Cluster Computing
https://doi.org/10.1007/s10586-021-03497-9

Derived blockchain architecture for security-conscious data dissemination in edge-envisioned Internet of Drones ecosystem

Maninderpal Singh[1] · Gagangeet Singh Aujla[2] · Rasmeet Singh Bali[1]

Received: 10 June 2021 / Revised: 28 October 2021 / Accepted: 24 November 2021

Abstract
Internet of Drones (IoD) facilitates the autonomous operations of drones into every application (warfare, surveillance, photography, etc) across the world. The transmission of data (to and fro) related to these applications occur between the drones and the other infrastructure over wireless channels that must abide to the stringent latency restrictions. However, relaying this data to the core cloud infrastructure may lead to a higher round trip delay. Thus, we utilize the cloud close to the ground, i.e., edge computing to realize an edge-envisioned IoD ecosystem. However, as this data is relayed over an open communication channel, it is often prone to different types of attacks due to it wider attack surface. Thus, we need to find a robust solution that can maintain the confidentiality, integrity, and authenticity of the data while providing desired services. Blockchain technology is capable to handle these challenges owing to the distributed ledger that store the data immutably. However, the conventional block architecture pose several challenges because of limited computational capabilities of drones. As the size of blockchain increases, the data flow also increases and so does the associated challenges. Hence, to overcome these challenges, in this work, we have proposed a derived blockchain architecture that decouples the data part (or block ledger) from the block header and shifts it to off-chain storage. In our approach, the registration of a new drone is performed to enable legitimate access control thus ensuring identity management and traceability. Further, the interactions happen in the form of transactions of the blockchain. We propose a lightweight consensus mechanism based on the stochastic selection followed by a transaction signing process to ensure that each drone is in control of its block. The proposed scheme also handles the expanding storage requirements with the help of data compression using a shrinking block mechanism. Lastly, the problem of additional delay anticipated due to drone mobility is handled using a multi-level caching mechanism. The proposed work has been validated in a simulated Gazebo environment and the results are promising in terms of different metrics. We have also provided numerical validations in context of complexity, communication overheads and computation costs.

Keywords Blockchain · Edge computing · Derived architecture · Internet of Drones · Security

1 Introduction

The globe is evolving to improve the quality of human life by easing their day-to-day activities through the intervention of advanced technologies. Starting with the invention of fire, then shaping stones into wheels, to all the way flying to space, the human race has shown significant advancements in all horizons. Transportation is one such area that has seen magnificent advancements and technological accomplishments. Initially, the focus was to come up with mechanical solutions, that went on to provide comfort and ease of driving, until recently the focus shifted towards autonomous operations of vehicles [1]. One such evolution in the transportation sector is the

This work is an extension of M. Singh, G. S. Aujla and R. S. Bali, "ODOB: One Drone One Block-based Lightweight Blockchain Architecture for Internet of Drones," IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2020, pp. 249–254.

✉ Gagangeet Singh Aujla
gagi_aujla82@yahoo.com

Maninderpal Singh
mpvirdi@gmail.com

Rasmeet Singh Bali
rasmeetsbali@gmail.com

[1] Computer Science and Engineering Department, Chandigarh University, Mohali, India

[2] Department of Computer Science, Durham University, Durham, UK

Published online: 11 January 2022

evolution of drones that can fly without the need for onboard human presence. Initially, the drones were developed for military applications like carrying out aerial surveys of the battlefield, providing essential supplies to soldiers in batter fields, and carrying out missions that are normally too risky for a plane with a pilot. But, in the past decade, the application of drones is advancing towards non-military and commercial domains at a consistent pace [2]. The factors that are favoring this advancement, include the progression of drone hardware (i.e, they are becoming small yet more powerful) and more importantly the software environments for drones are evolving towards the autonomy of their operations [3].

These advancements resulted in a powerful ecosystem known as the Internet of Drones (IoD), which supports the autonomous operations and mobility of drones. The IoD divides the airspace into logical partitions known as zones [4]. Drones move from one zone into another zone through zone intersection points [5]. The places of interest are designated as nodes in the IoD ecosystem. The zonal and inter-zonal movements of drones occur in a fixed path fashion which is predefined whereas once the drone reaches nodes they are free to move. These movements are controlled by the central controlling authority known as the aviation authority. Different zones have their designated service-providing and controlling entities that are known as zone service providers (ZSP). The ZSPs provide navigational information to drones and the aviation authority manages the mutual working of ZSPs. Using the above conceptual architecture, the IoD facilitates the autonomous operations of drones into every application (warfare, surveillance, photography, rescue, delivery, etc) across the world. The transmission of data (to and fro) related to these applications occur between the drones and the other infrastructure over wireless channels for processing or decision making. The applications desire a stringent latency restriction to improve the overall performance. Generally, this data is relayed to the central cloud facility providing computing power and storage capabilities. However, relaying this data to the core cloud infrastructure (for processing and storage) may lead to a higher round trip delay.

In [6–8] explored the fog-cloud interplay to resolve such challenges in the related domains. Similarly, in [9], a multi-agent-based fog computing model was proposed for task management jobs similar to the agent-based architecture proposed for UAV navigation in [10]. Thus, it becomes viable to utilize the cloud close to the ground, i.e., edge computing. This helps the drones to connect with the edge and process (or analyze) the data and meet the stringent latency requirements. Thus, this advent a novel edge-envisioned IoD architecture that combines the capabilities of edge with IoD for the overall welfare of the underlying applications.

However, the data in edge-envisioned IoD is relayed over an open channel, so it is often prone to different attacks and malicious activities. Moreover, as drones are small and can fly in partial to full stealth modes for humans they can be a big risk in terms of privacy infringement because of video surveillance capabilities and possess a big potential risk of transporting vulnerable materials like explosives to places of interest. The security of the IoD is very volatile and highly susceptible to security breaches. As the attack surface is really large because of various levels of communications like drone to drone (D2D), drone to infrastructure (D2I), and transmitting data to an edge or central cloud-based repository. Thus, a robust solution that can sustain the confidentiality, integrity, and authenticity of the data while providing desired services to the end-users is the utmost need of time. Being an amalgamation of heterogeneous networks, in an edge-envisioned IoD environment, the security implications are not just limited to the inherent security challenges brought by sensor networks, mobile communication and cellular networks, and the open channel Internet, but it also includes the privacy preservation and protection issues and challenges. Therefore, drones have to support the advanced security schemas and concepts such as authentication, authorization, data integrity and protection, confidentiality, authorization and access control, and cyber-attack protection and prevention under one umbrella.

Several existing proposals proposed different solutions to secure the IoD ecosystem by protecting the drones by using multilevel and multi-domain strategies. However, this often involves the use of heavy cryptographic computations which should be avoided in the drone-based environment to save resources as well as provide speedy operations. The challenge of enhancing the flight time within the battery power limits the usage of hard security primitives in the IoD. The existing security mechanisms aren't sufficient for the IoD ecosystems as they have inherited limitations of computational overhead leading to more power-intensive operations [11]. The IoD is a resource-constrained ecosystem (in terms of computing and battery power) and thus conventional security primitives pose stringent challenges in terms of the additional computational burden. Hence, the alternative mechanisms must be explored for their possible adaptability in IoD. Furthermore, the conventional security primitives are not suitable to scale up to meet the autonomous operations in the IoD ecosystem. Hence several researchers have suggested using alternative mechanisms to ensure security and privacy in the IoD.

IoD relies on resource-sensitive systems and machines and thus strong and heavy cryptographic mechanisms

impose additional overhead. Hence, the radical and unorthodox mechanisms must be explored and validated for their possible adaptability and suitability in the edge-envisioned IoD system. Blockchain is one emerging technology which although became famous for use in cryptocurrencies but its application in other areas (like the Internet of Things) is being explored extensively [12–14]. It ensures the security of data without the governance by central entities and provides de-centralized architecture comprising of a distributed, shared and immutable ledger [15, 16]. The success of blockchain is credited to its distributed ledger structure and immutable data recording capabilities. Moreover, the resilience and traceability is the most favoring factor for blockchain apart from its immutability and privacy preservation using the hashed identities to identify entities. So, in the IoD landscape, blockchain is a viable option to ensure security in a scalable manner [17, 18].

1.1 Research problem?

The legacy blockchain architecture requires heavy computations and involves huge network interactions that are again not suitable for the resource-limited drone environment if adopted in its standard form. The conventional blockchain architectures rely on mining processes for ensuring the trust of involved entities. Moreover, the various consensus algorithms like proof of work are very resource exhaustive as they require solving complex mathematical problems [19, 20]. Even huge mining farms with dedicated graphical processing units (GPU) take time to solve the problem used in conventional mining techniques. Hence, using blockchain in IoD like ecosystem brings in a big problem. Moreover, the blockchain architecture requires nodes to synchronize the blocks to maintain a consistent state, which requires a large number of network communications thus pressurizing the underlying network and further looking at the high mobility of IoD systems the network typologies tend to change frequently worsens the problem. Several researchers have proposed solutions like in [21, 22], to use a derived blockchain that copes with the above-discussed challenges must be developed for drone-like application areas. Some works like in [23] used an adapted blockchain architecture to meet the needs of intelligent transportation systems whereas the authors in [24] used an adapted blockchain architecture for meeting the needs of IoD.

The conventional blockchain stores the transactions in blocks linked together via hashes. New transactions go into new blocks and older blocks become immutable. Over time as newer blocks get added into the blockchain, the size tends to grow [25]. Moreover, as transactions about an entity are spread across multiple blocks the process of traversing transactions requires locating multiple blocks and then accessing them. In [23, 26] the authors suggested that the decoupling of data from the blockchain is a viable solution in the resource-constrained ecosystems. Hence, we need to split the data into header and trailer parts and keep only the headers in the blockchain. Moreover, it also becomes important to design an appropriate consensus mechanism for IoD and incorporate the performance improvement techniques like caching into the ecosystem. Further, as suggested by [27, 28] the special architectures of blockchain need to be analyzed for their real-time performance using which they can be further fine-tuned to offer better performance and throughput. Hence, a derived blockchain architecture based on off-chain data is more suitable than conventional blockchain in resource-constrained environments.

1.1.1 Research questions?

To realize the full potential of an unconventional blockchain-based security solution for an edge-envisioned IoD ecosystem several concerns must be effectively addressed. To this end, the security solution ecosystems should be both computer-friendly and auditable. However, to achieve this, we need to answer the following research questions (RQ):

- *RQ1* How to design an unconventional block structure (like one drone one block) through the decoupling of data from the blockchain (shifting it to off-chain storage).
- *RQ2* How to develop an access control and identity verification mechanism that considers the privacy of the participating entities.
- *RQ3* How to design a lightweight algorithm that allows only stochastically valid participants to ensure distributed consensus and timeliness in the IoD ecosystem?
- *RQ4* How to control the expanding storage requirements of conventional blockchain so that it can be suitable for the IoD?
- *RQ5* How to ensure that mobility associated with the drones do not hinder the near to real-time communication and decision making in the edge-envisioned IoD ecosystem.

1.2 Overview of the research approach and contributions

Looking into the research questions, we have extended our previous work [29] to propose a derived blockchain architecture for providing secure data dissemination in the edge-envisioned IoD ecosystem. The block is the vital

component of blockchain and with time they tend to pile up, moreover, the traversal of transactions for one entity may span across multiple blocks, which is complex and requires all participants to store all blocks. Hence, a special block structure that generates one block per entity and off-chain storage of data is proposed (Sect. 4). In our approach, the registration of a new drone is performed to enable legitimate access control thus ensuring identity management and traceability (Sect. 5.1). Further, the interactions in the ecosystem are performed in the form of transactions of the blockchain (Sect. 5.2). To overcome the existing challenges, our approach realizes a lightweight consensus based on the stochastic selective consensus algorithm (Sect. 5.3). This is followed by a transaction signing process to ensure that each drone is in control of its block (Sect. 5.4). Another big problem considered relates to the expanding storage requirements, that are handled with the help of data compression using a shrinking block mechanism (Sect. 5.5). Lastly, the problem of additional delay anticipated due to drone mobility while accessing services through edge infrastructure is handled using a multi-level caching mechanism. The overview of the research approach is depicted with the help of a schematic diagram in Fig. 1.

