Humphreys, T.E. Unmanned Aircraft Capture and Control Via GPS Spoofing: Unmanned Aircraft Capture and Control. J. Field Robot. 2014, 31, 617–636.
34. Shepard, D.P.; Bhatti, J.A.; Humphreys, T.E.; Fansler, A.A. Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks. In Proceedings of the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012), Nashville, TN, USA, 17–21 September 2012; pp. 3591–3605.
35. Brust, M.R.; Danoy, G.; Bouvry, P.; Gashi, D.; Pathak, H.; Goncalves, M.P. Defending Against Intrusion of Malicious UAVs with Networked UAV Defense Swarms. In Proceedings of the 2017 IEEE 42nd Conference on Local Computer Networks: Workshops (LCN Workshops), Singapore, 9 October 2017; IEEE: New York, NY, USA, 2017; pp. 103–111.
36. Akhloufi, M.A.; Arola, S.; Bonnet, A. Drones Chasing Drones: Reinforcement Learning and Deep Search Area Proposal. Drones 2019, 3, 58.
37. Pu, C. Jamming-Resilient Multipath Routing Protocol for Flying Ad Hoc Networks. IEEE Access 2018, 6, 68472–68486.
38. Aznar, F.; Pujol, M.; Rizo, R.; Pujol, F.; Rizo, C. Energy-Efficient Swarm Behavior for Indoor UAV Ad-Hoc Network Deployment. Symmetry 2018, 10, 632.
39. Secinti, G.; Darian, P.B.; Canberk, B.; Chowdhury, K.R. Resilient end-to-end connectivity for software defined unmanned aerial vehicular networks. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; IEEE: New York, NY, USA, 2017; pp. 1–5.
40. Choudhary, G.; Sharma, V.; You, I.; Yim, K.; Chen, I.-R.; Cho, J.-H. Intrusion Detection Systems for Networked Unmanned Aerial Vehicles: A Survey. In Proceedings of the 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), Limassol, Cyprus, 25–29 June 2018; pp. 560–565.

41. Sharma, M.; Saini, S.; Bahl, S.; Goyal, R.; Deswal, S. Modified Bio-Inspired Algorithms for Intrusion Detection System. In Proceedings of the International Conference on Innovative Computing and Communications, Delhi, India, 20–21 February 2021; Gupta, D., Khanna, A., Bhattacharyya, S., Hassanien, A.E., Anand, S., Jaiswal, A., Eds.; Springer: Singapore, 2021; pp. 185–201.
42. Phadke, A.; Ustymenko, S. Updating the Taxonomy of Intrusion Detection Systems. In Proceedings of the 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 12–16 July 2021; pp. 1085–1091.
43. Kumari, A.; Gupta, R.; Tanwar, S.; Kumar, N. A taxonomy of blockchain-enabled softwarization for secure UAV network. Comput. Commun. 2020, 161, 304–323.
44. Tan, X.; Su, S.; Zuo, Z.; Guo, X.; Sun, X. Intrusion Detection of UAVs Based on the Deep Belief Network Optimized by PSO. Sensors 2019, 19, 5529.
45. Li, L.; Zhang, H.; Peng, H.; Yang, Y. Nearest neighbors based density peaks approach to intrusion detection. Chaos Solitons Fractals 2018, 110, 33–40.
46. Sedjelmaci, H.; Senouci, S.M.; Ansari, N. A Hierarchical Detection and Response System to Enhance Security Against Lethal Cyber-Attacks in UAV Networks. IEEE Trans. Syst. Man Cybern. Syst. 2018, 48, 1594–1606.
47. Phadke, A.; Medrano, F.A.; Ustymenko, S. Applications of Blockchain in E-government. In Proceedings of the 2022 International Symposium on Electrical, Electronics and Information Engineering (ISEEIE), Chiang Mai, Thailand, 25–27 February 2022; pp. 157–164.
48. Jensen, I.J.; Selvaraj, D.F.; Ranganathan, P. Blockchain Technology for Networked Swarms of Unmanned Aerial Vehicles (UAVs). In Proceedings of the 2019 IEEE 20th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM), Washington, DC, USA, 10–12 June 2019; pp. 1–7.
49. Hu, N.; Tian, Z.; Sun, Y.; Yin, L.; Zhao, B.; Du, X.; Guizani, N. Building Agile and Resilient UAV Networks Based on SDN and Blockchain. IEEE Netw. 2021, 35, 57–63.
50. Scheutz, M.; Schermerhorn, P.; Bauer, P. The utility of heterogeneous swarms of simple UAVs with limited sensory capacity in detection and tracking tasks. In Proceedings of the 2005 IEEE Swarm Intelligence Symposium, 2005. SIS 2005, Pasadena, CA, USA, 8–10 June 2005; pp. 257–264.
51. Kwa, H.L.; Tokić, G.; Bouffanais, R.; Yue, D.K.P. Heterogeneous Swarms for Maritime Dynamic Target Search and Tracking. In Proceedings of the Global Oceans 2020: Singapore—U.S. Gulf Coast, IEEE/MTS OCEANS 2020, Singapore, 5–30 October 2020; pp. 1–8.
52. Gade, S.; Joshi, A. Heterogeneous UAV swarm system for target search in adversarial environment. In Proceedings of the 2013 International Conference on Control Communication and

Computing (ICCC), Thiruvananthapuram, India, 13–15 December 2013; pp. 358–363.

53. Ramana Makkapati, V.; Tsiotras, P. Apollonius Allocation Algorithm for Heterogeneous Pursuers to Capture Multiple Evaders. arXiv 2020, arXiv:2006.10253.
54. Xu, C.; Zhang, K.; Jiang, Y.; Niu, S.; Yang, T.; Song, H. Communication Aware UAV Swarm Surveillance Based on Hierarchical Architecture. Drones 2021, 5, 33.
55. Dewan, A.; Mahendran, A.; Soni, N.; Krishna, M. Optimization Based coordinated uGV-MAV exploration for 2D augmented mapping. In Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems, St. Paul, MN USA, 6–10 May 2013; pp. 1125–1126.
56. Wanasinghe, T.R.; Mann, G.K.I.; Gosine, R.G. Distributed Leader-Assistive Localization Method for a Heterogeneous Multirobotic System. IEEE Trans. Autom. Sci. Eng. 2015, 12, 795–809.
57. Kumar, P.S.; Emfinger, W.; Karsai, G. A testbed to simulate and analyze resilient cyber-physical systems. In Proceedings of the 2015 International Symposium on Rapid System Prototyping (RSP), Amsterdam, The Netherlands, 8–9 October 2015; pp. 97–103.
58. Sun, Q.; Li, H.; Zhang, Y.; Xie, Y.; Liu, C. A Baseline Assessment Method of UAV Swarm Resilience Based on Complex Networks. In Proceedings of the 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl’any, Slovakia, 21–23 January 2021; pp. 83–86.
59. Tierney, K.; Bruneau, M. Conceptualizing and measuring resilience: A key to disaster loss reduction. TR News 2007, 17, 14–15.