Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2023.026309

ARTICLE

Blockchain-Based Data Acquisition with Privacy Protection in UAV Cluster Network

Lemei Da[1], Hai Liang[1,*], Yong Ding[1,2], Yujue Wang[1], Changsong Yang[1] and Huiyong Wang[3]

[1]Guangxi Key Laboratory of Cryptography and Information Security, School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, 541004, China

[2]Cyberspace Security Research Center, Pengcheng Laboratory, Shenzhen, 518055, China

[3]School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin, 541004, China

*Corresponding Author: Hai Liang. Email: lianghai@guet.edu.cn

Received: 30 August 2022 Accepted: 08 December 2022

ABSTRACT

The unmanned aerial vehicle (UAV) self-organizing network is composed of multiple UAVs with autonomous capabilities according to a certain structure and scale, which can quickly and accurately complete complex tasks such as path planning, situational awareness, and information transmission. Due to the openness of the network, the UAV cluster is more vulnerable to passive eavesdropping, active interference, and other attacks, which makes the system face serious security threats. This paper proposes a Blockchain-Based Data Acquisition (BDA) scheme with privacy protection to address the data privacy and identity authentication problems in the UAV-assisted data acquisition scenario. Each UAV cluster has an aggregate unmanned aerial vehicle (AGV) that can batch-verify the acquisition reports within its administrative domain. After successful verification, AGV adds its signcrypted ciphertext to the aggregation and uploads it to the blockchain for storage. There are two chains in the blockchain that store the public key information of registered entities and the aggregated reports, respectively. The security analysis shows that the BDA construction can protect the privacy and authenticity of acquisition data, and effectively resist a malicious key generation center and the public-key substitution attack. It also provides unforgeability to acquisition reports under the Elliptic Curve Discrete Logarithm Problem (ECDLP) assumption. The performance analysis demonstrates that compared with other schemes, the proposed BDA construction has lower computational complexity and is more suitable for the UAV cluster network with limited computing power and storage capacity.

KEYWORDS

Unmanned aerial vehicle cluster network; certificateless signcryption; certificateless signature; batch verification; source authentication; data privacy; blockchain

1 Introduction

With the development of wireless networks, artificial intelligence, and other cutting-edge technologies, UAVs have been widely used in military and civil fields [1], such as intelligence acquisition, battlefield investigation, disaster rescues, security patrols, and natural disaster monitoring. In practical applications, multiple UAVs usually cooperate to complete complex tasks such as path planning,

situation awareness, and information transmission, which can effectively solve the problems of poor survivability and insufficient mission capability of a single UAV, so as to realize an efficient and intelligent UAV cluster network [2].

The communication modes of the UAV cluster network include UAV-UAV and UAV-CS, where the information is transmitted through wireless channels [3]. Therefore, the data transmission process will face many security problems, such as eavesdropping, identity impersonation, and message replay attacks [4]. UAVs used in the civilian/military field usually carry private information. If user data is intercepted, tampered or forged in transmission, it may lead to confidential information leakage, property losses, and even casualties. Therefore, it is significant to study the secure transmission mechanism of data in UAV cluster networks. Most existing solutions mainly focus on identity authentication security. For example, Wang et al. [5] designed an ID-based encrypted aggregate authentication framework; the airborne intelligence and control platform (named AC2P) broadcasted the authentication request to the UAV clusters, and each UAV returned a response with a signature after verifying the identity of AC2P, realizing mutual authentication between the two parties. Unfortunately, the cluster head used batch verification to improve computing efficiency, but lacked verification of other UAV responses, leaving a back door for attackers. To address the security problem left by Wang et al. [5], Li et al. [6] added an aggregation verification by the cluster head, and the response would be forwarded only after confirming that it was valid.

Both two schemes [5,6] mainly solve the problem of identity authentication, without mentioning the data transmission. If data transmission is required, it will be performed after identity authentication. However, in a large-scale UAV network, identity authentication and data transmission are executed in two processes will increase the number of interactions and reduce communication efficiency. It poses security and efficiency challenges for resource-constrained UAV cluster networks. In addition, for a complete UAV cluster network, the ground control station usually undertakes the functions of receiving, storing, and processing the acquisition data. If the communication signal of the ground control station is weak, or the ground control station suffers a serious physical attack, it is difficult for the UAV cluster to establish a communication link with the control station, and the acquisition data cannot be transmitted in time. Therefore, in order to ensure the normal operation of the UAV cluster network, an external database is needed to store the acquisition data, such as blockchain, cloud server, etc.