The salient contributions of this work are listed below.

- We have deployed an edge-envisioned IoD ecosystem to facilitate secure operations for drone-related applications using derived blockchain.
- A derived block structure that decouples the data part (or block ledger) from the block header is proposed to overcome the limitation of conventional block structure in the context of the IoD ecosystem.
- A block shrinking mechanism is proposed to ensure the data stored on the drone is retained within a stipulated size. This supplements the derived blockchain architecture to optimize the storage.
- We have proposed a stochastic consensus technique to ensure that only legitimate entities participate in the consensus thus ensuring the integrity of the system.
- We have proposed a caching mechanism in the edge-envisioned IoD that supports multi-level caching related to edge-to-edge, drone-to-drone, and edge-to drone context.
- The proposed work has been validated in a simulated environment and the initial results are promising in terms of different metrics. Also, we have provided numerical validation of the proposed architecture.

2 Related work

Various existing proposals have explored different dimensions related to the IoD in an attempt to devise a flawless system with viable work-ability. The relevant existing proposals are discussed based on different perspectives like drone navigation and placement, security in drones, legacy blockchain for drone security, adapted blockchain for resource-constrained environments.

2.1 Drone navigation/placement

The IoD environment provides a layered architecture for coordinating and enabling drone flights in a shared airspace [4]. Hence the navigation and placement of drones become a very crucial planning and operational aspect. Many researchers have explored this field from different aspects as in [30], the authors have studied the optimizations of drones along with corresponding energy consumption. Researchers in [31] have discussed the communication limitations of drones due to limited radio transmission power and constrained onboard computation and storage resources. The authors have proposed a cloud empowered mechanism to enable drone communications over the internet. This brings an advantage of management capabilities from any place, no matter how far from the physical location of the drone. A Work in this direction as presented in [32] has brought forward the attention towards special service scheduling needs of the drones in IoD. The authors in their work have proposed a priority-based service scheduling technique for IoD keeping ahead of the special communication needs of drones in uploading and downloading the information. Moreover, recent research trends towards usage of artificial intelligence and machine learning for navigating drones on their own are progressive like in [33]. In this work, the authors have studied the deep convolutional neural network for enabling the steering of drones as per real-life changing scenarios. Based on these works, the related key gaps in the domain of drone navigation and placement include the requirement of a mechanism for transferring the navigational information among

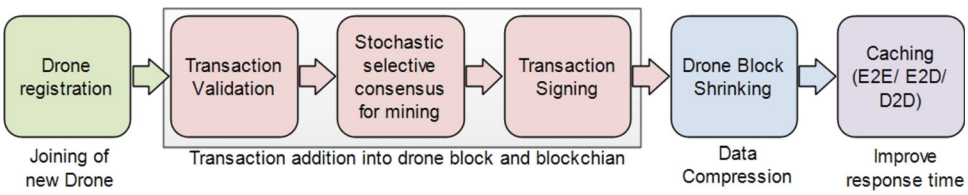

Fig. 1 Overview of the research approach

the various entities of the IoD ensuring data security and access to legitimate entities.

2.2 Security mechanisms for drone communication

The IoD environment functions in open air space that is shared by multiple others in terms of communications and drone movements. Because of this, the drone has various security challenges. Authors in [34] studied various aspects and horizons of security on IoD. A detailed analysis of each possible attack on IoD is analyzed very carefully concerning the impact domain security parameters. The authors have enlisted the taxonomy for the domain-wise attacks in IoD. Further, in [35] the researchers have emphasized various domains of security like the secure channel between the drones and access points. In existing works authentication, authorization, and confidentiality of data in the IoD environment is achieved in most approaches using public-private key pairs which are resource exhaustive. As a solution to which recent advances for lightweight authentications and access control techniques have been proposed in [11]. In a similar work, [36] the authors proposed a network encoding technique that saves the computational needs leading to improved power needs. Authors highlighted the idea that the legacy privacy-preserving techniques are not relevant to IoD as they are heavy for the ecosystem to handle on drones, hence a network coding-based pseudonym scheme is proposed. Authors in [37] have highlighted the need for countermeasures as drones lack chip-level security hence leading to attacks like the man in the middle. The above discussion justifies the need for security solutions suitable to the IoD requirements keeping in mind that the scalability to match the magnitude of the IoD landscape.

2.3 Legacy blockchain for drone security

As the IoD has a large attack surface area that is luring for attackers to exploit, researchers are exploring the possibility of non-conventional security mechanisms like blockchain to ensure a safe working ecosystem. Authors in [22] explored the use of blockchain for authentication and data sharing in a 5G-based drone network. In another work in [38] the authors have proposed a blockchain-based IoT platform for the management of autonomous operations of drones. A detailed system model for the deployment of service is presented along with various operations of the drones in the proposed ecosystem. Another work in [39] presents the blockchain to secure the drones. The author has justified the usage of blockchain in ensuring trust in the IoD along with performing the adoption tests of blockchain. In another work in [40] the authors propose the blockchain applicability in various applications of IoT. Recently IEEE has come up with a standard for blockchain-based IoT deployment [41]. This standard defines the data management functionality for IoT applications. Based on these works, blockchain appears to be a suitable candidate for the IoD, however, it has its limitations like low transaction rate, elevated storage requirements, and energy exhaustive consensus algorithm which needs to be addressed before it can be deployed at full scale in IoD.

2.4 Adapted blockchain for resource-constrained environments

Conventional blockchains are not designed for an IoD-like ecosystem and hence researchers have explored the suitable blockchain architecture. In this direction a few researchers like [23] in their work proposed an alternative version of blockchain that is adapted as per the needs of such environments. The authors proposed a framework known as speedy chain which decouples the data from the blockchain, hence eliminating the need to process large data per blockchain and saving space in terms of the storage of transactions as well. Another similar work in [25] proposes the decoupling of the data from the blockchain. The authors propose to store the data in OrbitDB using the Inter-Planetary File System (IPFS). The proposed work suggests computing the hash of the stored data in OrbitDb and then storing the hash inside the blockchain providing the immutability to the transactions and also keeping the burden of storage requirement of the whole blockchain away from the drones. In another blockchain framework that uses decoupling of data at its core in [21], the authors have proposed it for edge-based big data management. The above discussed lightweight architectures try to overcome the limitations of conventional blockchain but they are not specifically designed for IoD but in the related ecosystems. Hence, a customized blockchain solution for IoD security is still among the paths untraveled.

2.5 Edge caching to provide mobility tolerance

Work by researchers in [42] suggests shifting the burden of heavy computational work such as the mining process onto the edge devices, leading to resource-constrained devices not being burdened due to the functional dependencies of the blockchain. Authors have suggested creating a pool of mining resources that may comprise edge devices and also the mobile devices within the mining cluster whenever required. In another work, [43] the authors have presented an edge-based data processing model in a V2X environment. Authors have considered multiple objective solutions to deal with delay, energy consumption, service level

agreements while migrating tasks to edge devices. Leap forwarding in the same direction researchers in [44] proposed the use to edge-based caching in the UAV environment. Authors have brought attention towards using newer metrics to evaluate the reliability of the network regarded as ultra-reliability. The drones in the proposed work are used as on-demand caching nodes to improve mobile caching and hence improve the ultra-reliability. The aforementioned proposals suggest that the real-time communication in IoD is challenged by the mobility due to drone flights. Hence, a suitable caching mechanism that works at multiple levels should be explored to achieve robust performance.

Table 1 summarizes the various existing works related to the IoD-like environments and blockchain. A majority of them are using non-conventional blockchain to deal with the needs of the IoD. For example, in [21, 23, 25], a decoupled data blockchain is used to storage of data off the chain.

3 System model

The IoD ecosystem consists of various participating entities and underneath technologies that are tuned and coordinated to work in tandem with each other. IoD may have different geographical deployments, heterogeneous supporting network infrastructure, and control methods. In this section, the system architecture for the proposed derived blockchain for security-conscious data dissemination in the edge-envisioned IoD ecosystem is presented. The system model for the proposed approach depicting different communications types, controlling entities, and deployment environments is shown in Fig. 2.

The communication model for the IoD ecosystem is depicted in Fig. 3. It shows the ground stations as the point of contact for the drones to interact with the edge-based aviation authority. The drones are presented with nomenclature like D_i where i depicts the drone number. The communication between the ground stations and drones can be direct or through an ad-hoc network. On ground stations, the blocks of the different drones are identified using the D_i that serves as the unique identity of the drone.

3.1 Aviation Authority

The predominant entity in the proposed architecture is the aviation authority (A_v). Permissioned blockchain's access control is managed by A_v. When a drone wants to join the blockchain network for the first time it must get the registration done on A_v. The proposed techniques make use of hardware-based security to avoid identity theft. Each drone is enabled with a trusted platform module (*TPM*). *TPM* makes use of the endorsement key(E_k) which is pair of

Table 1 Comparative analysis of existing proposals

Proposal	1	2	3	4	5	6	7	8	9
[24]	Coupled	Regular updating	IoD	Pseudo-random IDs	Yes	No	No	Yes	No
[23]	Decoupled	Regular updating	ITS	Public key with periodic change	No	Yes	No	No	No
[26]	Coupled	Grouped transactions	ITS	–	No	No	No	No	No
[45]	Coupled	Grouped transactions	IoT	–	No	Yes	No	No	No
[21]	Decoupled	Express transactions	Edge	Wallet	Transaction offloading	No	No	No	No
[44]	Coupled	Regular updating	IoD	Public key cryptography	No	No	Yes	No	No
[46]	Coupled	Regular Updating	IoD	Session keys	No	No	No	Yes	No
[22]	Coupled	Grouped transactions	IoD	Smart contracts	No	No	No	Attribute-based encryption	No
[47]	Coupled	Regular updating in public blockchain	IoD	Public/Private Key pair	No	No	No	No	No
[48]	Coupled	Regular updates in private blockchain	IoD	Drone unique id	No	No	No	Encrypted payload	No
[25]	Decoupled	Regular transaction using orbitDB	IoD	Yes	No	No	No	Encrypted payload	No
Proposed Approach	Decoupled	Appendable	IoD	Public key	Shrinking	Yes	Yes	Yes	TPM

1: Data coupling, 2: Nature of block creation, 3: Application, 4:Identity management, 5:Data compression, 6:Amendable blocks, 7: Edge Caching, 8: Secure data delivery, 9: Hardware level key security

