future internet

MDPI

Article

Blockchain Technology for Secure Communication and Formation Control in Smart Drone Swarms

Athanasios Koulianos and Antonios Litke *

Department of Mathematics and Engineering, Hellenic Army Academy, 16673 Athens, Greece; koulianos.ath@gmail.com
* Correspondence: alitke@sse.gr

Abstract: Today, intelligent drone technology is rapidly expanding, particularly in the defense industry. A swarm of drones can communicate, share data, and make the best decisions on their own. Drone swarms can swiftly and effectively carry out missions like surveillance, reconnaissance, and rescue operations, without exposing military troops to hostile conditions. However, there are still significant problems that need to be resolved. One of them is to protect communications on these systems from threat actors. In this paper, we use blockchain technology as a defense mechanism against such issues. Drones can communicate data safely, without the need for a centralized authority (ground station), when using a blockchain to facilitate communication between them in a leader–follower hierarchy structure. Solidity has been used to create a compact, lightweight, and effective smart contract that automates the process of choosing a position in a certain swarm formation structure. Additionally, a mechanism for electing a new leader is proposed. The effectiveness of the presented model is assessed through a simulation that makes use of a DApp we created and Gazebo software. The purpose of this work is to develop a reliable and secure UAV swarm communication system that will enable widespread global adoption by numerous sectors.

Keywords: IoT; drone swarms; formation control; security; blockchain; smart contract; Gazebo; mavsdk

Citation: Koulianos, A.; Litke, A. Blockchain Technology for Secure Communication and Formation Control in Smart Drone Swarms. *Future Internet* **2023**, *15*, 344. https://doi.org/10.3390/fi15100344

Academic Editor: Yuansong Qiao

Received: 14 September 2023
Revised: 6 October 2023
Accepted: 16 October 2023
Published: 19 October 2023

1. Introduction

Drone swarms have gained a lot of traction in recent years for use in military applications thanks to their versatility in a range of tasks, including search and rescue operations, surveillance, reconnaissance, and more. These missions do, of course, have their limitations. Research has been conducted to identify the best solution to NP-hard problems like path planning and obstacle avoidance using swarm intelligence algorithms [1–3]. A detailed study on swarm intelligence and its role in UAV systems through a layered approach can be found in [4]. In addition to these problems, safeguarding the communications of these unmanned aerial vehicle systems is the key challenge we must overcome. Drones in a swarm have to communicate between them in order to efficiently coordinate their actions. However, there might be obstructions that prevent them from communicating, either naturally (like obstacles) or artificially (like cyberattacks) or because of poor network coverage in some places. The security of these technologies, as an issue, cannot be ignored. Cyber attacks are known to be susceptible to drones [5]. A successful attack on one drone could have severe effects on the entire mission. Furthermore, if the communication links are insecure and the data are not encrypted, an attacker might very simply acquire vital military data, exposing entire operations and endangering lives.

Another significant challenge with drone swarms is the problem of trust. A seamless and effective operation is facilitated by mutual trust among swarm members. Drones should put their trust in the other swarm members and make sure they behave properly and by the rules. On the other hand, drones may behave dishonestly or fail to make

effective decisions, which could, at best, lead to mission failure. A comprehensive review of these issues and the solutions researchers created to overcome them can be found in [6].

One of these solutions relies on the use of blockchains. Blockchain technology is an innovative and promising idea that can offer solutions to various problems in different sectors, including the defense industry. Its undisputed, unmodifiable, and fully transparent environment in terms of transactions is attracting more and more users every day.

Since its inception, researchers have focused their attention on the technology that underlies blockchains with the goal of employing it outside the finance industry. Works related to blockchains, as well as the areas they focus on, can be found in [7–9]. Applications of blockchains in the Internet of Things (IoT), storage and data transfer, network security, and data privacy are some of the modern issues that concern researchers worldwide. Therefore, blockchains find applications in businesses, governments, the health sector, and even the military due to the security and transparency they offer. This technology is especially appealing to the military, where data and communication security are crucial.

Research into the use of blockchain technology in drone swarms has huge implications for various industries and fields, including military, commercial, and political applications. Improved security, privacy, and trust among members, as well as better coordination and performance, are potential benefits of such systems. These studies may consequently result in a rise in the use and creation of technologies that alter how tasks like disaster relief and surveillance are carried out. In terms of ethical, legal, and social problems, research on blockchain technology and its use in the military operations of UAV squadrons in sensitive, high-risk areas is extremely important. For instance, the employment of autonomous drone swarms in military operations raises concerns about accountability and transparency in the decision-making process, the protection of human rights, and the risk of civilian casualties [10,11]. Researchers can contribute to the creation of a more safe, open, and moral drone swarm system whose advantages will be distributed in a just and equitable way by finding solutions to these problems. In order to shape the future of these technologies and their impact on society, as well as address some of the most pressing problems facing us today, research into the uses of blockchain technology in swarms of drones is, therefore, of utmost importance.

The purpose of this work is to provide a system that is efficient and resistant to outside attacks, where drones use blockchain technology to exchange data and coordinate their movements in a predetermined formation specified by the swarm's leader (mission-based). In order to mimic the behavior of the swarm, open source tools, including Ganache, Gazebo, PX4 models, and the MAVSDK library, were utilized. In addition, a smart contract was created that automates the process of choosing a specific position in the formation of the swarm members. A decentralized, secure communication network can be created thanks to this technology, which has the potential to improve network performance and member confidence.

This study contributes to advancements in knowledge in the following areas. Firstly, a comprehensive discussion has been undertaken to elucidate the significance of integrating blockchain technology within UAV clusters, thereby facilitating accelerated progress across multiple domains. Secondly, a novel and secure scheme has been introduced, featuring a leader-election mechanism that leverages drones as blockchain nodes. This innovation enhances the operational robustness of the system. Additionally, an innovative smart contract has been developed to enable decentralized decision making pertaining to formation in UAV swarms, contingent upon mission requirements. This approach deviates from the existing literature, offering unique insights. Lastly, the design of the proposed smart contract focuses on efficiency, resulting in a lightweight and easily deployable solution. This optimization streamlines the practical implementation of the smart contract within the UAV clusters.

The results are highly encouraging and it is certain that they could considerably increase the security of such systems. However, there are still problems that need to be resolved. Among them are the short battery life, price, scalability, storage, and ethical

issues. Researchers are urged to work on these in an effort to develop a system that could be widely adopted by many industries in the future.

This paper's structure is as follows. In Section 2, works related to this topic are discussed. Section 3 presents a thorough examination of some of the key theoretical ideas underlying blockchain technologies and their applications in UAV swarms. In Section 4, an in-depth review of the proposed smart contract, along with the tools that are used, is conducted. The results of the proposed scheme regarding the simulation and the functionality of this model are presented in Section 5. Finally, in Section 6, a discussion of the proposed model is provided, along with the limitations and the future work that researchers can undertake in order to improve the proposed scheme.

2. Related Works

The concept of leveraging blockchain technology to address problems with identification, security, and storage in UAV clusters is not new. There have been studies that have used novel methods to examine how we can use the benefits of blockchain technology to lessen these issues.

A more theoretical prospect of blockchain applications in UAV swarms can be found in [12–14]. A detailed explanation of how blockchain technology can be applied to drone clusters through various scenarios and examples is provided. Also, a thorough examination of the advantages of applying blockchain technology to these systems is conducted. Moreover, the potential problems and limitations of such applications are described. Works with specific results and contributions related to the technical aspects of this area are presented below.

For example, in [15], a lightweight blockchain is proposed, along with an interesting mechanism for message routing between UAV nodes to enhance the security of routing in 5G NR cellular networks. In addition, a way of identifying and managing malicious nodes is included. Proof of Traffic (PoT) is utilized for reaching a consensus, while the synchronization of the updated blocks is performed passively within the energy consumption limits. As it is based on the results of this work, the proposed scheme is very efficient and capable of ensuring the smooth operation of the swarm, even if there are malicious intruders inside it.

A novel approach in which a UAV swarm is able to conduct surveillance missions at specific points of interest (POIs) by utilizing blockchain technology is proposed in [16]. It is essentially a tangled-based model (IOTA), which relies on a decision-making strategy (smart contract) for the coordination of UAVs. The described idea is based on autonomous swarm operation (without the existence of some control center), in contrast to most of the published research. The blockchain is embedded in the UAVs, while there are some more nodes on the ground that have specific functionalities (route planning, financial transactions). The above-mentioned research is the only one that shares some components (smart contracts and autonomous decision making) with our suggested scheme.

The UAV-TIEN system is a blockchain-based data transmission model proposed in [17]. This specific scheme is secure due to the nature of blockchain technology and also robust against line-of-sight blocking problems. Monte Carlo simulations were carried out in order to evaluate the efficiency of the proposed model. The results were very promising due to the model's ability to increase the data broadcasting range. Scalability-related issues may arise due to the broadcasting mechanism that was used in the study (broadcasting to every UAV inside the range). However, by selecting a different mechanism and utilizing some ground station nodes, this problem can be solved. Also, by optimizing parameters such as the maximum blockchain length (MBL) and broadcasting frequency (BF), channel congestion can be improved.