1.1 Our Contributions

This article proposes a blockchain-based data acquisition scheme with privacy protection, which realizes the secure transmission of acquisition data in the UAV cluster network. The main contributions of this article are summarized as follows:

- **Privacy protection:** In the BDA scheme, the reconnaissance unmanned aerial vehicle collects important data and sends it to the aggregation unmanned aerial vehicle after being signcrypted, and then uploads it to the blockchain after being verified. In the transmission process, the acquisition data always remains in the form of ciphertext, and only the control station has the right to decrypt it.
- **Source authentication** [7]**:** RAVs signcrypts the acquisition data, both the aggregation unmanned aerial vehicle and the control station will verify the acquisition report to ensure the authenticity of the data source.

- **Lightweight:** The aggregate verification technology is used to improve the speed of report verification, and the pairless signcryption algorithm is used to improve computational efficiency. Therefore, the BDA scheme is suitable for resource-constrained UAV cluster networks.
- **Secure storage:** As the storage carrier of the acquisition data, blockchain has the characteristics of openness, transparency, and immutability, which can ensure the secure storage of the acquisition data.

Compared with the previous version [8], this paper adds blockchain as a repository for data acquisition and improves the system model, scheme construction, security analysis, and experimental analysis.

1.2 Related Works

UAV technology has developed rapidly in recent years [9]. It has been applied to various fields to replace human beings to complete difficult and complex tasks. Noguchi et al. [10] monitored the disaster area in real-time with the help of UAVs, and it was convenient for obtaining information and managing rescue operations. To improve rescue efficiency, Qu et al. [11] studied how to reduce deployment delays of UAVs in emergency situations, and they proposed a K-Means algorithm based on user bandwidth to filter UAVs. The deployment location of a UAV is determined by the transmitting power of the UAV, the channel gain per unit distance, the threshold of SNR, the ground user's noise power, and the UAV's coverage radius. In [12], Liang et al. acquired forest hyperspectral images by using UAVs and classified forest species. Huang et al. [13] studied the portable outdoor charging platform on the tower and designed the hardware structure and algorithm to realize the docking function of electric patrol UAVs in the independent detection of transmission lines.

With the increasing application of UAVs in various fields, security has attracted more attention from academia. Gao et al. [14] proposed a situational awareness method to improve the active defense capability of UAVs. Based on the subtle changes of UAV's state parameters in the process of electromagnetic interference, they realized abnormal behavior detection by the tracking comparison method and used fuzzy logic reasoning to realize the semantic analysis of link interference and intrusion. Omri et al. [15] studied communication security in the air eavesdropping channel. They deduced the expression of security interruption probability of a standard air communication network with a single eavesdropper, and evaluated the physical layer security of the air communication system based on the standard and beamforming when there are multiple eavesdroppers. Kim et al. [16] designed a security module to connect the UAVs and the mission computer to ensure the security of communication. The control signal and telemetry data of the UAV are encrypted by the module and sent to the control station, which can protect data privacy effectively. To protect the privacy and security of user data, Liu et al. [17] designed a homomorphic encryption framework to help UAV suppliers improve user trust and information transparency. Tian et al. [18] studied an authentication algorithm based on Mobile Edge Computing (MEC) to protect the privacy of UAV identity, location, and flight route. The UAV used the lightweight signature method proposed by Yao et al. [19] to register. After joining the Internet of Drones (IoD), it performed mutual authentication with the MEC device and realized fast authentication in U2U communication through the MEC device. Khan et al. [20] applied UAV to the intelligent transportation networks, and proposed a privacy-protecting authentication scheme under the hyperelliptic curve cryptography technology, which can achieve the same level of security as 160 bits under the Elliptic Curve Cryptosystem (ECC) with only 80 bits key.

Due to the limitation of computing power, the algorithm complexity must be considered in UAV scenarios. Li et al. [21] proposed a lightweight security authentication mechanism in UAV networks. It guaranteed mutual authentication and verified the consistency of the session key. Based on the Physically Unclonable Functions, Alladi et al. [22] proposed a lightweight mutual authentication scheme for UAV-GS authentication, and further expanded it to support UAV-UAV authentication. It has security features such as mutual authentication and user anonymity, and can resist a variety of security attacks. In [5], Wang et al. proposed an ID-based aggregation authentication scheme, which added an aggregation unmanned aerial vehicle to aggregate data, and used batch authentication instead of data-by-data authentication to improve computing efficiency. However, Wang et al. [5] lacked the authentication of the aggregated unmanned aerial vehicle, Li et al. [6] improved it and added the verification of a single UAV response.