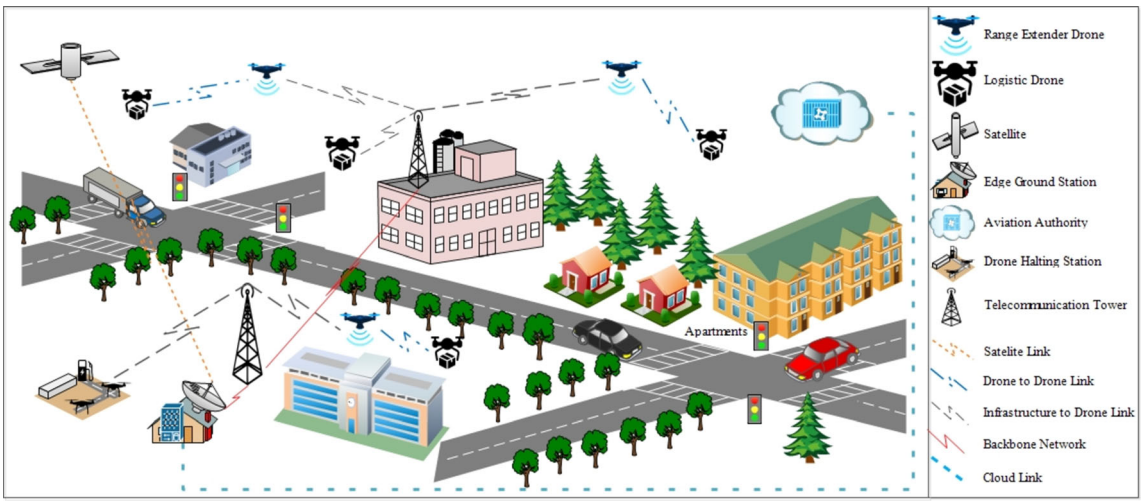

Fig. 2 System model

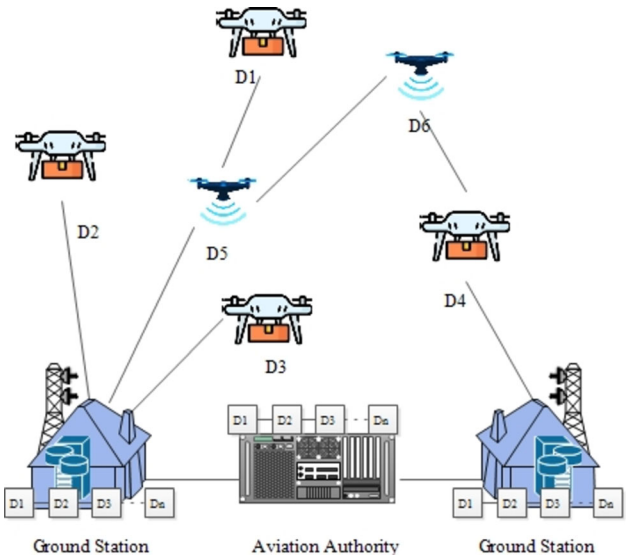

Fig. 3 Communication model

public and private keys. In TPM the E_k is stored within the chip and it is accessed using a drone onboard software. A storage root key (*SRK*) is generated when a new drone is registered with A_v. The *SRK* requires the drone owner to enter a password for its initialization. *SRK* along with E_k are used to generate public key (K_{pb}) and private key (K_{pr}) by the *TPM*.

3.2 Ground stations

The flying areas in the ecosystem are divided into zones. Each zone is served by ground stations (G_s). The G_s manages and controls all actions performed by various entities within its zone. In the proposed framework the data is decoupled from the blockchain. Unlike conventional blockchain, the data is not included in the hashed block linking, only the header is used to interlink various blocks to form a blockchain. As the block header contains the drone identity (D_i) which is a 64-bit unique identifier for a drone that is generated using the combination of the hash of the drone's serial number (D^{sr}_{hash}) and the on-board hash of on-board firmware (D^{fw}_{hash}). In the event of a drone being captured by an attacker and the firmware, tampering is attempted the drone gets auto unauthenticated from the blockchain. The drones are required to store only the blockchain of headers and only the individual data. On the other side the G_s store the blockchain of headers as well as block data for all drones. When the drone is registered successfully with the A_v, its K_{pb} is used by the G_s for identification of the corresponding data ledger of the D_i reading the D_i from the header derived blockchain. The transaction for D_i are authenticated using K_{pb} of D_i.

3.3 Drones

The drones are the core entity of the system model. In the proposed model, the drones are segregated into different categories depending upon their roles and responsibilities along with their physical attributes as enlisted below:

- *Logistic Drones* are used for transportation services. They deliver services and goods to the destination. The high load carry capacity is only feasible because of their size being in a miniature to a small category of drones. The maximum take of weight (*MTW*) of these drones is in up to 25 kilograms (*Kg*) for small and up to 10 *Kg* for miniature [49].
- *Range Extender Drones* enable the infrastructure available to other drones that don't have direct access to the infrastructural network. The hardware in terms of battery and load-carrying capacity on these drones is

relatively small to logistic drones yet their computational and communication resources are sufficient to enable them as service relays. The natural disaster hit areas or areas where the demand and supply of network resources vary with time can be serviced on-demand using the range extender drones. The size of the range extender drones is in the nano category weighing less than 500 grams (g), as they only carry their weight and no additional payload like the logistic drones. The charging of this special category of drones can be hybrid, i.e., they are charged either through the ground stations or on the go through wireless charging (Table 2).

All these drones are enabled with the *TPM* capabilities to ensure the security of the infrastructure even if a drone is captured by the attackers.

3.4 Edge layer

Edge computing is a felicitous tool to balance the performance requirements of the IoD system and blockchain. If the tasks are offloaded onto the cloud to get the work done, it introduces the communication overhead, hence edges computing surfaces as a surrogate. In the context of this work, edge computing is a concept where the cost of communication is reduced by either performing the computations on the drone itself or near to drone. The computation tasks can be further classified as

- *Lightweight tasks* Actions which can be performed on a drone without exhausting much of its resources are lightweight tasks. Such tasks are mapping of route plan and real-time positioning.
- *Heavyweight tasks* The operations which are performed on a drone that will exhaust its resources like battery power faster are regarded as heavyweight operations. A task like mining of the blockchain, full blockchain maintenance is categorized as heavyweight tasks.

The ground station (G_s) is the communication point between the drone and A_v. Also, G_s has relatively more computational and storage resources in comparison to drones. The drones are lightweight and the prime focus is the maximization application tasks. Hence, G_s is a suitable edge layer between the A_V and the drones. D_i can be highly mobile in the *IoD* environment. Hence, the ground station serving the drone ($G_{s\to d_i}$) tends to perform the handoff operations. A suitable $G_{s\to d_i}$ must satisfy the desired signal strength ($\tau_{D_i\to G_s}$) for communication between D_i and $G_{s\to d_i}$ and it must pose the computational resources to perform the blockchain operations like mining which is probabilistic and is dependent on the complexity of the nonce and the difficulty of the consensus algorithm. Resource requirement by D_i from G_s is shown as follows:

$$R_{D_i}^{rq} = R_{D_i}^{P} + R_{D_i}^{M} + R_{D_i}^{B} + R_{D_i}^{pow} \quad (1)$$

Where, $R_{D_i}^{P}, R_{D_i}^{M}, R_{D_i}^{B}, R_{D_i}^{pow}$ denotes the computational power, memory, bandwidth and the energy required to carry the task of drones on the edge G_S.

4 Block structure in ODOB

In the proposed scheme, each drone has its block (B_{D_i}), where it stores the data including the information related to the control and service relay. The control information comprehending the details related to drone flights, GPS coordinates, altitude information (to ensure the correct positioning of the drone in the three-dimensional space), source and destination position coordinates, complete movement plan, payload, amount of estimated power required by the drone to reach the destination, the intermediate halting station details which the drone can take in case it has to deliver services at multiple hops in a single flight. The block data part and the corresponding header are not stored onto the blockchain synced across all participating entities. The structure of blocks on drones and ground stations is as follows:

4.1 Drone block

In the proposed scheme the block maintained on the drone is specific to that drone to avoid overwhelming the drone

Table 2 Drone comparison

	Logistic drones	Range extender drones
Load type	Self + Payload	Self
Battery capacity	> 20,000 mah	< 20,000 mah
Size	Miniature—small	Nano
Battery recharging	Ground station	Hybrid
On-board processing	Self sufficient	Data relaying and managing
Memory	Small	Comparatively larger
Communication link	Point to point	Point to multipoint
MTW	< 25 kg	< 500 g

resources by storing the complete blockchain. The data stored in the drone's block is the control information for its operational functions and service relay data. The service relay data applies to the range extender drones category.

- *Control Data* The control data within the drone's block include B_n which is the unique serial number given to each block into the blockchain. Drones use public-key cryptography to ensure the integrity and authenticity along with confidentiality of the data stored in blocks. As a resultant the D_i stores its private key (K_{pr}) and public key (K_{pb}) those are generated through the *TPM* module. The mission-specific information like the geographical location coordinates of the destination (D_{gps}) and source location coordinates (S_{gps}) is also the ingredient of the drone block. The chronology of block generation is ensured through the time stamp (t_s) field inside the block header. For the current flight plan, the weight of the payload (W_p), type of the delivery (D_n) identify the nature of the flight. The delivery types are as below:
  - *Normal Delivery* The drones are allowed to halt at intermediate locations for delivering other consignments as well along the route. The value of D_n set to 0 indicates that it is the normal delivery mode.
  - *Urgent Delivery* In this category of deliveries the drones are not allowed to take halts from source to destination. Hence those drones are chosen that have sufficient flight range so that they need not halt at all. This ensures speedy delivery and also the security of the shipment being shipped. The urgent mode is reflected by D_n equal to 1.

  Flight plan (F_p) is the detailed route that the drone takes from sources to destination, along with the urgency flag (F_u) indicating the urgency of the process for process scheduling. Lastly, the older blocks that have shrunken after their scope comes to end their hash is computed using SHA256 and the hash is stored in the following block to ensure the integrity of the system i.e. when the i^{th} block (B_{D_i}) is shrunk and a new block $i+1^{st}$ is generated, the hash value of $H_{B_{D_i}}$ in inserted in the header of $i+1^{st}$.
- *Service Relay Data* This data applies to the service relay drones. As these drones provide on-demand network resources, form the bridge between the infrastructure and the drone network creating a FANET (Flying Ad-Hoc Network) [50]. The information held in includes the following:
  - *Topology* Each service relaying drones holds the network topology of the drones connected to it and the infrastructural node to which it is connected.
  - *Radio Communication* The UAVs make use of line of sight based on radio signal propagation for data dissemination.
  - *Cached data* The service relay drones provide better performance in terms of request-response resolution by providing caching of data to be relayed to drones to avoid re-transmissions from old ground stations (Fig. 4).

4.2 Ground station block

Ground stations (G_s) are the entities those act as edge computing layers in the communication framework of the proposed model. To keep drones free from the burden of storing complete blockchain, drones only store their current blocks. Whereas, the complete blockchain is maintained by the G_s. Since all ground stations keep the blockchain synchronized among themselves through the cloud-based cloud-based central entity A_v, data is provided to drones on demand. The block structure of the blockchain that is maintained at the G_s hence is different from that of the blocks maintained by drones. The ground station in each block maintains the header, as well as data for each drone and this, applies to all drones registered with the A_v. The detailed components of the header and the data part of the block are as below:

- *Header* The header of the block at G_s has the drone identifier field (D_i) which is a unique identifier registered with the A_v. Next, is the hash value computed from the header of the previous block (H_{ph}), the public key of the drone (K_{pb}) computed through the TPM module. The nature/role of the drone is indicated via (D_{role}). The D_{role} classifies the D_i as either range extender or logistic. The timeliness of the block is tracked through the time stamp of the block (t_s). The header also holds the parameters related to drones' capabilities like rated flight capacity (R_{cap}) in terms of

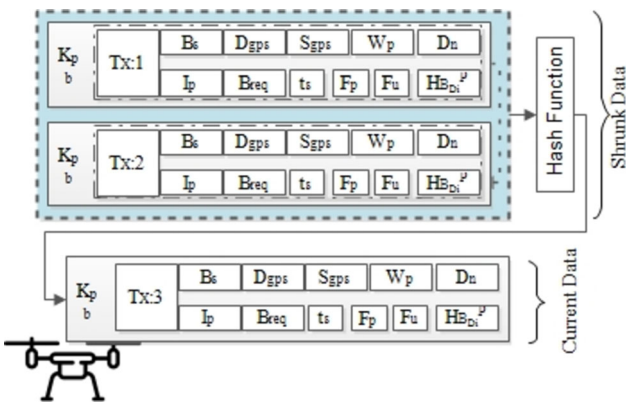

Fig. 4 Block structure used at the drones

distance coverage at given rated speed (R_s) subjected to rated payload capacity (R_w).
- *Data* The data part of the block at G_s comprises the data related to drones in incremental transaction execution order. Although the fields in the data part are identical to those of the block maintained on drones, but on G_s the shrinking of drones is not performed to keep a full copy of the data associated with a drone. This results in an immutable ledger of actions performed by the D_is.

The representation of the block structure at the ground station is in Fig. 5.