In order to ensure security in UAV networks during surveillance and identify potentially compromised UAVs based on trust policies, [18] suggests a method that utilizes blockchain principles. The method allows for the accurate detection of false data when an official UAV is compromised. To validate the suggested strategy, ABS-SecurityUAV,

a unique agent-based simulator, was employed. In the conducted tests, the majority of the UAVs were able to confirm information about a person moving through a regulated area, and none of them confirmed false information from a UAV that had been hijacked.

In [19], a blockchain-based data acquisition procedure is shown in which data are collected from loTs utilizing a UAV as a relay and securely stored in a blockchain on an MEC (Mobile Edge Computing) server. Data are encrypted using the proposed technique before being sent to the MEC server with the help of a UAV. The public key of the UAV is used to execute encryption, while the private key of the loID is used to establish a signature when transferring data from loIDs. After decrypting the data, the UAV checks the identity of the loID using an η-hash bloom filter. The MEC server verifies the data and the sender's identity after receiving it. After successful validation and after receiving approval from the validators, the data are stored in the blockchain. A security study is conducted to demonstrate the viability of the suggested secure solution. The effectiveness of the suggested strategy is assessed using MATLAB R2023b simulations and practical applications, with some extremely encouraging outcomes.

Only a few research articles on blockchains for automated decision making and control in drone swarms have been published; hence, the goal of the literature review is to demonstrate that there is a gap in the literature that this work covers. Additionally, very few studies have been conducted to assess the functionalities of proposed models through real-world tests and simulations. Also, the majority of relevant publications have overlooked the importance of the leader's security in such systems. In order to address this gap, an innovative, lightweight, and easy-to-deploy smart contract is proposed in this work, in which a new leader-election mechanism is introduced. A realistic simulation that combines blockchain technology and actual drone programming tools was used to test this work's findings. The simulation results were very promising, and this work could potentially be used as a starting point to create state-of-the-art drone systems with enhanced security and an improved decision-making process.

3. Background

3.1. Blockchains

The foundation of blockchain technology is the concept of a digitally decentralized system that enables fully transparent transactions to be carried out without the need for a central intermediary. Although it was initially designed for financial transactions and the use of cryptocurrencies, it has since gained popularity across a wide range of industries because of its capacity to enhance system security, performance, and transparency [20].

Blockchains consist of blocks (Figure 1), which are a data structure that encapsulates all the necessary data on a specific transaction. Their structure can be divided into two parts: the header and the body. The header contains the hash value of the previous block, the hash value of the Merkle tree (hash binary tree) root, a timestamp (which defines the creation of the block), and a nonce (a number that miners must find in order to match the exact hash pattern of a particular blockchain). The body part contains a transaction list and a counter [21].

From the standpoint of the network, blockchains are typically peer-to-peer (P2P) systems in which a transaction is broadcast to all nodes when it has been created. Every new transaction is signed using the creator's private key and can be validated by all other nodes using that node's public key. This feature of blockchain technology is quite useful since it enables UAV swarms to withstand unauthorized access attempts. Every transaction conducted by an intruder using a key that the swarm determines is invalid will not be recognized as genuine. It is actually a type of authentication mechanism that blockchains can offer. Also, secure communication can be established in the same channel by using public and private keys to encode and decode the data, respectively. Specifically, data can be encoded by the sender drone using the receiver's public key. Next, the receiver drone can decode the data using its private key. In this way, data theft by a threat actor can be avoided (Figure 2).

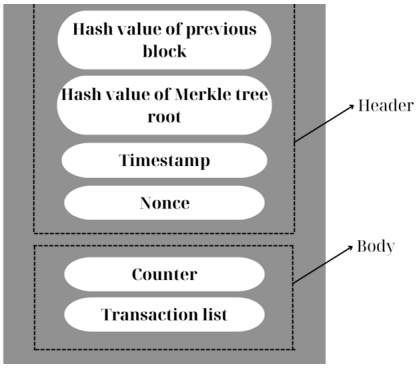

Figure 1. Typical block structure.

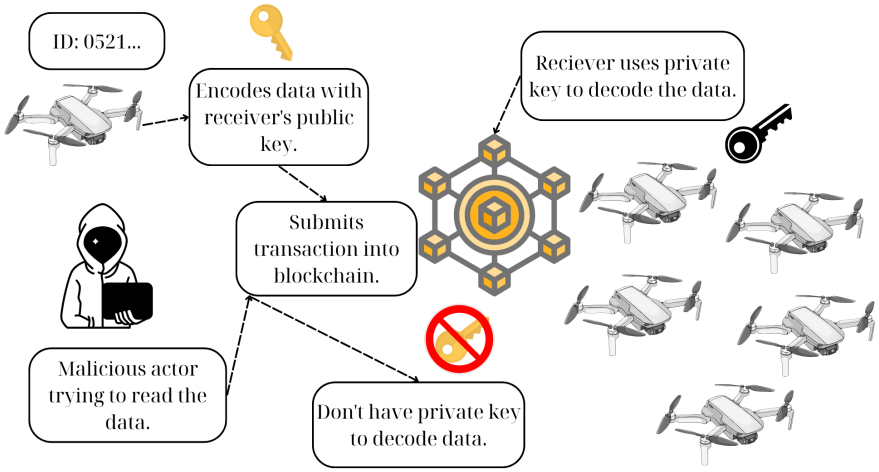

Figure 2. Data theft attacks can be avoided due to data encoding.

Another key feature of blockchains is the consensus mechanism [22]. Blockchains' decentralized nature necessitates a system so that all participants can agree on the legitimacy and order of the transactions. Therefore, there is no need for a central authority to be responsible for such actions. PoW (Proof of Work) is the most well-known algorithm miners use to try to determine the nonce in order to achieve a specific pattern in the block's hash (begins with four zeros, for example). Typically, brute-force techniques are used for this, which are highly computationally expensive. PoS (Proof of Stake), PoA (Proof of Authority), and PBFT (Practical Byzantine Fault Tolerance) are more deterministic, efficient, and environmentally friendly algorithms. This is the main reason why most blockchains tend to adopt these mechanisms instead of PoW.

The most practical technology in the context of blockchains is undoubtedly smart contracts. An automated transaction procedure can be created with just a few lines of code. When the parties reach an agreement, the transaction is signed and sent for verification over the network. The most well-known blockchain platform that supports the execution of smart contracts is Ethereum. For the creation of smart contracts on Ethereum, a Turing-complete language called Solidity has been created.

Based on the layered architecture (Figure 3) of blockchains, which is proposed in [23], this work focuses on the contract and application layers, as we can see in the following figure.

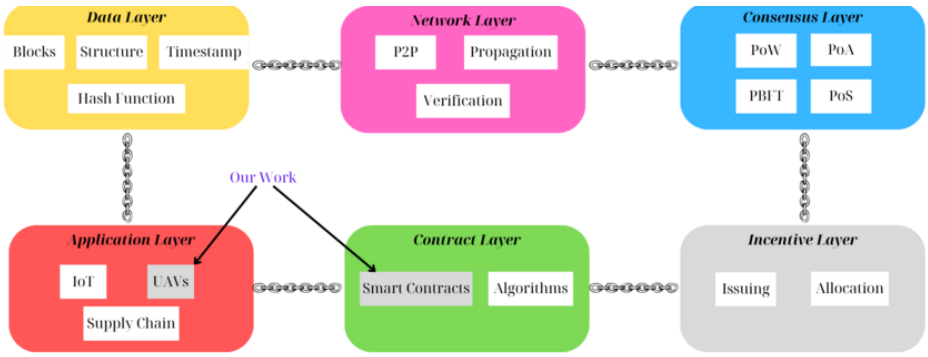

Figure 3. Layered structure of the blockchain architecture. This research focuses on the application and contract layers.

3.2. Motivation

The combination of blockchain technology and UAV swarms has the potential to significantly advance and address many problems in a variety of sectors, including the defense industry (Figure 4). Some of the greatest benefits of this concept are data integrity and security, transparent and controlled activities, autonomous decision making, and trust [24].

For drone swarms using blockchain technology, data integrity is crucial to ensuring the dependability and efficiency of cooperative operations. In order to prevent unwanted manipulation, the blockchain's immutability ensures that data generated by individual drones, including sensor readings, GPS locations, and mission status, are securely and permanently recorded on the blockchain. The swarm's data are further validated and synchronized via consensus methods, and data verification enables drones to cross-reference data with the blockchain to find errors or unlawful changes. The confidentiality and integrity of data during transmission are guaranteed by secure communication channels and data encryption. Timestamping provides accurate data synchronization, and smart contracts can be used to automate rule enforcement and data integrity checks. Additionally, the identity management and authentication capabilities of blockchains aid in validating the reliability of data sources, and the option of using public or private blockchains allows for customizing the level of transparency and privacy, depending on the particular use case. These factors work together to guarantee the reliability, security, and accuracy of the data created and shared inside the drone swarm, enabling cooperative and reliable operations for uses including surveillance, search and rescue, and more [25,26].