As an emerging distributed ledger, blockchain is often used in the field of the internet of things such as UAV cluster networks, intelligent transportation systems, smart grids, and wireless body area networks [23–26]. Ali et al. [27] adopted the certificateless public key signature (CLS) technology to provide conditional privacy protection for vehicle-to-infrastructure (V2I), and used the blockchain to store the valid pseudonym and the revoked pseudonym on PID-BC and RPID-BC, respectively, so as to realize the revocation transparency of pseudonym. Islam et al. [28] designed a blockchain-based secure health scheme, the UAV performed mutual authentication with the sensor to obtain a communication token, and then assisted the sensor to transmit the health data to the nearest server. The server stored the health data in the blockchain to realize secure sharing. Masuduzzaman et al. [29] proposed a scheme for real-time traffic management assisted by UAVs, blockchain was introduced to store traffic records to provide non-repudiation of data and avoid third-party interference with intelligent transportation systems. The proposals [30–32] used blockchain as a solution for UAV cluster network security.

1.3 Paper Organization

The remainder of this paper is organized as follows. Section 2 introduces ECDLP, consortium blockchain technology, and smart contract. Section 3 introduces the system model, security requirements, and system framework of BDA. Section 4 presents a basic construction in the elliptic curve group, and Section 5 improves basic construction and proposes a BDA construction, which security and performance are analyzed in Section 6. Section 7 concludes the paper.

2 Preliminary

ECDLP: Let p be a prime number and F_p be a finite field. An elliptic curve E defined over field F_p is defined as follows:

$$E(F_p) : y^2 = x^3 + ax + b \tag{1}$$

where $a, b \in F_p$. The points (x, y) on the curve $E(F_p)$ and a special point O_∞ called infinity point constitute a cyclic group $\mathbb{G}$. Let P be a generator of $\mathbb{G}$. ECDLP is a problem that given a point $Q \in \mathbb{G}$, calculates k such that $Q = kP$. The ECDLP assumption is that the advantage of solving the ECDLP problem for any algorithm $\mathcal{A}$ is negligible in any polynomial time algorithm, i.e., $\Pr[\mathcal{A}(P, Q) = k] \leq \varepsilon$, where ε denotes a negligible function in security parameter λ.

Consortium Blockchain: Consortium blockchain is a semi-open distributed system with a chain data structure. The generation of each block is jointly determined by pre-selected nodes, and other nodes can only trade through access control permissions [33]. Each transaction block combines data

blocks and information blocks in chronological order. As shown in Fig. 1, a block contains two parts, namely, block header and block body. The block header stores the hash value of the previous block, the hash value of the current block, and the timestamp to allow nodes to maintain the order of transactions. The block body stores transactions in a Merkle tree structure, and the Merkle root hash will be stored in the block header. Each block on the chain can record and store all transactions, the uploaded information can be automatically shared and distributed among nodes, and participants with access rights can query the records [34,35]. Transactions stored in the blockchain are always open and transparent.

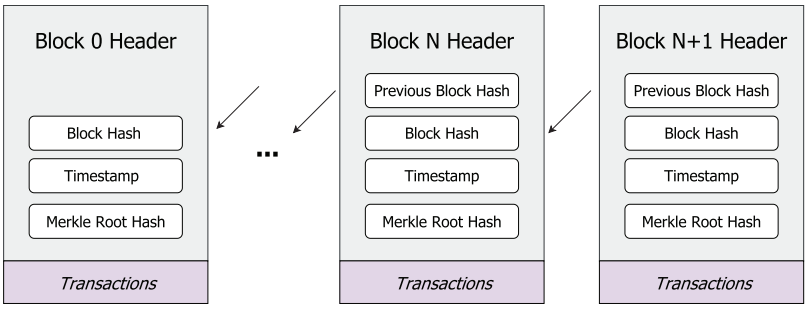

Figure 1: The structure of blockchain

Smart Contract: Smart contract is a digital contract deployed on the blockchain, which consists of many declarative statements with logical links, including execution conditions and execution logic [36]. When the condition is triggered, the corresponding logical statement will be executed automatically, and the relevant status and content will be updated. All transactions and updated states during execution are stored in the blockchain.

3 System Model and Security Requirements

3.1 System Model

As shown in Fig. 2, a BDA system contains five types of entities, namely, key generation center (KGC), reconnaissance unmanned aerial vehicles (RAVs), aggregation unmanned aerial vehicle (AGV), blockchain (BC) and control station (CS).