5 Blockchain-based security framework for data dissemination

The data dissemination is the structured method of delivering data to the end entities [51], i.e., drones. In the case of autonomous operations, the edge nodes are introduced to improve the performance of the system as suggested in [52]. To enable drone flights large volume of traffic data will be disseminated to the drones from the A_v. With a large number of drones the load on the G_s serving a group of drones in its service area increases. To maintain the performance of the system an edge enabled caching-based data dissemination is proposed in this work as suggested by [53]. The data dissemination model is categorized into three levels. The top-level closest to the end-user application level is the application layer. The second level beneath it is the network layer comprising network components. The lowest level is the Drones layer.

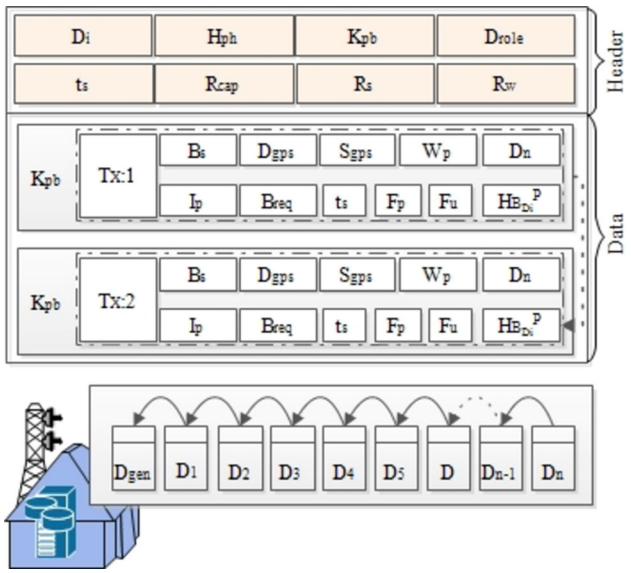

Fig. 5 Block structure at the ground station

- *Application Level* In this level the end-users are present. End-users use various services for which drones are the enablers. Blockchain-based user authorization and authentication are used for access control of users in this layer. The transactions which lead to the movement of drones are initiated via this level.
- *Network Level* The underlying network infrastructure is clubbed into this level. In the proposed system, the ground stations, aviation authorities, and communication components like radio antennas form this level.
- *Drone Level* The main service enablers in the proposed work are drones. Drone take flights based on the fixed flight plans those are offloaded to them by the network level. Although the data reaches the drones via the edge nodes, to improve the system performance and reachability the drone further forms the drone to drone ($D2D$) communication links.

The convention coupled data blockchain suffers from the issues of long delay (during the block update process), low transaction rate, and high storage requirement. This article proposes a decoupled data blockchain framework to overcome these limitations of coupled data blockchain. This framework is built in a modular manner comprising of phases to enable the adaptability of the model to requirements.

5.1 Drone registration

The integrity of the whole system is ensured by only allowing legitimate and verified drones into the ecosystem. Every drone has to register itself on the permissioned blockchain. The access controlling right i.e. which entities are allowed to perform transactions on the blockchain are with the A_v. When a new drone attempts to register itself in the ecosystem, a request (R_{reg}) is sent to the A_v, which includes the K_{pb} and the D_i. The D_i is generated using the Eqs. 2 and 3. Where D_M is the physical address of the network interface card of the drone and R_{and} is the pseudo-random number used to hide D_M.

$$D_i \equiv H_{sh} \leftarrow sh(D_M \oplus R_{and}) \tag{2}$$

$$sh = D_M \% n \tag{3}$$

When A_v receives R_{reg} it checks for the same in the blockchain of headers. If the drone is new and no existing drone block is present for it, a new one is created by the A_v for the seeking drone. As the new block is generated, its header is appended after validation from stochastic selective voting consensus into the blockchain. The registration process among A_v and a drone are shown in Fig. 6 in dialogue representation. The complexity of the proposed registration mechanism is liner, $O(P_{pool})$.

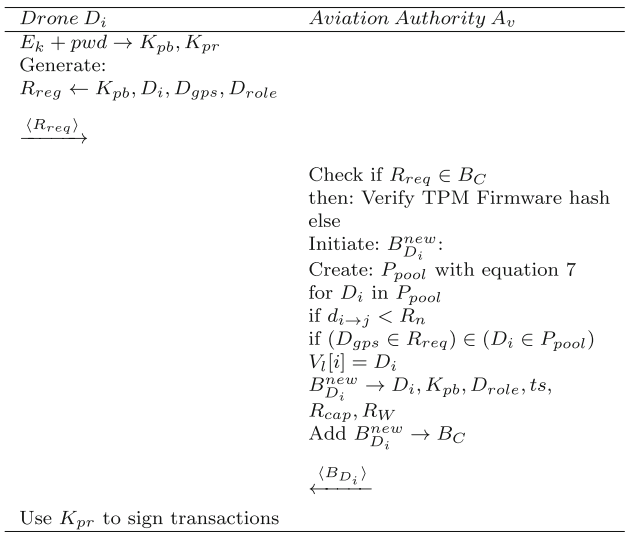

Fig. 6 Registration dialogue between drone and A_v

5.2 Transaction validation

A registered drone stores the transactions as presented in Eq. 4 in the respective block for that drone only. Before the transactions are added into the decoupled data part of the block, they are tested for possible forgery or eavesdropping attacks. The testing is achieved through the usage of the drone's K_{pb} and K_{pr}.

$$D_i(T_x) \supset D_i(T_{x_1} + T_{x_2} + T_{x_3} + ... + T_{x_n}) \tag{4}$$

$$B(D_i) \subseteq B(D_n), n \in (D_i) \tag{5}$$

$$K_{pr}(T_{x_n}) \in D_i(T_x) \tag{6}$$

As when a registered drone issues a transaction, then it is signed using the K_{pr} as in Eq. 6. When the same transaction is received by other entities in the framework, the K_{pb} is used from the blockchain of headers corresponding to D_i for transaction validation. On passing the validation the transaction gets added to the data block. This approach facilitates the transaction addition into the blocks without delay, as the waiting time for other transactions to arrive and then bundling them up into one block is not an issue with the proposed technique. The instantaneous addition is transaction is enabled by the architecture of assigning one specific block for each drone, whereas in the case of conventional blockchains the blocks contain transactions from heterogeneous entities.

5.3 Stochastic selective consensus

The existing blockchain schemes require all nodes of the blockchain network to participate in the consensus which is a challenging task [54] in resource-constrained environments like IoD. Hence in the proposed framework the transaction verification in the blockchain using the stochastic selective consensus algorithm is proposed. Valid voters are shortlisted using the voter validation algorithm. The distance of i^{th} drone (that performed the transaction) from the j^{th} drone is computed below [55]

$$d_{i \rightarrow j} = \left|\frac{d_i}{d}\right| \times d + \left|\frac{d_j}{d}\right| \times d + n_{i \rightarrow j} \times d \tag{7}$$

Algorithm 1 Selective consensus-based voter validation algorithm

Input: P_{pool}
Output: $S_{V_{voter}}$

```
for i do in n
    Call equation 7
    if d_{i→j} <= neighbourhood range then
        Set V_voter = true
    else
        Set V_voter = false
    end if
    Generate: R
    for r=0 do in V_voter
        if r < R − S_{V_voter}/V_voter then
            S_{V_voter} ← r
        end if
    end for
    Return S_{V_voter}
end for
```

Now, the Eq. 7 is used to create a pool of positions (P_{pool}), which is used as an input to the Algorithm 1 to shortlist the valid voter (V_{voter}). Fig. 7 shows the eligible and ineligible voters selected using the proposed stochastic selective consensus algorithm. The cooperative nodes that participate in the consensus are chosen randomly. The number of randomly chosen cooperative nodes (R) is not fixed and keeps on changing. Each node shortlisted using

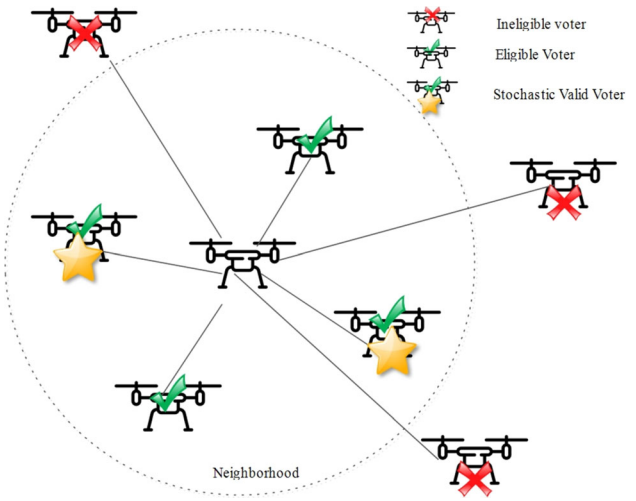

Fig. 7 Eligible and ineligible voter

Ground Station G_s	*Drone* D_i
Input: R_{ser} $R_{st} \leftarrow D_{gps}, B_s$	
$\xrightarrow{\langle R_{st} \rangle}$	
	$R_{res} \leftarrow K_{pr}(D_{gps}, B_s)$
	$\xleftarrow{\langle R_{res} \rangle}$
if $R_{res} \in R_{ser}^{SLA}$ Select D_i optimally $B_{D_i}^{G_s} \leftarrow K_{pb}(D_{gps}, S_{gps}, W_p,$ $I_p, D_n, F_p, B_{req}, F_u, H_{B_{D_i}}^{p})$	
$\xrightarrow{\langle B_{D_i} \rangle}$	
	$B_{D_i}^{G_s} \rightarrow K_{pr}(D_{gps}, S_{gps}, W_p,$ $I_p, D_n, F_p, B_{req}, F_u, H_{B_{D_i}}^{p})$ If $H_{B_{D_i}}^{p} == H_{B_{D_i}}^{c}, B_{req} < B_s$ Update $B_{D_i}^{l}$

Fig. 8 Dialog between ground station and drones for transaction signing

the Eq. 7 need not participate in the consensus. Only the nodes that are shortlisted by the algorithm 1 participate in the voting process. Each voter from V_{voter} pool has an equal probability of being selected in stochastic valid voter list $S_{V_{voter}}$. Suppose the event of a drone from V_{voter} being chosen as a voter is represented as E, then the probability of occurrence of E is represented by $P(E)$. If n number of drones are present in V_{voter} among which R drones are to be chosen for stochastic voting then the probability of each drone is R/n. Algorithm 1 returns the list of stochastic valid voters $S_{V_{voter}}$. The computational complexity of the selective consensus-based voter validation algorithm is $O(n * V_{voter})$ (Table 3).

5.4 Transaction signing

Drones are enabled to have their block in the proposed architecture. Drone requires the on-board storage of its block only, whereas the complete blocks of all drones are stored at the G_s and A_v. The G_s stores the blocks for those drones only which are being served through it. Further, G_s are responsible to keep the full copy of the blockchain. G_s uses the K_{pb} of D_i which results in secure data dissemination in the proposed architecture. The K_{pb} is fetched from the blockchain of headers by the G_s. The transaction signing process in the proposed model, among G_s and D_i, is represented in Fig. 8 as dialogue.