In the context of drone swarms, log files [27] offer a variety of advantages, chief among which are their immutability and ability to produce traceable recordings of all actions taken by the drones in the swarm. These log files provide data recordings that are permanent and unalterable, protecting the integrity of the historical record when used in conjunction with blockchain technology or other tamper-resistant systems. For post-mission analysis, assuring responsibility, and confirming compliance with established procedures and regulatory standards, this immutability is essential. Each drone's actions and events, including takeoff and flight trajectories, sensor readings, and landing procedures, are painstakingly recorded and include cryptographic timestamps. This leaves a clear trail that makes it possible to recreate events and mission-related decisions with accuracy. This is especially useful for incident analysis and the development of improved safety measures. In addition to encouraging accountability, these traceable records make it easier to analyze data for optimization and provide insights into operational effectiveness and opportunities for development based on past performance data. Additionally, access to these log files can be strictly managed by blockchain-based authentication and access control, guaranteeing that only authorized users can access and examine the records and enhancing data security and privacy [28].

Moreover, by utilizing the data gathered by their IoT sensors and integrating blockchain technology for safe data exchange among drones in a swarm, the decision making, coor-

dination, and optimization of collaborative operations are all significantly enhanced. A blockchain guarantees that crucial data are captured and shared among all swarm members in a transparent and tamper-proof manner, including drone locations, battery status, and environmental conditions. These shared data enable more accurate and well-informed decision making, allowing drones to successfully adapt to changing circumstances. The collected data also provide useful inputs for optimization algorithms that improve swarm synchronization, path planning, and obstacle avoidance. Real-time updates and security features ensure that the data are up to date and protected against manipulation, improving trust and enabling adaptive reactions to changing situations.

By utilizing smart contracts to set and enforce rules and behaviors, the use of blockchain technology in drone swarms promotes trust among swarm members. These self-executing contracts give each drone in the swarm a defined set of guidelines for behavior and regulating activities like coordination, data sharing, and navigation. This method's transparency and decentralization, which allow all swarm members to access and independently check the same set of rules without the need for a centralized authority, make it particularly effective at fostering confidence. Consensus techniques provide community validation and agreement on these norms, lowering the possibility of malevolent behavior from individual drones. In situations like search and rescue, surveillance, or emergency reactions, expedited cooperation, predictability, and unanimity in observing defined norms are crucial for effective joint actions that eventually result in the accomplishment of a mission. Additionally, the blockchain's immutability protects the integrity of the rules, fostering a secure environment within the swarm.

In situations involving drones from multiple parties, such as defense and public safety, during crisis management, blockchain technology can play a crucial role in improving coordination and communication between various government agencies. By immutably storing drone-collected data on the blockchain, it guarantees data security and integrity, rendering it trustworthy and tamper-proof. Smart contracts streamline drone operations by managing access control, designating no-fly zones, and standardizing communication protocols. The blockchain's shared platform enables interoperability among drones made by different manufacturers, and supply chain management gains from tracking drone availability, maintenance, and location, with smart contracts handling deployment and tracking [29]. Decentralized identity storage and access control are provided, streamlining identity and access management. All drone-related operations are transparent and accountable thanks to immutable audit trails, and security and privacy are guaranteed through fine-grained data sharing. In addition to facilitating public–private partnerships, regulatory compliance, and partnership management, the blockchain's cryptographic security is essential for safeguarding sensitive government operations. As a result, the effectiveness and accountability of drone operations in crisis management are maximized.

Lastly, blockchain technology has the potential to significantly improve the ability of aviation authorities to enforce drone laws, including the use of geo-fencing [30]. Aviation authorities may provide a safe and open platform for monitoring and enforcing airspace limitations by integrating blockchains into drone management systems. By comparing real-time drone GPS data with pre-defined geographic boundaries stored on the blockchain, smart contracts can be used to autonomously enforce no-fly zones and height restrictions, making sure that drones do not trespass into prohibited areas. In addition to improving safety, this blockchain-based solution offers an unchangeable record of compliance, allowing aviation authorities to carry out regulatory audits, investigations, and post-incident analysis. Additionally, it enables the dynamic updating of airspace limits through the seamless integration of real-time data from numerous sources, such as weather and air traffic data. By supporting responsible drone use and reducing security threats in the airspace, the blockchain's decentralized and cryptographic security features secure data integrity and improve confidence between aviation authorities and drone operators.

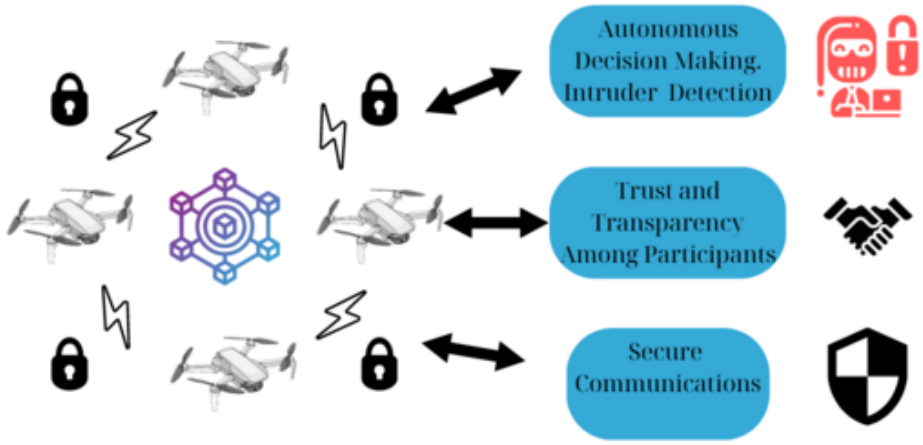

Figure 4. Advantages of using blockchain technology in UAV clusters.

4. Materials and Methods

4.1. Proposed Scheme

The scheme that is studied in this work follows the leader–follower strategy. The leader (only) can register (or delete) drones in the swarm based on their addresses (Listing 1). If drones have not been registered by the leader, they cannot use any of the smart contract's functionality. Drone registration is a method of authorization in order to enhance the security of the proposed system.

Listing 1. Drone registration function.

```
function addDrone(address drone) public onlyLeader {
    drones[drone] = true;
    droneCount++;
    emit DroneAdded(drone);
}
```

This specific function can be called only by the swarm's leader; therefore, the `onlyLeader` modifier is used to prevent access from the other members. All drone addresses are stored in the `drones[drone]` mapping, along with a Boolean variable that is set to true if the drones have been successfully registered in the swarm. This function also has an opposite function called `removeDrone`, with the main difference being that the Boolean value is changed to false. The opposite function can be called by the leader, for instance, if a voting procedure is conducted and the majority of the swarm members decide that a specific drone has been compromised. In this way, the malicious drone can be stopped from using the smart contract's features. Therefore, the hacker will not be able to obstruct the task that the rest of the drones have to finish. The challenge of creating such a mechanism still needs to be addressed in future research.

Subsequently, a new transaction can be started by the leader utilizing the `submitData` function to submit data along with its location (Listing 2). The swarm's followers can also utilize this specific function. For instance, location or other information that their sensors may have gathered can be submitted to the blockchain by the members of the swarm using this method.

Listing 2. Data submission function.

```
function submitData(string memory _location, string memory_data) public {
    require(drones[msg.sender], "Error: A drone must be in the
    swarm in order to submit data.");
    droneData.push(DataCollectedByDrone(block.timestamp, _location, _data
, msg.sender));
}
```

Once these tasks are finished, the leader can submit the mission information to the blockchain to generate a new transaction. The mission is identified by its name, a distinctive identifier (`id`) that sets it apart from other missions; its type (`missionType`); suitable formation (`formationType`); and a Boolean parameter that indicates whether the particular mission is operational (Listing 3). A specific type of mission (dive and attack, search and rescue, city surveillance) can be selected by the leader in the proposed smart contract. These missions are illustrative, though it would not be difficult to add to or modify the list. A mission can be created by the leader using the following method:

Listing 3. Mission creation function.

```
function createMission(string memory name, MissionType missionType,
FormationType formationType) public onlyLeader {
    uint16 missionId = missionCount;
    missions[missionId] = Mission(missionId, name,
    missionType, formationType, false);
    missionCount++;
    emit MissionCreated(missionId);
}
```

A mapping `missions[missionId]` is used, which takes as a key the ID of each mission in order to identify it. One can see that initially, the mission is declared inactive (a false Boolean) during its definition. This is so it can be activated or deactivated whenever needed by the leader (Listing 4). In this way, there is better synchronization in cases where several missions have been declared. The formation that the swarm must follow depends on the type of mission assigned by the leader (`Search, DiveAttack, CitySurveillance`) and is determined automatically (Ring, V, and Line, respectively). Activation of the declared mission is carried out using the following function:

Listing 4. Mission activation function.

```
function activateMission(uint16 missionId) public onlyLeader {
    require(missions[missionId].id == missionId, "No such mission");
    missions[missionId].active = true;
    emit MissionActivated(missionId);
}
```

In the suggested smart contract, a mechanism for confirming that the leader is alive has also been included. For the followers to be certain that their leader is still active, the leader must send a heartbeat message that can be read by them. An indicative threshold of 180 s has been used, even though this can be easily modified. If the leader does not report his status, a new leader-election process will follow. The outcome of this process will favor the drone with the highest battery life. Leader's functionality flowchart can be seen below (Figure 5).