- **KGC:** Key generation center is mainly responsible for executing the initialization algorithm, generating system parameters, and distributing partial private keys *ppk* for each entity.
- **RAVs:** Reconnaissance unmanned aerial vehicle is a data acquisition device in the UAV cluster network, which has short-range communication capabilities and limited computing capabilities. There are multiple RAVs in each UAV cluster.
- **AGV:** Aggregate unmanned aerial vehicle is a data processing device in a cluster, with moderate computing and communication capabilities. There is only one aggregate unmanned aerial vehicle in each cluster, acting as the manager of the cluster.
- **BC:** Blockchain is a distributed ledger responsible for storing data. There are two chains in BDA, the public key management chain (PMC) is responsible for storing the public key information of all entities, and the data management chain (DMC) is responsible for storing the acquisition data of each cluster.
- **CS:** The control station has strong computing and communication capabilities and acts as a data processing and analysis organization. There is only one control station in a BDA system.

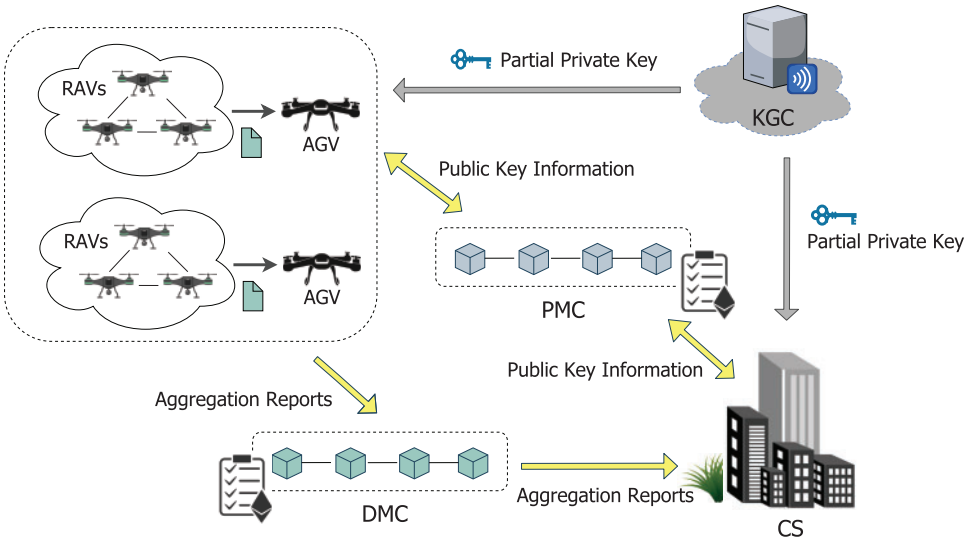

Figure 2: System model for data acquisition in UAV cluster network

In BDA system, when a new UAV cluster joins, it must first register with KGC. KGC issues an exclusive partial private key for each UAV, and UAV combines the partial private key and the secret value to generate a unique private key as the identification. Similarly, when CS is newly added, it also needs to register with KGC to generate its private key. Note any legal entity can access and upload its public key information to the public key management chain. And all nodes on the chain can obtain a table pk_{list} storing public key information after the consensus algorithm.

Each UAV cluster is a self-organizing network composed of multiple RAVs and one AGV. All RAVs in the cluster generate the acquisition reports and send them to the management AGV. AGV performs batch verification and adds its own acquisition report to generate an aggregation report when the verification is successful. In each cluster, only the AGV has the authorization to access the data management chain. Therefore, AGV will upload the aggregation report to DMC as a representative of its cluster, and DMC will create a table $data_{list}$ to record it. When CS needs the acquisition data of a UAV cluster, it can be obtained only by querying the data from DMC.

3.2 System Requirements

The data acquisition system in UAV cluster network must satisfy seven requirements, which are as follows:

- *Data confidentiality*: During data acquisition, any attacker cannot obtain the acquisition data through any channel. Even if an attacker monitors or intercepts the data, it would be impossible to decrypt it.
- *Data integrity*: Any external adversary cannot tamper with or forge a valid acquisition report of UAVs without being detected by CS.
- *Data authenticity*: The real source of acquisition data can be validated by both AGV and CS. That is, any external adversary cannot impersonate a legal entity to participate in data transmission.
- *Resistance of replay attack*: Any attacker intercepts and resends an expired message, AGV and CS can detect and reject the message.

- *Resistance of malicious KGC*: KGC only participates in part of the key generation process, it cannot obtain the real private key of the entity in communication. And the authenticity of partial private keys generated by KGC should be verified by the corresponding entities.
- *Resistance public-key substitution attack*: No adversary can use a fake public key to replace the real public key of the legitimate entity to participate in the communication.
- *Lightweight*: With limited storage capacity and computing resources, UAVs cannot support resource-intensive computations. Therefore, the algorithm must have high efficiency and low computational complexity.