Table 3 Voter comparison

	Ineligible voter	V_{voter}	$S_{V_{voter}}$
Neighbourhood	–	✓	✓
Voting rights	–	✓	✓
Distance criteria	✓	✓	✓
Stochastic selection	–	–	✓

5.5 Shrinking blocks

The proposed technique aims at maintaining a fast and yet lightweight architecture of blockchain. It is achieved through the shrinking mechanism. The shrinking mechanism applies to B_{D_i} and does the job on every flight basis. When a flight completes successfully, the content of the block is hashed. Hence, D_i now holds only the hash of the previous transactions executed instead of complete data. Shrinking of blocks for D_i only happens on the drone, whereas the G_s and A_v always hold full versions of B_{D_i}. This enables the retrieval of information for past events and still keeps the minimum burden on the D_i. Block shrinking is performed through the formulae in Eq. 8.

$$Tx = \int_{t_n}^{t_{n+1}} Tx_j^{D_i} dt \tag{8}$$

Using Eq. (8), we compute $S_D(B_i(t_{n-1}))$ that is defined as below.

$$S_D(B_i(t_{n-1})) = sh(Tx) \tag{9}$$

Algorithm 2 Block Shrinking

INPUT: Tx_n
OUTPUT: $H^P_{B_{D_{n-1}}}$

1: INITIALIZE: $B_{D_i} \leftarrow \mathrm{H}^P_{B_{D_i}} = 0$
2: **while do** $B_{D_i} \in B_C$
3: **if** $n \neq 1$ **then**
4: **if** $Tx_n \rightarrow Tx_{n'}$ **then**
5: $B_{D_i}(Tx_n) \leftarrow Tx_{n'}$
6: **else**
7: Compute: $H(T_{n-1})$
8: $H^P_{B_{D_{n-1}}} \leftarrow H(T_{n-1})$
9: **end if**
10: **end if**
11: **end while**

An algorithm has been designed for this mechanism which is presented in Algorithm 2. The D_i receives data in form of n^{th} transaction Tx_n. The algorithm gets initialized by setting the hash of previous block($H^P_{B_{D_i}}$) to zero to mark the beginning of the block. it is checked is the drone holds any previous data. $H^P_{B_{D_i}}$ is inserted into the block (B_{D_i}) of D_i B_{D-i}. Further the B_{D_i} is authenticated on blockchain (B_C). If the Tx_n is the first transaction for the drone, it is added into the B_{D-i}'s Tx'_n part. In other case when there are existing older transactions on drone, they are hashed into $H^P_{B_{D_i}}$ of D_i header part and then the data is added to the Tx'_n. The block shrinking mechanism has the linear complexity represented by $O(T_{X_n})$.

6 Caching mechanism

In an IoD environment, the availability of required information with a small delay is a necessity. To achieve the low delay requirement caching is highly useful [53]. Moreover, caching is helpful to solve the bottleneck problem by reducing the requirement of the number of requests made to the A_v. The caching at various levels helps achieve low latency and high performance in data dissemination. In the proposed work caching is performed at the following levels as discussed below.

6.1 Edge to edge caching

Ground stations (G_s) are the edge nodes in the proposed work. G_s are the access points through which the drones access various services of the framework and also disseminate data to drones. G_s can cater to the service requirements via the cached data. Due to the high mobility operations of drones, they move from the service area of one G_s to another. For implementing edge-to-edge caching the size of the data unit matters as suggested in [2]. In the proposed model the edges store the complete B_C so that whenever a drone is being serviced by the edge, requires data it can be served readily. As the lightweight blockchain is of the header part of blocks of drones, the corresponding data part is maintained in a local cache at the ground station. With the least recently used (*LRU*) block data being aged out of the cache of the ground station. The master copy is retained by A_v at all times. The aging out is performed based on cache size C_s. The request rate for a D_i's $B^{G_s}_{D_i}$ at G_{s_i} is $Q(D_i)$. The rate $Q(D_i)$ signifies the popularity of the $B^{G_s}_{D_i}$. [2] defines the t_{C_s} as the time it takes to fill the C_s with unique $B^{G_s}_{D_i}$.

$$C_s = \sum_{u=1}^{N} (1 - e^{-Q(u)t}) \tag{10}$$

The probability of not finding $B^{G_s}_{D_i}$ for D_i is $pm(D_i)$:

$$pm(D_i) \approx e^{-Q(D_i)t_{C_s}} \tag{11}$$

Edge to edge caching is effective in dealing with the spatial locality in the proposed framework as the D_i move among G_S. As suggested in [43], the caching among G_s is performed based on the route a drone will follow during its flight. The wireless cellular network is divided into cells. Whenever the G_{s_n} performs the drone handoff to $G_{s_{n+1}}$ the cache content corresponding to D_i is synchronised with the cache of the ground station ($G_{S_{n+1}}$) which is now serving the D_i. As depicted in Fig. 9 when the D_i is being served through the G_{S_n} it does the caching of data fetched from cloud-based A_v to improve latency, access time and reduce the communication cost. As during the flight D_i moves to the zone that is served through $G_{s_{n+1}}$, instead of $G_{s_{n+1}}$ again contacting A_v for D_i data directly at first the cache from G_{s_n} are copied into its cache and responses are given to the requests coming from G_{s_n}.

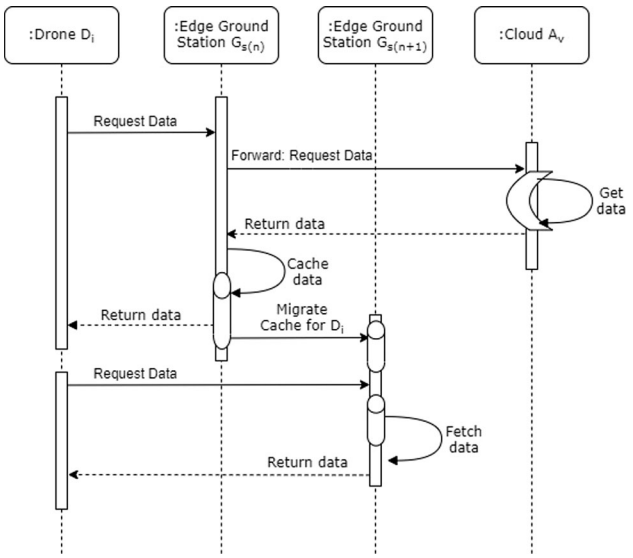

Fig. 9 Edge to edge caching

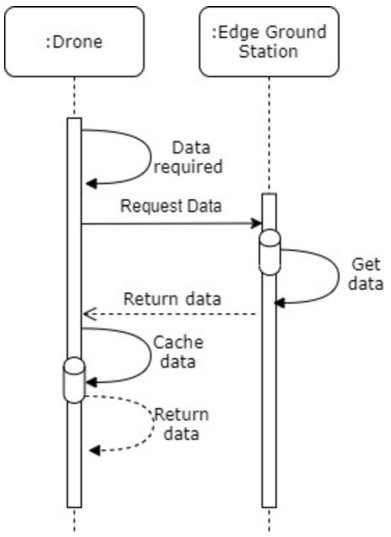

Fig. 10 Edge to drone caching

6.2 Drone to edge caching

To further improve the system performance, drone-to-edge (*D2E*) caching is performed. The *D2E* caching helps in providing drones with required data with the minimum delay. A G_s caters to multiple drones. Each drone in the proposed ecosystem has its block of information in form of blockchain. Only the headers of the drone blocks are chained together to keep the blockchain lightweight. Whenever, the drone wants data, the same is fetched from the cache maintained in G_s serving the drone. Moreover, the bandwidth available in *D2E* is an important aspect for development of effective caching policy. The bandwidth between the D_i and G_s is $B_{D_i \leftrightarrow G_s}$. This $B_{D_i \leftrightarrow G_s}$. As in [53] the maximum number of data chunks (M_l) that can be sent from G_S to D_i is given by

$$M_l = \lfloor \frac{t_s * B_{D_i \leftrightarrow G_s}}{L} \rfloor \tag{12}$$

Whether the data is fetched from the G_s or it is fetched from the cached data available on the drone is represented by z.

$$z = \begin{cases} 0 & D_i \text{ gets data from } G_s \\ 1 & D_i \text{ uses cached data} \end{cases} \tag{13}$$

As depicted in Fig. 10, when the D_i requires any data, it first checks its cache. If the data as per request is not available in its cache the request is escalated to the edge ground station G_s that is serving the zone in which D_i is present. G_s fetches the data from its cache and returns to the D_i. On receiving the data D_i caches it. Then the entry from the cache is returned to the requesting process.

6.3 Drone drone caching

In the proposed architecture the range extender drones serve the purpose of extending the services to drones that can not be directly serviced by the ground station. In such scenarios, the range extender drones help in improving the overall communication cost by caching the content for the drones being serviced through it. The data is available with neighboring drones can be fetched in time effective manner. This can be done by using the drone to drone caching. Whenever a D_i enters into the service area S_{G_s} it has to acquire the blockchain of its concern i.e the selective consensus needs the miners to have the transaction for signing. The transaction information is forwarded by D_i to $D_{(n-i)}$ in S_{G_s}. As depicted in Fig. 11 the drone D_i is the one that is being served by range extender drone D_{re_i}. When D_i sends data request to D_{re_i} if the data is available within the cache of D_{re_i} it is send back to the D_i as represented in case 1. In case when the data is not directly available with D_{re_i} and it is taking another range extender drone $D_{re_{i_1}}$ to form the flying ad-hoc network, then the data request from D_i is escalated by D_{re_i} to $D_{re_{i_1}}$ for its potential availability in the cache. This saves the overall communication cost and improves the latency.

7 Results and evaluation

The proposed framework has been validated and verified in different phases. Firstly, the system evaluation has been performed based on computational and communication costs, calculated numerically for various operations and interactions. Secondly, the simulations were performed to

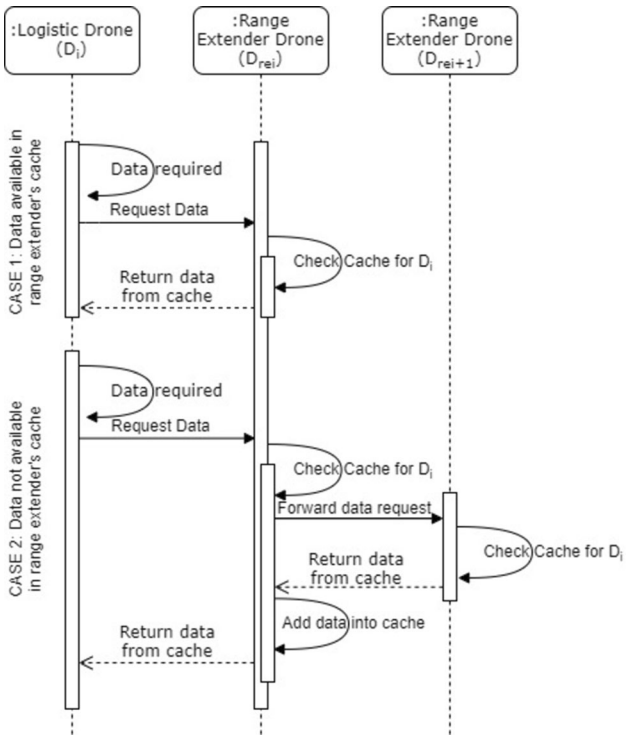

Fig. 11 Drone to drone caching

evaluate the performance of the proposed derived architecture in the IoD ecosystem. IoD has been simulated with a bunch of tools that work together to create new to real-time deployments. Lastly, the proposed model is analyzed for its resistance against various security risks and compared to some existing techniques in a similar field.

7.1 Numerical evaluation

Proposed technique has been evaluated for the performance based on communication and computational cost. Broadly the proposed model can be segregated in two ways, new drone registrations and registered drones coordinating with G_s and A_v for its functioning.

7.1.1 Computational cost

The associated computational cost with the proposed technique is explored as below:

- *Drone Registration Phase* The process of drone registration includes a XOR operation among *pwd* and E_k, append operation is performed four times on $K_{pb}, D_{gps}, D_{role}, D_i$. As soon as the R_{req} is received by A_v, it performs a search operation on B_C for D_i *and* K_{pb}. The search is linear in nature resulting to two linear searches on data of magnitude n. If A_v is able to trace the R_{req} on B_C, it further authenticates the drone. Drones hashed firmware from TPM, this operation requires one SHA-1 hash operation. Further, the P_{pool} is created for which Eq. 7 is calculated involving three multiplication, two division, to modulus, and two addition operations. For co-relating theR_n and $d_{i\rightarrow j}$ one comparison operation is performed and three more comparisons are required. The addition of D_i into $V_l[i]$ requires one assignment operation. But, if the R_{req} cannot be traced in B_C, the new block $B_{D_i}^{new}$ creation is started which needs six append operations.
- *Drone Operation Phase* When a registered drone D_i needs any operation performed it does the data retrieval through the B_c. For which it need to prepare R_res and R_{ser}, where the R_res requires one RSA encryption operation. Then the matching of criteria is verified through a comparison operation. Then D_i is optimally chosen for which the available drones are sorted ascending on metric as in $B_{D_i}^{G_s}$. The $B_{D_i}^{G_s}$ needs to perform the append operation on nine entities. On the drone front, two comparison operations are performed to validate the block consistency at D_i and G_s. Once the drones are operational the block shrinking mechanism also starts working which needs one search operation on all blocks in B_C. If the drone doesn't have legacy data it requires one assignment operation else it requires one SHA-1 hashing operation along with one assignment operation.