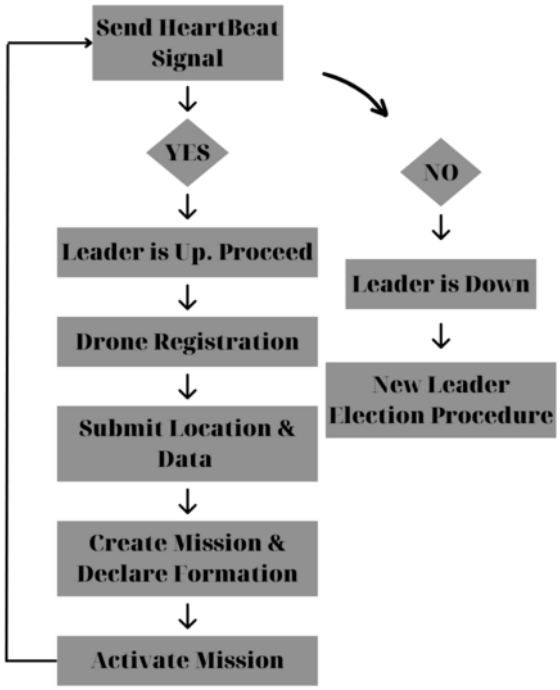

Figure 5. Leader's functionality flowchart.

As for the followers, the first thing they do is make sure their leader is still active.

Followers can inspect whether a heartbeat message has been sent by the leader by invoking the smart contract's `checkLeaderStatus` method (Listing 5). If not, they go through a process of choosing a new leader. The drone with the highest battery life is chosen as the new leader of the swarm, presuming that all of the drones in the swarm are of the same type (so there are no connectivity, design, or other issues). Drones can use the following function to periodically report their battery levels (Listing 6):

Listing 5. Checking the leader's status.

```
function checkLeaderStatus() public {
    if (block.timestamp - lastHeartbeat > heartbeatTimeout){
        leaderIsAlive = false;
        if (msg.sender != leader) {
            electNewLeader();
            }
        }
    }
```

Listing 6. Battery level submission for followers.

```
function submitBatteryLevel(uint8 batteryLevel) public {
    require(drones[msg.sender], "Only drones can submit battery level.");
    batteryLevels[msg.sender] = batteryLevel;
}
```

The aforementioned procedure does not take place when the leader is active. The mission and the declared formation can now be checked by followers by reading the data that their leader has submitted. Knowing the position of their leader, they can select the location of the formation that is closest to them. A formula is required to translate angles and distances into coordinates in order to create a formation scheme. The Haversine equation is the ideal solution for this problem. If the coordinates of two points are $(\phi_1, \lambda_1), (\phi_2, \lambda_2)$

(latitude, longitude), the central angle θ between the two points can be determined using the equation:

$$hav(\theta) = hav(\phi_2 - \phi_1) + cos(\phi_1)cos(\phi_2)hav(\lambda_2 - \lambda_1), \quad (1)$$

where hav is the Haversine function $hav(\theta) = sin^2(\frac{\theta}{2})$.

Subsequently, the distance between the two points along the geodesic can be calculated using the formula $d = r\theta$, where r is the earth's radius. The desired coordinates for the formation can be determined by following the same approach in reverse. For example, if the desired formation is V, for angles $\theta \in \{-30, 30\}$ and distances $d = 10$ m between drones, the inverse Haversine formula calculates the desired coordinates.

This information can be used by the drones to determine the location that is closest to them (through a loop). Assume that they are located at (ϕ_d, λ_d). If the coordinates determined using the Haversine formula are (ϕ_i, λ_i), by using the Euclidean distance equation

$$r_i = \sqrt{(\lambda_d - \lambda_i)^2 + (\phi_d - \phi_i)^2}, \quad (2)$$

for all of them, the position nearest to them can be calculated. First, the available positions in the formation are examined by the drones through the blockchain. If the position is free, a submission process will occur through which the position will be submitted to the blockchain. Next, this exact location will be occupied by the swarm member who selects it. Otherwise, if the desired position is not available, the next nearest position is selected, and so on.

In the illustration above (Figure 6), the leader's coordinates are $(0, 0)$. In the V formation, the leader will, therefore, be in position 0. Consider a follower (i) with the coordinates $(x_j, y_j) = (2.5, 8)$. Based on Equations (1) and (2), the follower determines all r_{jp} Euclidean distances for positions 1, 2, 3, 4, 5, and 6. From these, he selects the position that is nearest to him, in this case, position 4, and moves there. Repeating this procedure allows each follower in the swarm to communicate its position to the blockchain, where it is visible to all other followers. If this position is already taken, the next one closest to it is chosen (in this case, position 5). This strategy (submitting chosen positions to the blockchain and determining their availability) avoids situations where the same position is chosen by two (or more) separate drones, which may result in collisions.

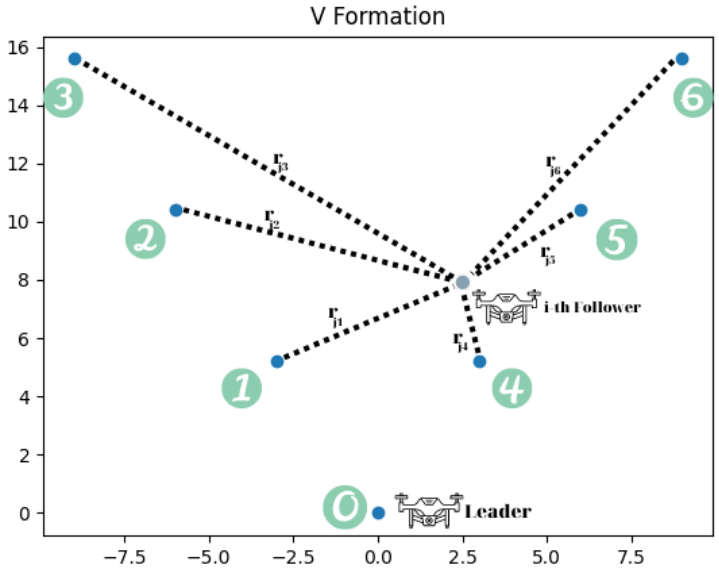

Figure 6. Position selection. Each number corresponds to a specific position in the vee formation. Position 0 has been occupied by the leader. The i_{th} follower selects position 4 in the formation. If position 4 is taken, it selects position 5, and so on.

One can see the position submission utilized by the drones to disclose their positions through the smart contract in the aforementioned function (Listing 7). Also, as previously stated, specific measures have been taken in order to ensure that the position submitted to the blockchain is valid and not occupied. In any other case, the transaction will be reverted. A typical flowchart for follower functionalities is shown below (Figure 7).

Listing 7. Position-selection function.

```
function assignPosition(uint8 position) public {
    require(drones[msg.sender], "Only drones can assign positions");
    require(position > 0 && position < droneCount, "Invalid position");
    require(positions[msg.sender] == 0, "Position already assigned");
    require(positionsTaken[position] == address(0), "Position taken");
    positions[msg.sender] = position;
    positionsTaken[position] = msg.sender;
    emit PositionAssigned(msg.sender, position);
}
```

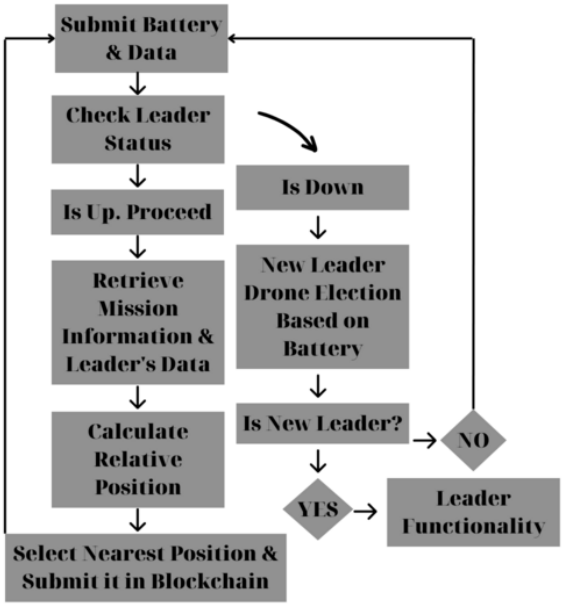

Figure 7. Follower's functionality flowchart.

Although the suggested smart contract is only defined for use in the military domain, it can still be updated and applied to other sectors as well. The basic concept behind this model is that each smart contract will only be effective for a set amount of time before becoming obsolete and needing to be reprogrammed in accordance with the new task that the drones must complete. In the matter of unplanned situations during a mission where the drones must operate differently, the `deactivateMission` function can be used in order to abort the pre-defined mission.

4.2. Tools Deployed for the Experimental Setup

All the tools used to test the functionality of the proposed smart contract are open source and are described briefly below.