3.3 System Framework

A BDA system for data acquisition in a UAV cluster network consists of the following six efficient procedures.

- **Setup:** On input a security parameter l, the setup algorithm, which is performed by the key generation center KGC, generates the system parameter *params* and the master private key s.
- **UAVReg:** The UAV registration algorithm is jointly performed by KGC and UAV. On input the system parameter *params*, the UAV's identity ID_i, and the master private key s, KGC outputs the partial private key ppk_i. And then with the system parameter *params* and the partial private key ppk_i, UAV ID_i outputs a public-private key pair (pk_i, sk_i) and the hash value $h_{k,i}$. The public key information $(ID_i, pk_i, h_{k,i})$ is uploaded to the PMC.
- **CSReg:** The CS registration algorithm is jointly performed by KGC and CS. On input the system parameter *params*, the CS's identity ID_c, and the master private key s, KGC outputs the partial private key ppk_c. And then with the system parameter *params* and the partial private key ppk_c, CS ID_c outputs a public-private key pair (pk_c, sk_c) and the hash value $h_{k,c}$. The public key information $(ID_c, pk_c, h_{k,c})$ is uploaded to the PMC.
- **DataAcq:** On input the system parameter *params*, the acquisition data m, the UAV's identity ID_i, and the private key sk of ID_i, the data acquisition algorithm, which is performed by each UAV, outputs the acquisition report σ on m.
- **DataAgg:** On input the system parameter *params* and each UAV's identity ID_i and its public key pk_i, the data aggregation algorithm, which is performed by AGV, outputs the aggregation report Ω and uploads it to data management chain DMC if all acquisition reports are validated, Otherwise, AGV verifies the acquisition reports one by one.
- **CSPro:** On input the system parameter *params*, the aggregation report Ω, and the private key sk_c, the CS processing algorithm, which is performed by CS, outputs the acquisition data $\{m\}$ of UAVs.

A correct BDA construction must satisfy the following conditions:

1) the partial private key generated by KGC can be successfully validated by the corresponding entity, i.e., RAV or AGV;
2) the ciphertext can be correctly decrypted by CS;
3) the acquisition reports of RAVs can be successfully validated by AGV;
4) the acquisition reports of UAVs (including RAVs and AGV) can be successfully validated by DMC.

4 Basic Construction

The frequently used symbols are shown in Table 1.

4.1 System Setup

On input a security parameter $l \in Z^+$, KGC chooses an elliptic curve additive group G with prime order q, where P is a generator of group G. Then KGC randomly chooses $s \in Z_q^*$ as the master private key and calculates

$$P_{pub} = sP \tag{2}$$

KGC also picks four collision-resistant hash functions H:$\{0,1\}^* \rightarrow \{0,1\}^{2\log q}$, $H_i : \{0,1\}^* \rightarrow Z_q^*$, where $i = 1, 2, 3$. Finally, KGC publishes the system parameters $params = (q, G, P, P_{pub}, H, H_1, H_2, H_3)$ and keeps the master private key s secret.

4.2 UAV Registration

Each UAV (RAVs and AGV) must be registered with the KGC before joining. For UAV ID_i (where $1 \leq i \leq n$), KGC picks a random value $r_i \in Z_q^*$ and calculates

$$R_i = r_i P \tag{3}$$

$$d_i = (r_i + sh_{1,i}) \bmod q \tag{4}$$

where $h_{1,i} = H_1(ID_i, R_i, P_{pub})$. Then, the partial private key $ppk_i = (d_i, R_i)$ is sent to ID_i through a secure channel.

Table 1: Notations

Notation	Description
G	An elliptic curve additive group
P	A generator of group G
q	Prime order of group G
s	Master private key
$params$	System parameters
$H_1, H_2, \ldots, H_4, H$	Collision-resistant hash functions
P_{pub}	Master public key
ID_c	Identity of CS
ID_n	Identity of AGV
ID_i	Identity of UAV
ppk	Partial private key
pk_c, sk_c	Public-private key pair of CS
pk_n, sk_n	Public-private key pair of AGV
pk_i, sk_i	Public-private key pair of UAV
m_i	Acquisition data by ID_i
m_n	Regional location of the AGV cluster
T_i	Timestamp
c_i	Ciphertext of acquisition data

(Continued)

Table 1 (continued)

Notation	Description
σ	Acquisition report
Ω	Aggregation report
V_{list}	A set of legal identities
A_{list}	A set of legal AGV identities
PMC	Public key management chain
DMC	Data management chain
pk_{list}	A list of public keys
$data_{list}$	A list of aggregated data