Hence, the architecture needs one XOR, ten comparisons, two SHA-1, one RSA, two sortings, two divisions, three multiplications, two modulus, two additions, three assignments, and twenty-one append operations as a whole.

7.1.2 Communication cost

Communication cost is the numerical measurement of the data transfer requirement of the proposed model across the network excluding the overhead of the underlying communication protocols. The cost is computed as follows:

- Registration Phase: In this phase D_{gps} of 24 bits, K_pb of 256 bits, D_{role} of 3 bits and D_i of 64 bits are transferred between D_i and A_v. In response for the same 128 bits D_i, 256bits K_{pb},3 bits D_{role} ,20 bits t_s, 7 bits $R_{cap}$and 16 bits R_W is reverted by A_v to D_i.
- Drone Operations: In the operational phase the data transfer of 7 bit B_s and 24 bit D_{gps} occurs to G_s from D_i. A 31 bit reply is sent to D_i from G_s. S_{gps} 30 bits, D_{gps} 30 bits, I_p of 512 bits, W_p 16 bits, D_n of 4 bits, B_{req} of 7 bits, F_u of 1 bit, $H_{B_{D_i}}^{p}$ of 256 bits F_p of 256 bits are transferred from G_s to D_i in the form on new transaction. .

This brings the overall communication cost of the model is 1951 bits. Which favors the task of reducing overhead incurred to the bare minimum through the proposed technique.

7.2 Simulation experiments

The proposed architecture is heterogeneous and hence one single tool is not available to perform the experiments. Hence, to overcome this challenge, the simulated validations have been done in two parts. The first part covers the IoD implementations and the second part covers the implementation of derived blockchain. The detailed environmental setups for these are given below:

7.2.1 IoD and drone swarm simulation setup

The proposed model has been evaluated for its IoD capabilities using the simulated environment. As mobility is a very major factor in IoD, so simulation environment has been set up using Gazebo. Fig. 12 depicts the real-world scenario creation using the Gazebo environment. Here, *runway.world* simulation model has been used to provide the swarm of drones with the geographical conditions as depicted in Fig. 13. The simulated experiments help to deploy the proposed IoD ecosystem and the underlying modules are controlled programmatically thus enabling the integration of heterogeneous technologies. The proposed framework aims to address the validity of the research questions through mobility-based experimental setup in the IoD ecosystem.

Further, for emulation of the flight board, the Ardupilot has been used (refer. Figs. 14 and 15). The ground station capabilities have been implemented using QGroundControl. The links of drones to the ground station have been set up using MAVlink. The robotic operating system (ROS) works at the core to enable intercommunication between all these components. The environment has

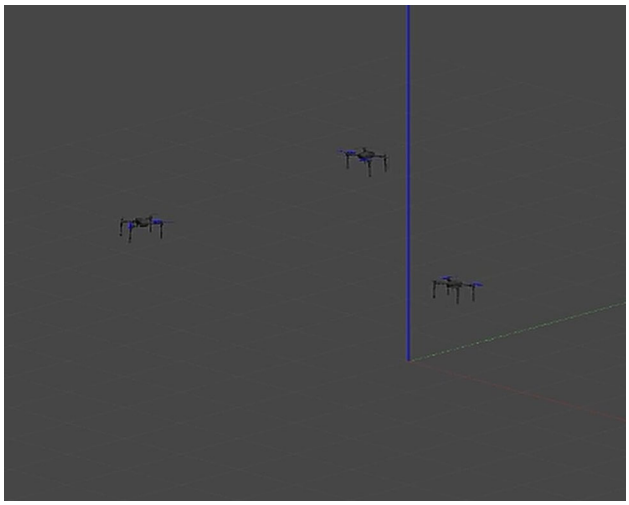

Fig. 13 Drone swarm in gazebo GUI

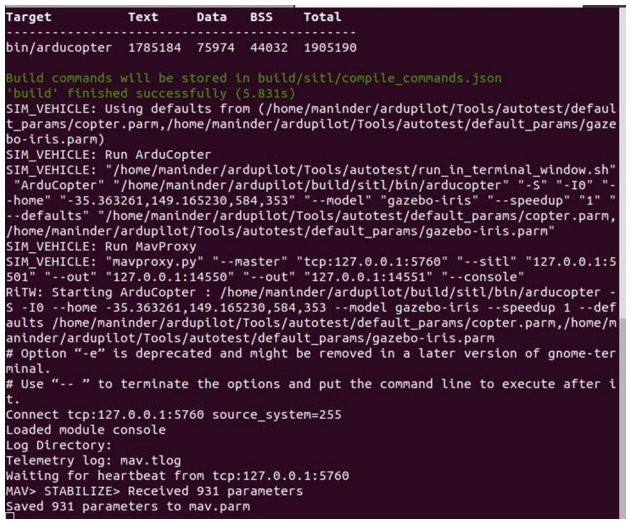

Fig. 14 Ardupilot flight control

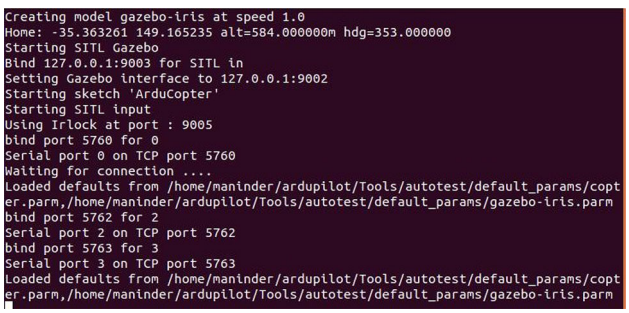

Fig. 15 Ardupilot flight control terminal

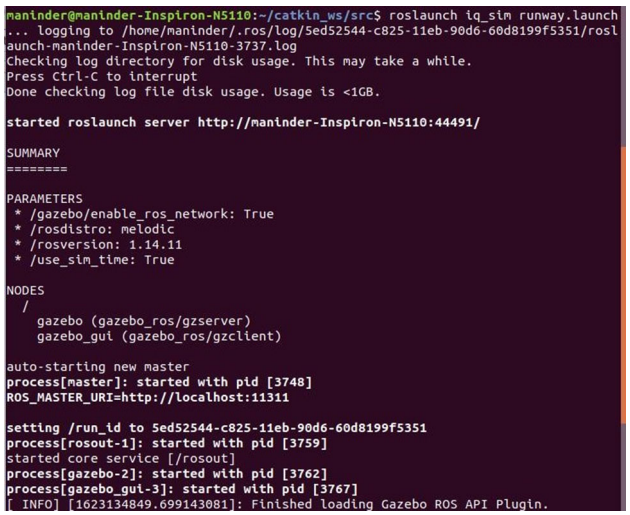

Fig. 12 Gazebo through robotic operating system

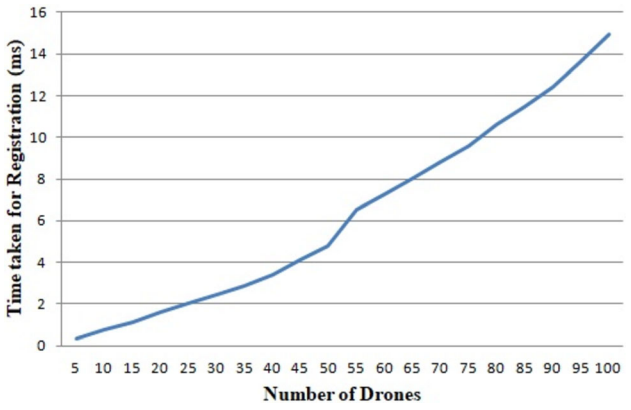

Fig. 16 Drone identity generation time

been set up in Ubuntu 18.04 on a physical system (with Intel i5-2450m processor-3 GHz, with 8 GB of RAM, NVIDIA 525M GPU with 1GB memory). Based on the system configuration the swarm size is maximized.

7.2.2 Result and discussions

As proposed in the framework, the data is offloaded into B_{D_i}'s of D_i. Then, D_i reads the data from B_{D_i} and ardupilot performs the operations accordingly. The performance of the blockchain architecture is validated based on various factors like, identity generation time, block generation time, block generation delay, voter selection time, and shrinking block size. The first horizon of blockchain validation is the time taken for D_i generation, i.e. identity generation, concerning the number of drones requesting for registration simultaneously. The experimental results depicted in Fig. 16 suggests the proposed approach shows almost linear growth in the amount of time required for D_i generation up to 45 drones. Thus it shows that the time taken for the identity generation of each requesting drone is not substantial. After this, the D_i time is marginally elevated than the initial part but this is just a 1% growth that almost negligible. The drone identity generation time helps to validate the key question of providing a lightweight blockchain mechanism for managing access control and identity validation in IoD.

Further, the proposed work was compared with the existing work in [56] in context to the identity generation time. The existing proposal uses the blockchain 1.0 in contrast to the derived blockchain architecture used in the proposed work. The results show the proposed technique exhibits timely and faster registrations than the existing variant. The findings are depicted in Fig. 17. This comparison depict that the superiority of the proposed approach in terms of achieving the goal related to the design of a lightweight blockchain architecture for edge-envisioned IoD ecosystem.

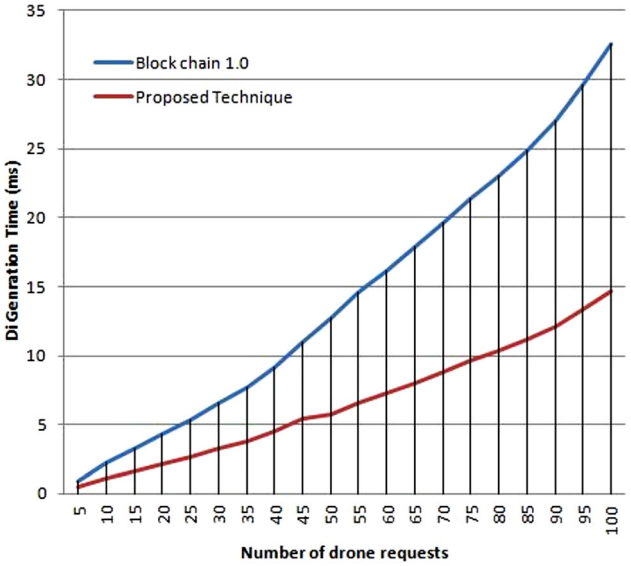

Fig. 17 Drone identity generation time comparison

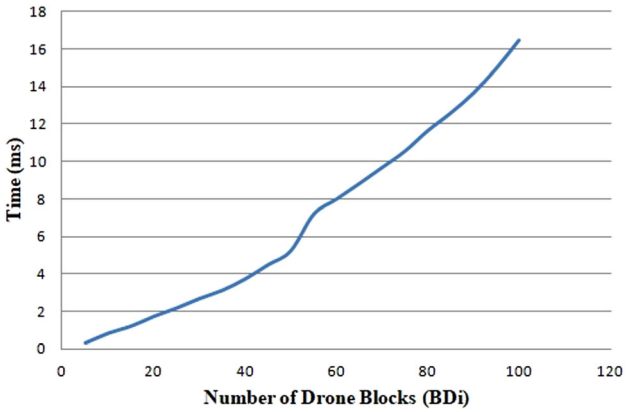

Fig. 18 Drone block generation time

In the proposed framework, we have designed an unconventional block structure specifically for drones. So, to understand the usability and benefits of this derived block structure, we have validated the time taken to generate the drone blocks. The time required for generation of B_{D_i} for D_i in the simulated scenario is shown in Fig. 18. It clearly shows a reasonable time growth to generate new blocks in contrast to an increase in the number of drones. The time taken to generate new drone blocks has marginal increase after 50 drones but it stabilizes thereafter.