Ganache (https://trufflesuite.com/ganache/, accessed on 20 July 2023) is a tool for the creation of local blockchain networks that helps in the development and testing of smart contracts. It can be used as an Ethereum blockchain alternative and can be deployed quickly, easily, and locally in the developer's test machine. It is lightweight and offers a variety of options that can be set according to the developer's needs (changing, for example, the block creation time). By adjusting these parameters, one can achieve a more realistic simulation.

Gazebo (https://gazebosim.org/home, accessed on 20 July 2023) is a 3D robotics simulator. Its accuracy regarding real-world physics makes it the best option for a developer

who wants to test a robotic system without risking breaking real hardware. Additionally, the library of tools it provides (sensors, lidars, cameras, indoor and outdoor world creation) makes the simulation easier and more realistic. The use of Gazebo for this work can provide us with a visual result of how the UAVs would coordinate in a real-life scenario.

PX4 (https://docs.px4.io/v1.12/en/simulation/gazebo.html, accessed on 20 July 2023) models were used for achieving a simulation of a real iris quadrotor drone in Gazebo for the needs of this work. Also, the MAVSDK (https://mavsdk.mavlink.io/main/en/python/quickstart.html, accessed on 20 July 2023) Python library was used to control the behavior of the swarm and coordinate their actions. This particular library is typically used in real-world drone programming applications.

In addition to these tools, Python was used for the development of the DApp, including the web3 module for interaction with the smart contract. A variety of options are available for drones through the DApp, such as reading data from the blockchain, submitting data, checking leader status, reading mission details, checking position availability in the formation, selecting and submitting the desired position, etc. (Figure 8).

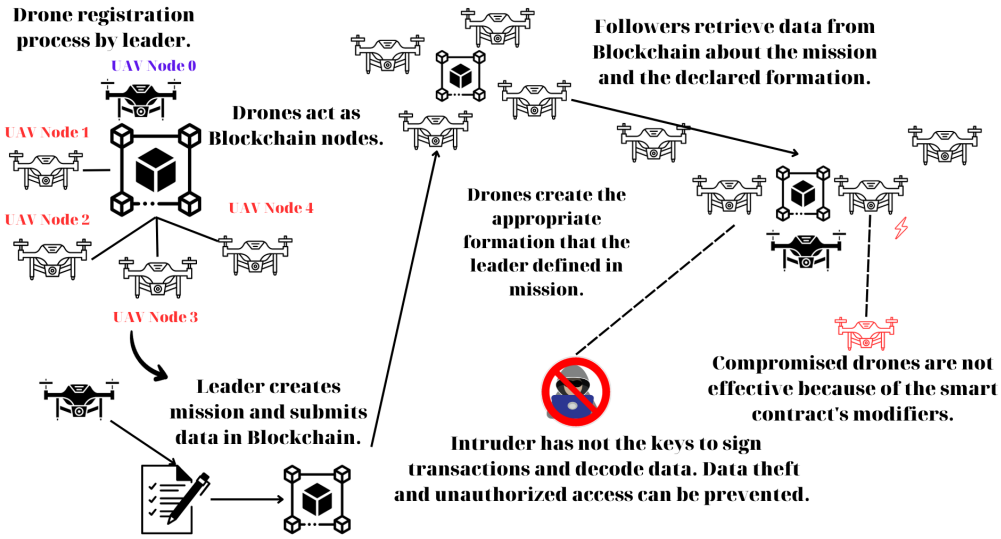

Figure 8. Proposed model.

5. Results

A thorough explanation of every outcome obtained via testing the suggested model and smart contract is provided in this section. A realistic simulation using the Gazebo and MAVSDK frameworks was tested, allowing for the evaluation of the model's functionality. This avoided the risk of testing the suggested system on actual hardware and allowed for a visual demonstration of its effectiveness in real-world circumstances.

5.1. Simulation

For testing and evaluating the functionalities of the smart contracts, a Python DApp has been developed using the web3 module. Also, for better interaction with the smart contract, a UI has been created.

Figure 9 shows the functionalities of the leader. As can be seen on the left-hand side of the figure, the leader first submitted a heartbeat message to the blockchain in order to confirm he was active. In addition, the menu with all the options available to the leader while interacting with the smart contract is shown below. By choosing option 1, the drone registration process in the swarm occurred.

On the right-hand side of the figure, the option of submitting data to the blockchain is shown. In this particular case, the leader submitted his location, along with the data "Hello from leader!". A mission creation process is also shown below. A dive-and-attack mission by the name of "Impossible" was created and submitted successfully to the blockchain.

Finally, the last image represents the activation of this particular mission with ID = 0 (the first mission created was ID = 0, the second was ID = 1, and so on).

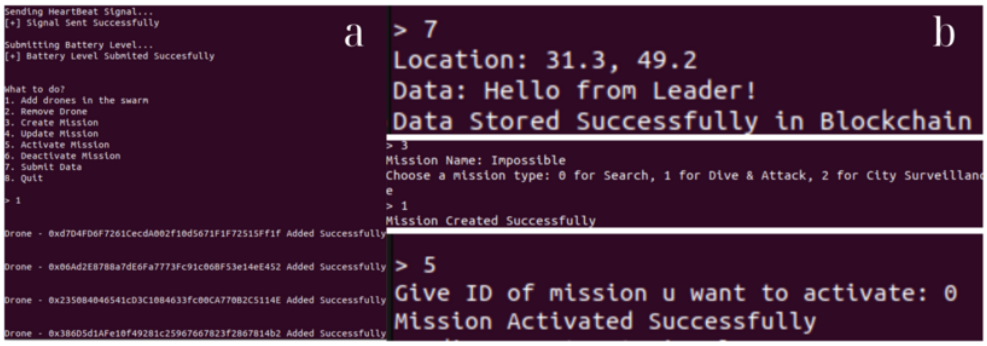

Figure 9. Testing leaders' functionalities in the smart contract: (**a**) On the left, the leader's menu, along with the drone registration process, can be seen. (**b**) On the right, the data submission, mission creation, and activation processes are shown.

Figure 10 illustrates the functionalities of the followers. As can be seen, the first action was to check whether their leader was active and submit their battery level. It is clear from the first terminal that the leader is alive, and it is definitely not the current drone with ID = 1. One can observe the procedure for reading data from the blockchain in the second terminal. The mission details, declared by the leader, were retrieved by the drone with ID = 1. These details included the mission's name ('Impossible' (string)), the mission's type (dive and attack, integer 1 in the smart contract), the formation that drones must make (V, integer 1 in the smart contract), and the Boolean value (true), indicating that the mission was active. After retrieving the mission's details, the nearest position in the formation could be selected by the drones. First, the available positions were examined through the blockchain (based on Figure 6, terminal 3), and then the best option for each of them was chosen and submitted to the chain (terminal 4).

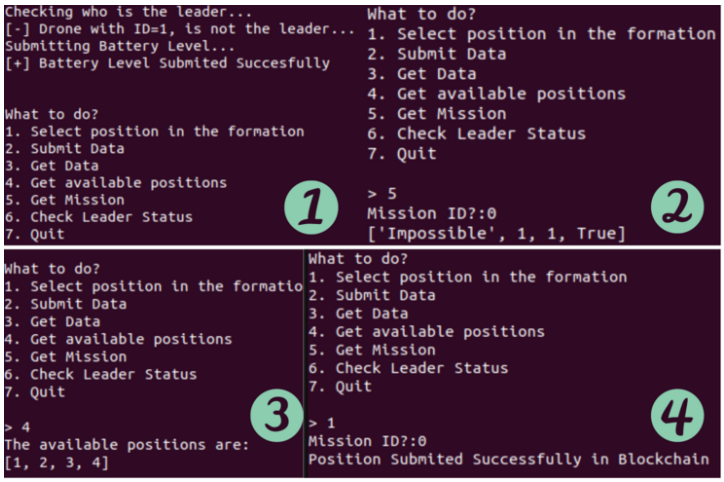

Figure 10. Testing followers' functionalities in the smart contract: (**1**) Follower's menu, along with the leader-checking and battery-submission processes. (**2**) Process of retrieving mission details for the followers. (**3**) Checking the available positions through the blockchain. (**4**) Position-selection process in the formation and position submission to the blockchain.

The process was repeated for the other followers. The outcomes of the position selection of the other drones are shown in Figure 11. Position 2 in the V formation was selected by the drone with ID = 1 since it was the one nearest to it. Then, the available locations were examined by the drone with ID = 2, and position 2 was selected. The remaining options were inspected by the drone with ID = 3 through the chain. Only positions 1 and 4 were

available, so position 4 was selected, and so on. For a drone that has already chosen a position, this step cannot be repeated because the blockchain would reverse the transaction.

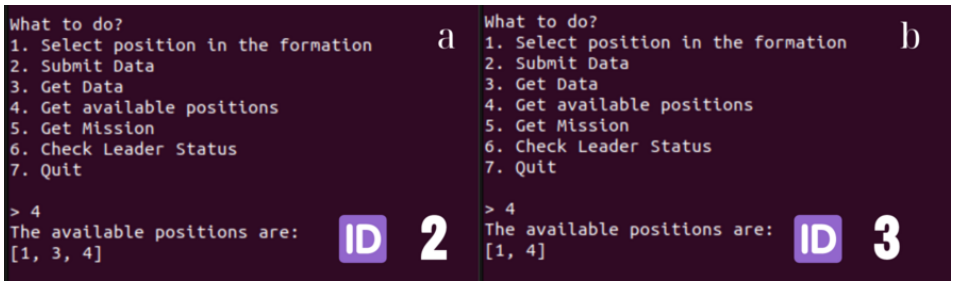

Figure 11. Follower's position-selection process: (**a**) In the left panel, the follower with ID = 2 is selected. (**b**) In the right panel, the follower with ID = 3 is selected.