After receiving the partial private key ppk_i, UAV ID_i first verifies it by checking the following equality:

$$d_i P \stackrel{?}{=} R_i + h_{1,i} P_{pub} \tag{5}$$

If it holds, ID_i randomly chooses a secret value $x_i \in Z_q^*$ and calculates

$$X_i = x_i P \tag{6}$$

$$Q_i = R_i + h_{2,i} X_i \tag{7}$$

where $h_{2,i} = H_2(ID_i, X_i)$. Finally, UAV ID_i gets the public-private key pair (pk_i, sk_i), where $pk_i = (Q_i, R_i)$ and $sk_i = (d_i, x_i)$.

4.3 CS Registration

The control station also must be registered with the KGC. For CS ID_c, KGC picks a random value $r_c \in Z_q^*$ and calculates

$$R_c = r_c P \tag{8}$$

$$d_c = (r_c + sh_{1,c}) \bmod q \tag{9}$$

where $h_{1,c} = H_1(ID_c, R_c, P_{pub})$. Then, the partial private key $ppk_c = (d_c, R_c)$ is sent to ID_c through a secure channel.

After receiving the partial private key ppk_c, CS ID_c first verifies it by checking the following equality:

$$d_c P \stackrel{?}{=} R_c + h_{1,c} P_{pub} \tag{10}$$

If it holds, ID_c randomly chooses a secret value $x_c \in Z_q^*$ and calculates

$$X_c = x_c P \tag{11}$$

$$Q_c = R_c + h_{2,c} X_c \tag{12}$$

where $h_{2,c} = H_2(ID_c, X_c)$. Finally, CS ID_c gets the public-private key pair (pk_c, sk_c), where $pk_c = (Q_c, R_c)$ and $sk_c = (d_c, x_c)$.

4.4 Data Acquisition

In a cluster, each RAV captures environmental information or critical data and generates acquisition reports. For a message $m_i \in \{0,1\}^*$, RAV ID_i (where $1 \le i \le n-1$) generates a timestamp T_i, and then chooses a random number $\lambda_i \in Z_q^*$ and calculates the acquisition data ciphertext $c_i = (c_{1,i}, c_{2,i})$ as follows:

$$c_{1,i} = (m_i \| T_i \oplus H(\lambda_i(Q_c + h_{1,c}P_{pub})) \tag{13}$$

$$c_{2,i} = \lambda_i P \tag{14}$$

where $h_{1,c} = H_1(ID_c, R_c, P_{pub})$. To ensure the authenticity of the acquisition report, RAV ID_i picks a random number $u_i \in Z_q^*$ and calculates

$$U_i = u_i P \tag{15}$$

$$v_i = u_i + h_{3,i}(d_i + h_{2,i}x_i) \bmod q \tag{16}$$

where $h_{3,i} = H_3(ID_i, c_i)$. Finally, ID_i sends the acquisition report $\sigma_i = (c_i, U_i, v_i)$ to AGV ID_n.

4.5 Data Aggregation

For the received $n-1$ acquisition reports $\{\sigma_1, \ldots, \sigma_{n-1}\}$ from RAVs in the same cluster, AGV ID_n calculates

$$U = \sum_{i=1}^{n-1} U_i \tag{17}$$

$$v = \sum_{i=1}^{n-1} v_i \mod q \tag{18}$$

Then, ID_n verifies the authenticity of the acquisition reports by checking the following equation:

$$vP \stackrel{?}{=} U + \sum_{i=1}^{n-1} h_{3,i}Q_i + \left(\sum_{i=1}^{n-1} h_{3,i}h_{1,i}\right)P_{pub} \tag{19}$$

If holds, the acquisition reports from RAVs are valid. Otherwise, AGV verifies the acquisition reports one by one and filters out the invalid reports. Next, AGV ID_n generates a timestamp T_n, randomly picks $\lambda_n \in Z_q^*$, and generates the ciphertext $c_n = (c_{1,n}, c_{2,n})$ for the regional location m_n of the cluster as follows:

$$c_{1,n} = (m_n || T_n) \oplus H(\lambda_n(Q_c + h_{1,c}P_{pub})) \tag{20}$$

$$c_{2,n} = \lambda_n P \tag{21}$$

Then ID_n continues to choose a random value $u_n \in Z_q^*$ and calculate

$$U_n = u_n P \tag{22}$$

$$v_n = u_n + h_{3,n}(d_n + h_{2,n}x_n) \bmod q \tag{23}$$

The pair (U_n, v_n) is further added to the aggregated data as follows:

$$\hat{U} = U + U_n \tag{24}$$

$$\hat{v} = v + v_n \mod q \tag{25}$$

Finally, AGV ID_n outputs the aggregation report $\Omega = \left(\hat{U}, \hat{v}, c_1, \ldots, c_n\right)$ and sends it to CS.