Now, it is very important to analyze the impact of delay on the block generation time using the proposed approach. Thus, the validity of unconventional block structure is further validated for delays emerging as a result of simultaneous drone registration requests under different load patterns. Fig. 19 shows the pattern of delay incurred during the time drone blocks are generated for heterogeneous

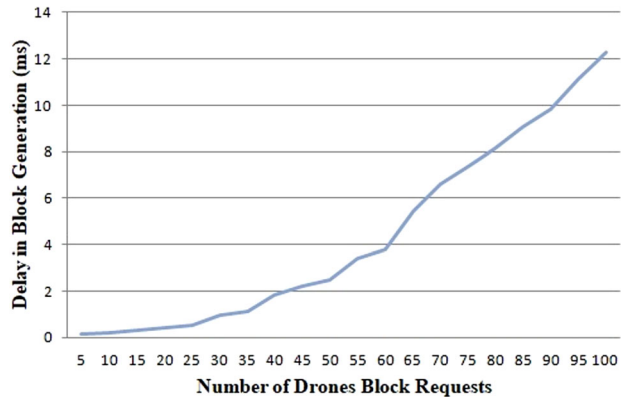

Fig. 19 Delay in drone block generation

requests. The outcome depict that the proposed work performs fairly well with a small delay concerning an increase in the number of requests.

To design a lightweight blockchain architecture, we strongly rely on the proposed consensus mechanism that choose miners in a stochastic manner, ensuring distributed yet timely consensus. The performance of the proposed stochastic selective voter selection mechanism is evaluated and shown in Fig. 20. The results indicate that the time to choose the voter is less when the number of voters is small in number. It shows a tendency to increase gradually with the number of participating drones. The existing blockchain frameworks have limited customization options to their architecture for which custom derived blockchain was developed and deployed. The proposed work also utilize a shrinking mechanism for block data to keep a check on the increasing demand for storage space as an answer to the one of the key research questions seeking a check on expanding storage requirements of blockchain over time. The block shrinking mechanism is validated and compared with a scenario where no shrinking mechanism is adopted. The performance is measured in terms of the block size when new transactions are amended into the block and

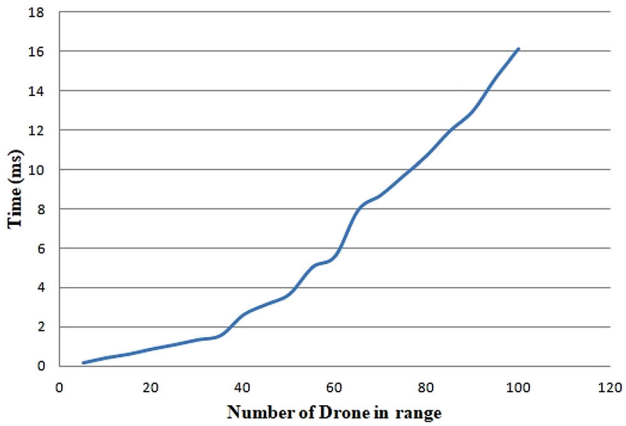

Fig. 20 Stochastic selective voter selection time

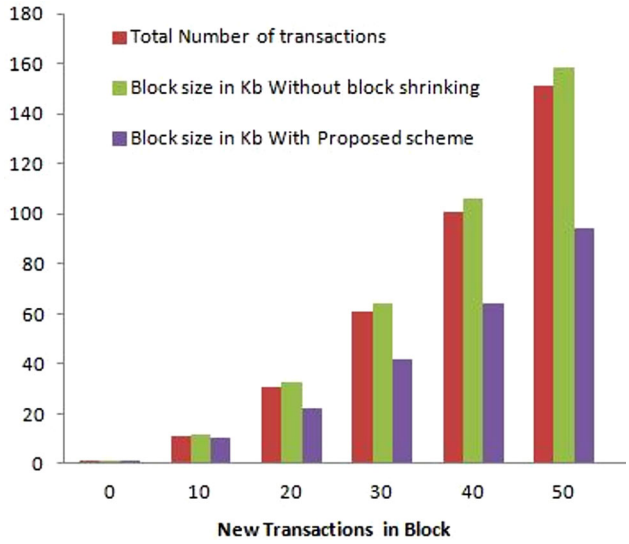

Fig. 21 Shrinking block performance analysis

Table 4 Resistance to various security concerns

Attack	[57]	[58]	Proposed architecture
Sybil attack	✓	–	✓
GPS spoofing	✓	✓	✓
Data manipulation	✓	✓	✓
Gray hole attack	–	–	✓
Black hole attack	–	–	✓
Hardware Trojan	–	–	✓
Falsified information ejection	✓	✓	✓

analyzed in concern with an increase in the number of transactions. The results are depicted in Fig. 21. The trend depicted in this figure indicates that the size requirement for proposed approach is comparatively lower than the non-shrinking block architecture over time.

7.3 Comparative analysis

The proposed architecture has been compared for its resistance against various attacks on drones with existing works in Table 4. The proposed technique is comparatively more resistant to various security concerns in contrast to the other works.

The proposed technique is an amalgamation of various techniques and technologies. The computational complexity of the various key contributions is compared with existing similar proposals and is presented in Table 5.

The results of the system show the competitiveness of the proposed model for the IoD ecosystem but there are

Table 5 Comparison of computational complexities

Method	[57]	[56]	Proposed architecture
Drone registration	$O(n^3)$	$O(n^3)$	$O(n)$
Miner selection	–	$O(n)$	$O(n * m)$
Block compression	–	–	$O(n)$

some limitations of the work. The proposed system is based on the limited mobility scenarios through simulated environments thus the evaluation in a realistic scenario is still required. Further, the miner stability has an impact on the system and this aspect must be further analyzed.

8 Conclusion

The technique proposed in this paper focuses on the adaptation of blockchain in the IoD ecosystem. As blockchain is an unconventional security mechanism, it has its limitations for adoption into IoD. In this paper, the blockchain technique which brings in a lightweight architecture suitable for IoD is designed and developed. The proposed architecture provides the advantage of secure data dissemination along with minimalist overhead that is present in conventional blockchain to maintain the blockchain. The work progresses in the horizon where the data is being decoupled from blockchain and hence shunting offload in terms of storage requirement from light nodes. Usage of TPM in the technique brings the chip-level security to the security keys stored on drones as if the drones are high jacked an attacker tries to manipulate its firm aware it can be tracked. Further, the blockchain itself is special as the data is not included in the blockchain as such. Only headers of blocks form the blockchain which makes the execution, synchronization as well as communication more efficient. The data doesn't pile up inside the drones is checked through the shrinking block mechanism along with the stochastic selective voter selection mechanism to choose the voters. This probabilistic technique fairly well counters possible attacks by manipulating voters. Further, the model incorporates caching at various levels to provide real-time experience to the ecosystem. The proposed ecosystem has been validated for its performance in terms of communication costs and computation costs. The simulated results favor the ecosystem's adaptability to IoD. In the future, the proposed work would be implemented on a practical scaled version of the IoD ecosystem to analyze the system behavior in real-life scenarios.

Author Contributions All authors contributed to the concept and design of this work. Manuscript preparation was performed by MS and GSA. The experimental evaluation and analysis were performed by MS. All authors edited the manuscript. All authors have read and approved the final manuscript.

Funding This work was partially supported by Start-up Fund received from the Durham University, UK [Grant No. 090614].

Data availability Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Informed consent Not applicable

Compliance with Ethical Standards The authors have complied with the ethical standards.

References

1. Dikmen, M., Burns, C.: Trust in autonomous vehicles: the case of tesla autopilot and summon. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017, pp. 1093–1098
2. Dabirmoghaddam, A., Barijough, M.M., Garcia-Luna-Aceves, J.: Understanding optimal caching and opportunistic caching at "the edge" of information-centric networks. In: Proceedings of the 1st ACM Conference on Information-Centric Networking, ser. ACM-ICN '14. Association for Computing Machinery, New York, NY, USA, pp. 47–56 (2014). https://doi.org/10.1145/2660129.2660143
3. Vashist, S., Jain, S.: Location-aware network of drones for consumer applications: supporting efficient management between multiple drones. IEEE Consumer Electron. Mag. **8**(3), 68–73 (2019)
4. Gharibi, M., Boutaba, R., Waslander, S.L.: Internet of drones. IEEE. Access **4**, 1148–1162 (2016)
5. 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. **56**(1), 64–69 (2018)
6. Lakhan, A., Mastoi, Q-U-A., Elhoseny, M., Memon, M.S., Mohammed, M.: Deep neural network-based application partitioning and scheduling for hospitals and medical enterprises using iot assisted mobile fog cloud. Enterprise Information System 02 (2021)
7. Lakhan, A., Memon, M.S., ul-ain Mastoi, Q., Elhoseny, M., Mohammed, M.A., Qabulio, M., Abdel-Basset, M.: Cost-efficient mobility offloading and task scheduling for microservices IoVT