Finally, the program was expected to surpass the predetermined threshold without taking any actions as the leader in order to test the new leader-election procedure. As can be seen in Figure 12, the new leader is the drone with ID = 1, as it was the one with the highest battery life, while the old leader was down. Also, it can be observed that the follower's menu has been replaced by the leader's menu.

The integration of the DApp with the tools covered in the previous section; PX4 iris quadcopter models (by Dronecode Foundation, San Francisco, CA, USA), Gazebo simulator software (version 11.0, by Open Robotics, Mountain View, CA, USA), and the MAVSDK Python library (version 1.4.14, by Dronecode Foundation, San Francisco, CA, USA); was not very difficult to achieve. Following the dive-and-attack mission created by the leader, followers started choosing their positions and moving behind the leader in the areas of the V formation that were closest to them (Figure 13). Note that in the proposed smart contract, a position-cleaning procedure has also been included in case the mission has been completed or aborted by the leader.

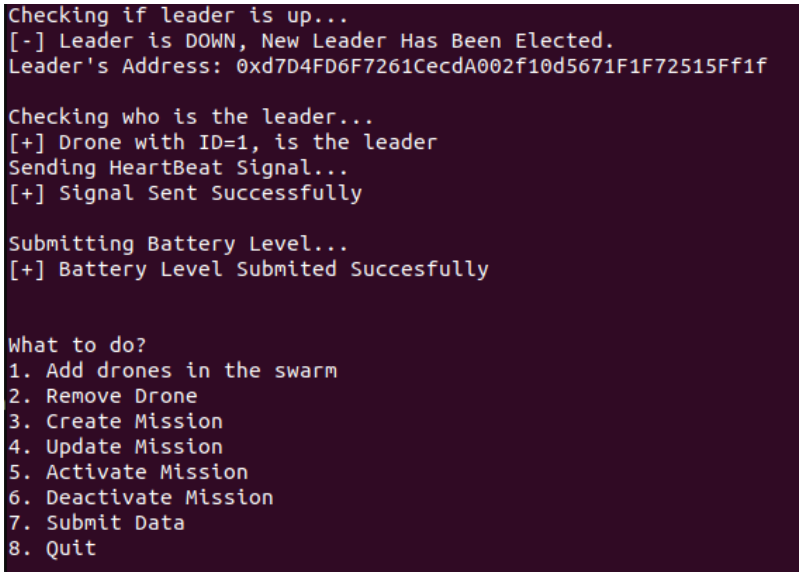

Figure 12. New leader election.

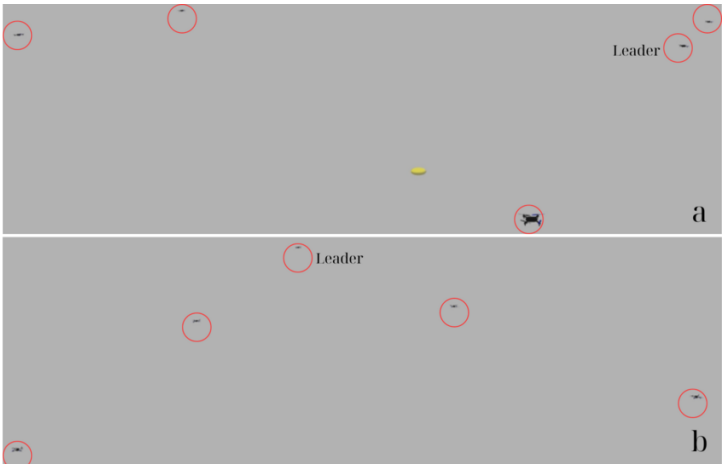

Figure 13. Gazebo simulation: Red circles were used to highlight the drones in the simulation. (**a**) Drones can be seen in random positions. (**b**) Drones forming a V formation following their leader after the dive-and-attack mission has been created.

A Ganache instance was utilized as a personal Ethereum blockchain network to test the proposed smart contact's capabilities (Figure 14). The leader from which the smart contract was deployed in the network is represented by the first address. Because of this, this account's overall ETH token waste was higher. Additionally, the number of transactions was higher for the leader because of the additional duties he was responsible for (drone registration, mission creation, etc.).

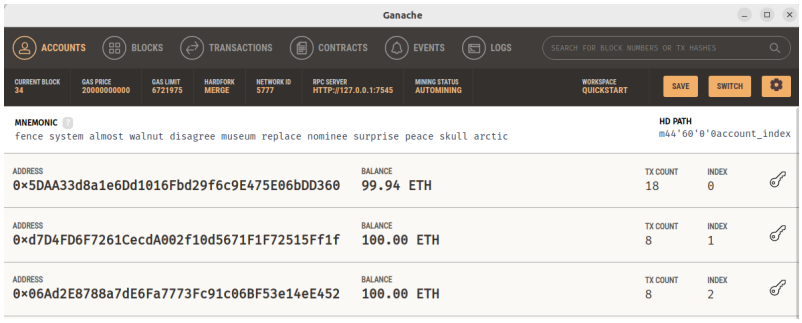

Figure 14. Ganache instance.

Also, in Figure 15, one can see the information of the first block where the smart contract was implemented in the Ganache instance.

```
AttributeDict({
   hash:"0x9bc8fbae27277d31deda74fa1fd24b1c119b5eb637364ef694192c088da2452d",
   parentHash:"HexBytes(""0x837b7d1cbd6adcc6bb9f1b541e5ed4576e8223db6e839cb37fb2077971920bbc",
   sha3Uncles:"HexBytes(""0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
   miner:"0x0000000000000000000000000000000000000000",
   stateRoot:"0xbabb0018c4b1df067521fc3fa92a626df0a12c74e8716ebfa338f411349e5a3d",
   transactionsRoot:"0x61e101ec12988ead97b1d5b3cfd702dca3d45968e564ac5c204627473054 4ac0",
   receiptsRoot:"0x62a286902eebd904ef6a7ee56c20138a4c7ba8281f744cf480bc0e7dafa0860b",
   logsBloom:"0x00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000...0",
   difficulty:0,
   number:1,
   gasLimit:6721975,
   gasUsed:2930836,
   timestamp:1696440794,
   extraData:"0x",
   mixHash:"0xe6c42469f6a77461b6852a75ce95311cb13a999bf8557038019284eeac88a3bf",
   nonce:"0x0000000000000000",
   totalDifficulty:0,
   baseFeePerGas:875000000,
   size:13736,
   transactions:[
      "0x7f37501060ac1e2ce7869e81318cb290b9fb053bc402cf6bf6b97f2818c4d3d1"
   ],
   uncles:[
   ]
})
```

Figure 15. Smart contract's block.

5.2. Evaluation

The proposed scheme was successfully implemented and performed as expected, without any problems. The smart contract's functions could be called by the leader and the followers in order to create transactions according to the swarm's needs. Therefore, regarding the smart contract's functionality, no issues needed to be resolved.

5.2.1. Computational Costs

In our scheme, many computing initiatives are combined to enable smooth coordination between drones and blockchain technology. While leader–follower communication involves data transmission and reception, the path planning of the leader drone necessitates processing resources for environmental data analysis and the best route selection. For each drone to participate in the network and use blockchain technology, it is necessary to run and maintain a blockchain node, implement smart contracts to automate swarm operations, and carry out data encryption and verification processes. Drones may participate in resource-intensive calculations or allocation processes, depending on the blockchain's consensus algorithm. Computing power is needed for the swarm's networking and communication, as well as for security measures, data processing, storage, and retrieval. For efficient operation, real-time data processing and resource monitoring are crucial.

Another level of computing complexity is added by addressing latency issues and enabling real-time decision making. The particular computational requirements vary depending on the complexity of the swarm, the blockchain platform, the number of drones, the frequency of interactions, and the tasks at hand, necessitating a delicate balance between the fundamental operations of the drones and their energy limitations for optimal performance.

5.2.2. Scalability

The scalability of these structures is another factor to consider. The addition of extra nodes to the network ensures that it is more robust and efficient against attacks. Nonetheless, a major drawback is the increased delays, since the consensus-reaching process of participants requires broadcasting the message throughout the network, thus performing more hops in this P2P structure. It is possible to avoid major scalability-related delays by using selected drones as nodes in the blockchain (rather than all of them). Another solution is to use tangled-based blockchain models, which tend to be more scalable. Also, the investigation of different consensus mechanisms could potentially lead to better outcomes.

The complexity of the architecture, along with the manner in which these systems are implemented, is also influenced by the increasing number of drones in the swarm. In a UAV swarm, the addition (or removal) of nodes, along with their continuous movement, location, and speed change, could lead to unstable communications. A real-world application of the suggested model needs to be implemented in order to test the proposed model and produce realistic findings. Further investigation into this matter remains an open question.