4.6 CS Processing

When receiving the aggregation report $\Omega = \left(\hat{U}, \hat{v}, c_1, \ldots, c_n\right)$, CS first verifies its source as follows:

$$\hat{v}P \stackrel{?}{=} \hat{U} + \sum_{i=1}^{n} h_{3,i} Q_i + \left(\sum_{i=1}^{n} h_{3,i} h_{1,i}\right) P_{pub} \tag{26}$$

If holds, the acquisition data of the cluster are valid. Then, CS decrypts the n ciphertexts $(c_1, c_2, \ldots, c_n)$ one by one to obtain messages $(m_1, m_2, \ldots, m_n)$ acquired by UAVs. Namely, for each $c_i = (c_{1,i}, c_{2,i})$ where $i = 1, \ldots, n$, CS calculates

$$m_i \| T_i = c_{1,i} \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \tag{27}$$

where $h_{2,c} = H_2(ID_c, X_c)$.

4.7 Correctness

Theorem 4.1. The proposed basic construction is correct.

Proof. For the acquisition data ciphertext c_i, Eq. (27) satisfies as follows:

$$\begin{aligned} m_i \| T_i &= c_{1,i} \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \\ &= (m_i \| T_i) \oplus H(\lambda_i (Q_c + h_{1,c} P_{pub})) \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \\ &= (m_i \| T_i) \oplus H(\lambda_i (R_c + h_{2,c} X_c + h_{1,c} P_{pub})) \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \\ &= (m_i \| T_i) \oplus H(\lambda_i (r_c + h_{2,c} x_c + sh_{1,c}) P) \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \\ &= (m_i \| T_i) \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \oplus H((d_c + h_{2,c} x_c) c_{2,i}) \\ &= m_i \| T_i \end{aligned} \tag{28}$$

For the partial private key ppk_i issued by KGC, Eq. (5) satisfies as follows:

$$\begin{aligned} d_i P &= (r_i + sh_{1,i}) P \\ &= r_i P + sh_{1,i} P \\ &= R_i + h_{1,i} P_{pub} \end{aligned} \tag{29}$$

The correctness of Eq. (10) can be proved in a similar way to Eq. (5).

For the acquisition reports $\{\sigma_1, \ldots, \sigma_{n-1}\}$ from RAVs in the same cluster, Eq. (19) satisfies as follows:

$$\begin{aligned} vP &= \sum_{i=1}^{n-1} v_i P \\ &= \sum_{i=1}^{n-1} \left(u_i + h_{3,i}\left(d_i + h_{2,i}x_i\right)\right) P \\ &= \sum_{i=1}^{n-1} \left(u_i + h_{3,i}\left(\left(r_i + sh_{1,i}\right) + h_{2,i}x_i\right)\right) P \\ &= \sum_{i=1}^{n-1} u_i P + \sum_{i=1}^{n-1} h_{3,i}\left(\left(R_i + h_{1,i}P_{pub}\right) + h_{2,i}X_i\right) \\ &= \sum_{i=1}^{n-1} U_i + \sum_{i=1}^{n-1} h_{3,i}\left(Q_i + h_{1,i}P_{pub}\right) \\ &= U + \sum_{i=1}^{n-1} h_{3,i}Q_i + \left(\sum_{i=1}^{n-1} h_{3,i}h_{1,i}\right) P_{pub} \end{aligned} \tag{30}$$

For the aggregation report $\Omega = \left(\hat{U}, \hat{v}, c_1, \ldots, c_n\right)$ from AGV in the same cluster, Eq. (26) satisfies as follows:

$$\begin{aligned} \hat{v}P &= \sum_{i=1}^{n} v_i P \\ &= \sum_{i=1}^{n} \left(u_i + h_{3,i}\left(d_i + h_{2,i}x_i\right)\right) P \\ &= \sum_{i=1}^{n} \left(u_i + h_{3,i}\left(\left(r_i + sh_{1,i}\right) + h_{2,i}x_i\right)\right) P \\ &= \sum_{i=1}^{n} u_i P + \sum_{i=1}^{n} h_{3,i}\left(\left(R_i + h_{1,i}P_{pub}\right) + h_{2,i}X_i\right) \\ &= \sum_{i=1}^{n} U_i + \sum_{i=1}^{n} h_{3,i}\left(Q_i + h_{1,i}P_{pub}\right) \\ &= \hat{U} + \sum_{i=1}^{n} h_{3,i}Q_i + \left(\sum_{i=1}^{n} h_{3,i}h_{1,i}\right) P_{pub} \end{aligned} \tag{31}$$

5 BDA Construction

In practical applications, the control station cannot guarantee that it is always in a normal communication state. Therefore, we extend the basic construction to store the data on the blockchain for special circumstances (i.e., communication interruption). The implementation processes of UAV registration and data acquisition are shown in Figs. 3 and 4, respectively.