applications in container-based fog cloud network. Clust. Comput. (2021). https://doi.org/10.1007/s10586-021-03333-0
8. Lakhan, A., Mohammed, M.A., Rashid, A.N., Kadry, S., Panityakul, T., Abdulkareem, K.H., Thinnukool, O.: Smart-contract aware ethereum and client-fog-cloud healthcare system. Sensors **21**(12), 4093 (2021)
9. Mutlag, A.A., Khanapi Abd Ghani, M., Mohammed, M.A., Maashi, M.S., Mohd, O., Mostafa, S.A., Abdulkareem, K.H., Marques, G., de la Torre Díez, I.: Mafc: multi-agent fog computing model for healthcare critical tasks management. Sensors 20(7), 1853 (2020)
10. Mostafa, S., Mustapha, A., Saraswathy, S., Mohammed, M., Parwekar, P., Kadry, S.: An agent architecture for autonomous uav flight control in object classification and recognition missions. Soft Comput. 02 (2021)
11. Zhang, Y., He, D., Li, L., Chen, B.: A lightweight authentication and key agreement scheme for internet of drones. Comput. Commun. **154**, 455–464 (2020)
12. Mohanta, B., Jena, D., Satapathy, U., Patnaik, S.: Survey on iot security: challenges and solution using machine learning, artificial intelligence and blockchain technology. Intern. Things **11**, 100227 (2020)
13. Garg, S., Aujla, G.S., Erbad, A., Rodrigues, J.J., Chen, M., Wang, X.: Guest editorial: Blockchain envisioned drones: Realizing 5g-enabled flying automation. IEEE Network **35**(1), 16–19 (2021)
14. Aujla, G.S., Singh, M., Bose, A., Kumar, N., Han, G., Buyya, R.: Blocksdn: Blockchain-as-a-service for software defined networking in smart city applications. IEEE Netw. **34**(2), 83–91 (2020)
15. Dorri, A., Steger, M., Kanhere, S.S., Jurdak, R.: Blockchain: A distributed solution to automotive security and privacy. IEEE Commun. Mag. **55**(12), 119–125 (2017)
16. Dorri, A., Kanhere, S.S., Jurdak, R., Gauravaram, P.: Lsb: A lightweight scalable blockchain for iot security and anonymity. J. Parallel Distrib. Comput. **134**, 180–197 (2019)
17. Yazdinejad, A., Parizi, R.M., Dehghantanha, A., Karimipour, H., Srivastava, G., Aledhari, M.: Enabling drones in the internet of things with decentralized blockchain-based security. IEEE Internet of Things J. **8**, 6406 (2020)
18. Bera, B., Chattaraj, D., Das, A.K.: Designing secure blockchain-based access control scheme in iot-enabled internet of drones deployment. Comput. Commun. **153**, 229–249 (2020)
19. Aggarwal, S., Chaudhary, R., Aujla, G.S., Kumar, N., Choo, K.-K.R., Zomaya, A.Y.: Blockchain for smart communities: applications, challenges and opportunities. J. Netw. Comput. Appl. **144**, 13 (2019)
20. Jindal, A., Aujla, G.S.S., Kumar, N., Villari, M.: Guardian: blockchain-based secure demand response management in smart grid system. IEEE Trans. Serv. Comput. **13**, 613 (2019)
21. Xu, C., Wang, K., Li, P., Guo, S., Luo, J., Ye, B., Guo, M.: Making big data open in edges: a resource-efficient blockchain-based approach. IEEE Trans. Parallel Distrib. Syst. **30**, 870 (2018)
22. Feng, C., Yu, K., Bashir, A.K., Al-Otaibi, Y.D., Lu, Y., Chen, S., Zhang, D.: Efficient and secure data sharing for 5g flying drones: a blockchain-enabled approach. IEEE Netw. **35**(1), 130–137 (2021)
23. Michelin, R., Dorri, A., Steger, M., Lunardi, R., Kanhere, S., Jurdak, R., Zorzo, A.: Speedychain: a framework for decoupling data from blockchain for smart cities, pp. 145–154 (2018)
24. Ge, C., Ma, X., Liu, Z.: A semi-autonomous distributed blockchain-based framework for uavs system. J. Syst. Archit. **107**, 101728 (2020)
25. Allouch, A., Cheikhrouhou, O., Koubâa, A., Toumi, K., Khalgui, M., Nguyen Gia, T.: Utm-chain: blockchain-based secure unmanned traffic management for internet of drones. Sensors **21**(9), 3049 (2021)
26. Kang, J., Xiong, Z., Niyato, D., Ye, D., Kim, D.I., Zhao, J.: Toward secure blockchain-enabled internet of vehicles: optimizing consensus management using reputation and contract theory. IEEE Trans. Veh. Technol. **68**(3), 2906–2920 (2019)
27. Zhu, L., Chen, C., Su, Z., Chen, W., Li, T., Yu, Z.: Bbs: Microarchitecture benchmarking blockchain systems through machine learning and fuzzy set. In: IEEE International Symposium on High Performance Computer Architecture (HPCA), vol. 2020, pp. 411–423 (2020)
28. Imani, M., Ghoreishi, S.F.: Graph-based bayesian optimization for large-scale objective-based experimental design. IEEE Trans. Neural Netw. Learn. Syst. 1–13 (2021)
29. Singh, M., Aujla, G.S., Bali, R.S.: Odob: One drone one block-based lightweight blockchain architecture for internet of drones. In: IEEE INFOCOM 2020—IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 249–254 (2020)
30. Wang, L., Hu, B., Chen, S.: Energy efficient placement of a drone base station for minimum required transmit power. IEEE Wirel. Commun. Lett. **9**, 2010–2014 (2018)
31. Koubâa, A., Qureshi, B., Sriti, M.-F., Allouch, A., Javed. Y., Alajlan, M., Cheikhrouhou, O., Khalgui, M., Tovar, E.: Dronemap planner: a service-oriented cloud-based management system for the internet-of-drones. Ad Hoc Netw. 86, 46–62 (2019)
32. Pu, C., Carpenter, L.: *psched*: a priority-based service scheduling scheme for the internet of drones. IEEE Syst. J. 1–10 (2020)
33. Amer, K., Samy, M., Shaker, M., ElHelw, M.: Deep convolutional neural network based autonomous drone navigation. In: Osten, W., Nikolaev, D.P., Zhou, J. (eds.) Thirteenth International Conference on Machine Vision, vol. 11605, pp. 16–24. International Society for Optics and Photonics. SPIE (2021)
34. Choudhary, G., Sharma, V., Gupta, T., Kim, J., You, I.: Internet of drones (iod): threats, vulnerability, and security perspectives. arXiv preprint arXiv:1808.00203 (2018)
35. Ilgi, G.S., Kirsal Ever, Y.: Chapter eleven—critical analysis of security and privacy challenges for the internet of drones: a survey. In: Al-Turjman, F. (ed.) Drones in Smart-Cities, Elsevier, pp. 207–214 (2020)
36. Chen, Y.-J., Wang, L.-C.: Privacy protection for internet of drones: A network coding approach. IEEE Internet Things J. **6**(2), 1719–1730 (2019)
37. Vattapparamban, E., Guvenc, I., Yurekli, A.I., Akkaya, K., Uluagac, S.: Drones for smart cities: Issues in cybersecurity, privacy, and public safety. In: 2016 International Wireless Communications and Mobile Computing Conference (IWCMC), pp. 216-221 (2016)
38. Dawaliby, S., Aberkane, A., Bradai, A.: Blockchain-based iot platform for autonomous drone operations management. In: ser. DroneCom '20. Association for Computing Machinery, New York, NY, USA, pp. 31–36 (2020)
39. Ossamah, A.: Blockchain as a solution to drone cybersecurity. In: 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), pp. 1–9 (2020)
40. Dai, H.-N., Zheng, Z., Zhang, Y.: Blockchain for internet of things: A survey. IEEE Internet of Things J. **6**(5), 8076–8094 (2019)
41. Ieee standard for framework of blockchain-based internet of things (iot) data management. IEEE Std 2144.1-2020, pp. 1–20 (2021)
42. Guo, S., Dai, Y., Guo, S., Qiu, X., Qi, F.: Blockchain meets edge computing: Stackelberg game and double auction based task offloading for mobile blockchain. IEEE Trans. Veh. Technol. **69**(5), 5549–5561 (2020)

43. Aujla, G.S., Singh, A., Singh, M., Sharma, S., Kumar, N., Choo, K.R.: Blocked: Blockchain-based secure data processing framework in edge envisioned v2x environment. IEEE Trans. Veh. Technol. **69**(6), 5850–5863 (2020)
44. Sharma, V., You, I., Jayakody, D.N.K., Reina, D.G., Choo, K.K.R.: Neural-blockchain-based ultrareliable caching for edge-enabled uav networks. IEEE Trans. Industr. Inf. **15**(10), 5723–5736 (2019)
45. Zorzo, A., Nunes, H., Lunardi, R., Michelin, R., Kanhere, S.: Dependable iot using blockchain-based technology. 10 (2018)
46. Bera, B., Saha, S., Das, A.K., Kumar, N., Lorenz, P., Alazab, M.: Blockchain-envisioned secure data delivery and collection scheme for 5g-based iot-enabled internet of drones environment. IEEE Trans. Veh. Technol. **69**(8), 9097–9111 (2020)
47. Yazdinejad, A., Parizi, R.M., Dehghantanha, A., Karimipour, H., Srivastava, G., Aledhari, M.: Enabling drones in the internet of things with decentralized blockchain-based security. IEEE Internet Things J. **8**(8), 6406–6415 (2021)
48. Bera, B., Das, A.K., Sutrala, A.K.: Private blockchain-based access control mechanism for unauthorized uav detection and mitigation in internet of drones environment. Comput. Commun. **166**, 91–109 (2021)
49. Boukoberine, M.N., Zhou, Z., Benbouzid, M.: A critical review on unmanned aerial vehicles power supply and energy management: Solutions, strategies, and prospects. Appl. Energy **255**, 113823 (2019)
50. Chriki, A., Touati, H., Snoussi, H., Kamoun, F.: Fanet: communication, mobility models and security issues. Comput. Netw. **163**, 106877 (2019)
51. Aggarwal, S., Shojafar, M., Kumar, N., Conti, M.: A new secure data dissemination model in internet of drones. In: ICC 2019 - 2019 IEEE International Conference on Communications (ICC), 2019, pp. 1–6
52. Gupta, R., Kumari, A., Tanwar, S.: A taxonomy of blockchain envisioned edge-as-a-connected autonomous vehicles. Trans. Emerg. Telecommun. Technol. **32**, e4009 (2021)
53. Yang, L., Zhang, L., He, Z., Cao, J., Wu, W.: Efficient hybrid data dissemination for edge-assisted automated driving. IEEE Internet Things J. **7**(1), 148–159 (2020)
54. Chen, Y., Wang, L., Wang, S.: Stochastic blockchain for iot data integrity. IEEE Trans. Netw. Sci. Eng. **7**(1), 373–384 (2020)
55. Jindal, A., Aujla, G.S., Kumar, N.: SURVIVOR: a blockchain based edge-as-a-service framework for secure energy trading in sdn-enabled vehicle-to-grid environment. Comput. Netw. **153**, 36–48 (2019). https://doi.org/10.1016/j.comnet.2019.02.002
56. Singh, M., Aujla, G.S., Bali, R.S.: A deep learning-based blockchain mechanism for secure internet of drones environment. IEEE Trans. Intell. Transp. Syst. **22**(7), 4404–4413 (2021)
57. He, S., Wu, Q., Liu, J., Hu, W., Qin, B., Li, Y.-N.: Secure communications in unmanned aerial vehicle network. In: Liu, J.K., Samarati, P. (eds.) Information Security Practice and Experience, pp. 601–620. Springer, Cham (2017)
58. Vanitha, N., Padmavathi, G.: A comparative study on communication architecture of unmanned aerial vehicles and security analysis of false data dissemination attacks. In: International Conference on Current Trends towards Converging Technologies (ICCTCT), vol. 2018, pp. 1–8 (2018)

Maninderpal Singh is working toward his Ph.D. in the Computer Science and Engineering Department, Chandigarh University, India, where is also an assistant professor. He received his M.Tech. from the Computer Science and Engineering Department, Lovely Professional University, India, in 2013. He received his B.Tech. from the Computer Science and Engineering Department, Punjab Technical University, India, in 2010.

Gagangeet Singh Aujla is working as an Assistant Professor of Computer Science at Durham University, UK. Before this, he worked as a post-doctoral research associate at Newcastle University, a research associate at Thapar University (India), a visiting researcher at University of Klagenfurt (Austria) and on various academic positions for more than a decade. He received my Ph.D. degree from the Thapar University (India), my Master and Bachelor degrees from the Punjab Technical University (India). For his contributions to the area of scalable and sustainable computing, he was awarded the 2018 IEEE TCSC Outstanding Ph.D. Dissertation Award of Excellence. He is the recipient of 2021 IEEE System Journal Best Paper Award. The main theme of his research is energy-efficient, resilient and intelligent surfaces (smart city, smart grid, IoT-Edge-Cloud systems, healthcare systems, transportation systems). He worked on various research projects awarded by EPSRC, the Department of Science and Technology (India), and the Austrian Federal Ministry of Education, Science and Research. This facilitated him to collaborate with international researchers from the UK, US, Canada, Australia, China, Austria, Brazil, and India. Together, he published several research papers in the top tier IEEE journals (like, IEEE TKDE, IEEE TDSC, IEEE TVT, IEEE TITS, IEEE TSC, IEEE TCC, IEEE TII, IEEE JSAC, IEEE IoTJ, IEEE System Journal, IEEE Communication Magazine, IEEE Network, IEEE Consumer Electronics Magazine, IEEE Internet Computing) and conferences (like, IEEE ICC, IEEE Globecom, IEEE WoWMoM, IEEE Infocom, ACM Mobicom, ACM MobiHoc, IEEE IC2E). He led the team organizing workshops (SecSDN and BlockSecSDN) in conjunction with different IEEE Communication Society conferences like IEEE Infocom, IEEE Globecom, IEEE ICC, IEEE PiCom. He is also leading a workshop series on 'Blockchain for Cyberphysical Systems' in conjunction with ACM/IEEE UCC. He is an Area Editor of Adhoc Networks (Elsevier) and a Topic Editor for Sensors. He has been a Guest Editor for different special issues organized in IEEE Transaction on Industrial Informatics, IEEE Network, Neural Computing and Applications (Springer), Computer Communications (Elsevier), and Transactions on Emerging Telecommunications (Wiley).

Rasmeet Singh Bali received the M.E. degree from the National Institute of Technical Teachers Training and Research, Chandigarh University, Punjab, India, in 2013, and the Ph.D. degree from the Thapar Institute of Engineering and Technology, Punjab, India, in 2018. He is currently a Professor with the Computer Science and Engineering Department, Chandigarh University. His research interests include vehicular ad-hoc networks, data dissemination, patttern clustering, public key cryptography, biomedical communication, cloud computing, information dissemination, and intelligent transportation systems.