5.2.3. Security

The outcomes of the security evaluation of the proposed scheme are summarized below.

First, the follower registration mechanism works as an authorization process since drones that have not been registered by the leader in the blockchain do not have the right to call the functions of the smart contract and therefore access sensitive information (data that are collected by sensors, such as the location of drones). Therefore, the risk of a network intrusion is reduced.

Moreover, supposing that the private keys are safe, every transaction that occurs on the blockchain is unquestionably reliable. This is because they are verified for legitimacy by the other members using the matching public key after being signed with the drone's private key. Due to this functionality, data theft attempts are ineffective.

Also, the chance that the data stored in the blockchain could be altered is negligible due to the unchangeable and open nature of blockchain technology. The smart contract

itself has been recorded in a block of the chain, so there is no chance of alteration for the same reason. As a result, participants' sense of trust is boosted.

Furthermore, another key feature of the proposed system is its robustness against isolated faults. Due to the nature of blockchain technology, even if one or more drone nodes malfunction, the operation of the system as a whole will continue to function normally since there will always be some members that are working normally.

In addition, modifiers have been utilized in the smart contract to restrict followers' access to certain functions that belong exclusively to the leader. For instance, if a drone belonging to the followers is compromised, the threat actor will not be able to access the leader's sole function of registering new drones to the swarm (removing drones, creating missions, activating or deactivating missions, etc.).

Since the leader in this particular scenario behaves as a central authority, the leader–follower model violates the blockchain's decentralized nature, which is a security drawback. However, this concept can be readily adapted by giving the swarm more leaders. As a result, the concept of decentralization is revived, and the system's defenses against leader-related attacks are strengthened (since the proposed model is weak in this area).

Lastly, another drawback that has to be addressed is related to private key management. The possession of a private key could lead to the creation and signing of fault transactions, thus questioning the effectiveness and performance of the swarm. The development and integration with the proposed scheme of a mechanism for handling private keys could lead to a more reliable and secure model.

5.2.4. Costs

Creating and maintaining such a system will undoubtedly be expensive. In addition to the cost of purchasing drones, external devices are required, such as a Raspberry Pi running a Linux-based OS so that the drone can behave as an autonomous node in the blockchain while also being able to execute code according to the researcher's preferences, SSD disks as a storage medium for the data gathered by the sensors and the recording of the blockchain data, and a flight controller board that will run the drone's flight software (PX4, Axon, etc.). Also, flight-related sensors are required, including GPS, cameras, and lidars for obstacle avoidance. Depending on the nature of the operation, additional cargo may be needed (infrared cameras, night cameras, etc.).

It is not an easy process for anyone to install, integrate, and maintain these technologies; hence, some money should be invested in staff training. Naturally, such an endeavor would involve a sizable time commitment; therefore, new positions may be required to hire experts with specific knowledge of these systems. On the one hand, this increases the total budget's expenses, but on the other hand, it undoubtedly opens up new career opportunities for scientists who pursue studies in these domains of technology.

6. Discussion

UAV cluster technology has recently gained significant attention, especially in the defense industry, due to its ability to perform a variety of tasks (surveillance, rescue, monitoring, and attacking), without the need to expose military troops to harsh environments. The idea of integrating blockchain technology into these systems in order to enhance security is not new. However, a novel scheme is proposed in the context of this work.

To the best of our knowledge, the idea of using blockchain technology for mission creation and formation control in such systems has so far not been considered in the literature. The method examined in this paper is unique, at least in terms of the published works, as indicated by the lack of available smart contracts with this capability. In contrast to the findings of related studies, the proposed smart contract can be used for decentralized decision making in relation to missions, the formation of the swarm, and the election of a new leader when necessary. Since drones are, as already indicated, vehicles with constrained capabilities, the purpose of such Internet of Things applications is not to

necessitate complicated and expensive calculations. Because of this, the smart contract's design was made to be as lightweight and user-friendly as possible.

The proposed model's functionality and security were tested and simulated, and the results were highly encouraging. The presented simulation provides us with a good idea of how these systems would operate under actual circumstances. Such efforts will undoubtedly speed up the development of these technologies and address a number of security-related problems. However, there are still some restrictions that we should be aware of, which we outline below.

6.1. Limitations

Some of the limitations of such systems in real-world scenarios include battery life (drones and sensors), overall cost, scalability-related issues, and storage issues. Notably, a major problem with these devices is their battery capacity. The main factor in the successful completion of a mission by drones is their lifespan. Large battery capacities are not a solution to this issue, as they increase the weight of the battery and, consequently, the energy required by the drone to lift that weight [29]. Finding the golden ratio is, therefore, necessary.

Additionally, sophisticated algorithms [31], can aid in optimizing drones' battery usage. Of course, in this case, a powerful yet efficient algorithm is needed. Utilizing the many recharge stations across a specific area is an additional strategy [32]. Drones will, therefore, be able to land at the stations as needed and recharge their batteries without having to return to the base. Finally, using panels is yet another suggestion [33]. The drone will have to carry more weight, which will make flying more challenging, but there is also a sustainable energy source. This strategy is less useful for missions that take place at night or without much sunlight. The employment of potentially hybrid solutions, such as combination panels with backup batteries or relatively large-capacity batteries paired with optimization algorithms for more efficient power distribution, is more efficient. However, no precise solution to this problem has yet been discovered.

As for the cost, the price of putting such a plan into action could be high. The price of computer boards, communication devices, networks, and sensors is in addition to the price of drones. Furthermore, a drone must be adequate in terms of size, design, and material construction (it cannot be a small, low-cost vehicle) in order to be efficient in the sense that it is stable in flight, has a long lifespan, and can carry a sufficient amount of payload. As a result, the financial requirements significantly increase. Moreover, the need for experts in this area in order to build and maintain such complicated systems could potentially significantly increase the overall cost.

Scalability is a problem that plagues all blockchain-related apps. Adding more drones to the swarm makes it more resistant to attacks or single points of failure, but on the other hand, it adds to the time it takes for the nodes to reach a consensus and secure transactions. This might necessitate more computational power, which is not always acceptable. Additionally, as the number of drones in swarms grows, so does the complexity of such systems' implementations and architectures.

Finally, the transactions that take place must be replicated by each node in the blockchain. Therefore, sufficient storage space is needed, especially in dense clusters where drones would conduct a lot of transactions while flying. Pruning methods, IPFS, and data optimization can all be used to address this issue.

6.2. Ethical Considerations

Despite the significance and seriousness of such systems, particularly in military and defense activities, a number of ethical issues arise in relation to the use of such autonomous robotic swarms [34–36]. The first major issue is related to target discrimination. Such systems often fail to distinguish between an enemy target and an ally (or civilian), leading to an increased likelihood of accidental losses due to the increased number of aircraft in these clusters. This risk of collateral losses could be decreased by carefully and effectively

programming the drones to distinguish between their targets. In addition, in combat situations, the level of force that is employed must be proportional to the target. Different tactics should be used for a military base as opposed to an isolated target.

Another critical issue is about taking responsibility for autonomous drone decisions. Nowadays, the majority of these types of schemes are managed by people via a control center, and the operator takes responsibility for poor judgment and bad actions. Something like this would be challenging for automated systems (it is impossible to hold an autonomous drone responsible for its actions). Therefore, there must be specific rules to ensure that the "operators" (those who program, design, and maintain these systems) are held accountable for the decisions they make.

Additionally, security and data privacy within these systems are definitely something to be concerned about. Drones are susceptible to malicious attacks, just like any other IoT device. The mission could end in disaster if successful attempts are made against them and the sensors they are carrying. Information and data leaks, illegal use of force against unarmed civilians, and other incidents can have a severe impact on the mission. Therefore, it is crucial to include strong security measures and encryption techniques to protect data integrity and lower the likelihood of harmful behavior.

6.3. Future Work

The insights and findings presented in the current research study open up avenues for potential future work on blockchain applications for secure communication and control in drone swarms. Below, we outline some of these potential paths for future research and suggest other potential research directions that could be pursued to enhance and extend the current study's findings.

One area of future work is to apply the results of this work to real-world implementations. The functionality of the smart contract has been developed, but it is still necessary to put it into practice in order to obtain a clearer picture and assess the performance of the proposed model (scalability, latency, and congestion).

Another problem that could be further investigated is related to storage. The storage space in such models is an issue, as already indicated. Since drones will be able to securely store enormous amounts of data on the proposed file system, the construction of an off-chain IPFS decentralized network and its integration into this system could offer a solution to this specific problem.

Furthermore, a topic for future research is private key management in these systems. Private keys are used for the encryption and signing of transactions. The potential leak of these keys could lead to unauthorized actions that would be crucial and devastating to the system's performance. The development of a mechanism for securing and managing these keys could enhance this work and contribute to a more secure and robust scheme.

Finally, in these systems, the development of a recovery mechanism is essential. In situations where a drone (or a group of drones) becomes compromised, the deployment of a recovery mechanism is necessary. This ensures that the drone is safely returned to the base after being isolated from the blockchain so that it is not lost, as any loss from the system incurs costs.

In conclusion, drone swarm technology is gaining popularity due to its adaptability and effectiveness in a variety of activities. By strengthening security and enhancing autonomous decision making, the proposed model has the potential to contribute to the rapid development of UAV cluster technology.

Author Contributions: Conceptualization, A.L. and A.K.; Methodology, A.L. and A.K.; Software, A.K.; Validation, A.L. and A.K.; Formal Analysis, A.K.; Investigation, A.K.; Resources, A.K.; Data Curation, A.K.; Writing—Original Draft Preparation, A.K.; Writing—Review & Editing, A.L.; Visualization, A.K.; Supervision, A.L.; Project Administration, A.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Source code available from: https://github.com/CoolHeis/msc01182023, accessed on 1 September 2023.

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

References

1. Zhou, X.; Gao, F.; Fang, X.; Lan, Z. Improved bat algorithm for UAV path planning in three-dimensional space. *IEEE Access* **2021**, *9*, 20100–20116. [CrossRef]
2. Liu, H.; Chen, Q.; Pan, N.; Sun, Y.; Yang, Y. Three-Dimensional Mountain Complex Terrain and Heterogeneous Multi-UAV Cooperative Combat Mission Planning. *IEEE Access* **2020**, *8*, 197407–197419. [CrossRef]
3. Arafat, M.Y.; Moh, S. Localization and clustering based on swarm intelligence in UAV networks for emergency communications. *IEEE Internet Things J.* **2019**, *6*, 8958–8976. [CrossRef]
4. Zhou, Y.; Rao, B.; Wang, W. UAV swarm intelligence: Recent advances and future trends. *IEEE Access* **2020**, *8*, 183856–183878. [CrossRef]
5. Abro, G.E.M.; Zulkifli, S.A.B.M.; Masood, R.J.; Asirvadam, V.S.; Laouti, A. Comprehensive Review of UAV Detection, Security, and Communication Advancements to Prevent Threats. *Drones* **2022**, *6*, 284. [CrossRef]
6. Pandey, G.K.; Gurjar, D.S.; Nguyen, H.H.; Yadav, S. Security threats and mitigation techniques in uav communications: A comprehensive survey. *IEEE Access* **2022**, *10*, 112858–112897. [CrossRef]
7. Leible, S.; Schlager, S.; Schubotz, M.; Gipp, B. A review on blockchain technology and blockchain projects fostering open science. *Front. Blockchain* **2019**, *2*, 28. [CrossRef]
8. Casino, F.; Dasaklis, T.K.; Patsakis, C. A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telemat. Inform.* **2019**, *36*, 55–81. [CrossRef]
9. Taylor, P.J.; Dargahi, T.; Dehghantanha, A.; Parizi, R.M.; Choo, K.K.R. A systematic literature review of blockchain cyber security. *Digit. Commun. Netw.* **2020**, *6*, 147–156. [CrossRef]
10. Konert, A.; Balcerzak, T. Military autonomous drones (UAVs)-from fantasy to reality. Legal and Ethical implications. *Transp. Res. Procedia* **2021**, *59*, 292–299. [CrossRef]
11. Vacca, A.; Onishi, H. Drones: Military weapons, surveillance or mapping tools for environmental monitoring? The need for legal framework is required. *Transp. Res. Procedia* **2017**, *25*, 51–62. [CrossRef]
12. Castelló Ferrer, E. The blockchain: A new framework for robotic swarm systems. In Proceedings of the Future Technologies Conference, Vancouver, BC, Canada, 15–16 November 2018.
13. Kuzmin, A.; Znak, E. Blockchain-base structures for a secure and operate network of semi-autonomous unmanned aerial vehicles. In Proceedings of the IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), Singapore, 31 July–2 August 2018.
14. Hafeez, S.; Khan, A.R.; Al-Quraan, M.; Mohjazi, L.; Zoha, A.; Imran, M.A.; Sun, Y. Blockchain-Assisted UAV Communication Systems: A Comprehensive Survey. *IEEE Open J. Veh. Technol.* **2023**, *4*, 558–580. [CrossRef]
15. Wang, J.; Liu, Y.; Niu, S.; Song, H. Lightweight blockchain assisted secure routing of swarm UAS networking. *Comput. Commun.* **2021**, *165*, 131–140. [CrossRef]
16. Santos de Campos, M.G.; Chanel, C.P.; Chauffaut, C.; Lacan, J. Towards a blockchain-based multi-uav surveillance system. *Front. Robot. AI* **2021**, *8*, 557692. [CrossRef] [PubMed]
17. Chao, H.; Maheshwari, A.; Sudarsanan, V.; Tamaskar, S.; DeLaurentis, D.A. UAV traffic information exchange network. In Proceedings of the Aviation Technology, Integration, and Operations Conference, Atlanta, GA, USA, 25–29 June 2018.
18. García-Magariño, I.; Lacuesta, R.; Rajarajan, M.; Lloret, J. Security in networks of unmanned aerial vehicles for surveillance with an agent-based approach inspired by the principles of blockchain. *Ad Hoc Netw.* **2019**, *86*, 72–82. [CrossRef]
19. Islam, A.; Shin, S.Y. BUAV: A blockchain based secure UAV-assisted data acquisition scheme in Internet of Things. *J. Commun. Netw.* **2019**, *21*, 491–502. [CrossRef]
20. Golosova, J.; Romanovs, A. The Advantages and Disadvantages of the Blockchain Technology. In Proceedings of the IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Vilnius, Lithuania, 8–10 November 2018.
21. Monrat, A.A.; Schelén, O.; Andersson, K. A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access* **2019**, *7*, 117134–117151. [CrossRef]
22. Wang, W.; Hoang, D.T.; Hu, P.; Xiong, Z.; Niyato, D.; Wang, P.; Wen, Y.; Kim, D.I. A survey on consensus mechanisms and mining strategy management in blockchain networks. *IEEE Access* **2019**, *7*, 22328–22370. [CrossRef]
23. Cui, G.; Shi, K.; Qin, Y.; Liu, L.; Qi, B.; Li, B. Application of block chain in multi-level demand response reliable mechanism. In Proceedings of the 3rd International Conference on Information Management (ICIM), Chengdu, China, 21–23 April 2017.
24. Alladi, T.; Chamola, V.; Sahu, N.; Guizani, M. Applications of blockchain in unmanned aerial vehicles: A review. *Veh. Commun.* **2020**, *23*, 100249. [CrossRef]
25. Liang, X.; Zhao, J.; Shetty, S.; Li, D. Towards data assurance and resilience in IoT using blockchain. In Proceedings of the MILCOM 2017–2017 IEEE Military Communications Conference, Baltimore, MD, USA, 23–25 October 2017.
26. Lin, C.; He, D.; Kumar, N.; Choo, K.K.R.; Vinel, A.; Huang, X. Security and privacy for the internet of drones: Challenges and solutions. *IEEE Commun. Mag.* **2018**, *56*, 64–69. [CrossRef]

27. Gipp, B.; Kosti, J.; Breitinger, C. Securing video integrity using decentralized trusted timestamping on the bitcoin blockchain. In Proceedings of the 10th MCIS Conference, Paphos, Cyprus, 4–6 September 2016.
28. White, R.; Caiazza, G.; Cortesi, A.; Im Cho, Y.; Christensen, H.I. Black block recorder: Immutable black box logging for robots via blockchain. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3812–3819. [CrossRef]
29. Liao, S.; Wu, J.; Li, J.; Bashir, A.K.; Yang, W. Securing collaborative environment monitoring in smart cities using blockchain enabled software-defined internet of drones. *IEEE Internet Things Mag.* **2021**, *4*, 12–18. [CrossRef]
30. Dasu, T.; Kanza, Y.; Srivastava, D. Geofences in the sky: Herding drones with blockchains and 5G. In Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 6–9 November 2018.
31. Chang, T.; Yu, H. Improving electric powered UAVs' endurance by incorporating battery dumping concept. *Procedia Eng.* **2015**, *99*, 168–179. [CrossRef]
32. Eldeeb, E.; de Souza Sant'Ana, J.M.; Pérez, D.E.; Shehab, M.; Mahmood, N.H.; Alves, H. Multi-UAV path learning for age and power optimization in IoT with UAV battery recharge. *IEEE Trans. Veh. Technol.* **2022**, *72*, 5356–5360. [CrossRef]
33. Santin, R.; Assis, L.; Vivas, A.; Pimenta, L.C. Matheuristics for multi-uav routing and recharge station location for complete area coverage. *Sensors* **2021**, *21*, 1705. [CrossRef] [PubMed]
34. Shiau, J.K.; Ma, D.M.; Yang, P.Y.; Wang, G.F.; Gong, J.H. Design of a solar power management system for an experimental UAV. *IEEE Trans. Aerosp. Electron. Syst.* **2009**, *45*, 1350–1360. [CrossRef]
35. Lachow, I. The upside and downside of swarming drones. *Bull. At. Sci.* **2017**, *73*, 96–101. [CrossRef]
36. Grimal, F.; Sundaram, J. Combat drones: Hives, swarms, and autonomous action? *J. Confl. Secur. Law* **2018**, *23*, 105–135. [CrossRef]