5.1 System Setup

The Setup algorithm in Section 4 has generated system parameters *params*. In BDA, in addition to the above parameters, KGC also needs to select another one-way hash function H_4:$\{0,1\}^* \rightarrow Z_q^*$, and then create a set of legal identities V_{list} and a set of legal AGV identities A_{list}. Finally, KGC publics the system parameters $params' = (q, G, P, P_{pub}, H, H_1, H_2, H_3, H_4, V_{list}, A_{list})$.

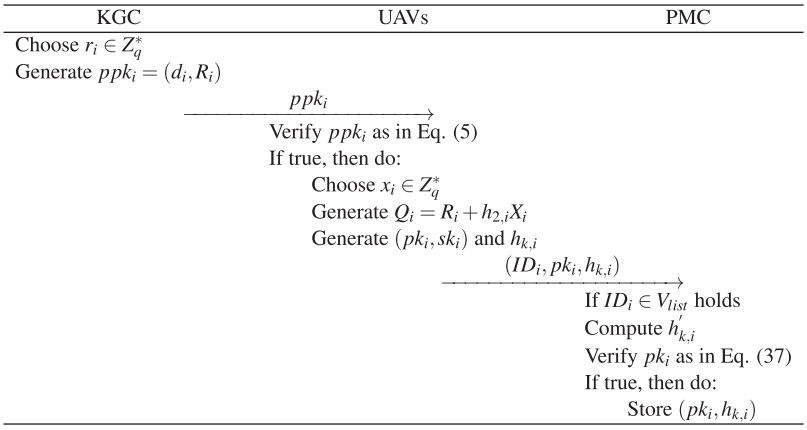

Figure 3: A procedure of UAV registration

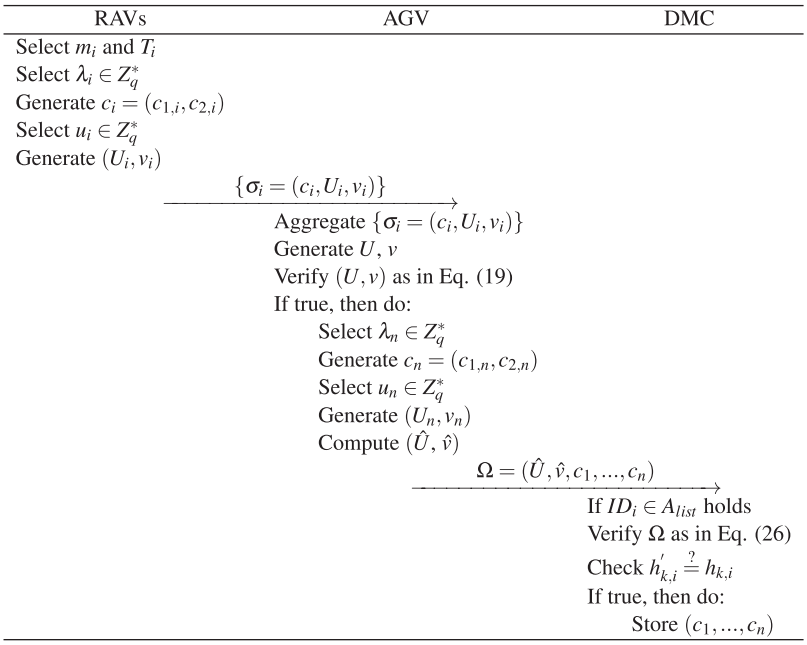

Figure 4: A procedure of data acquisition in UAV cluster

5.2 UAV Registration

In BDA, all entities first execute the UAV registration algorithm in Section 4 to generate a public-private key pair, and then upload the public key information to PMC. For UAV ID_i, it calculates the hash value of pk_i as follows:

$$h_{k,i} = H_4(pk_i, ID_i) \tag{32}$$

then uploads $(ID_i, pk_i, h_{k,i})$ to the blockchain, PMC automatically executes **STORAGE_PK**.

STORAGE_PK is a public key storage contract deployed on PMC. When a UAV uploads the public key information $(ID_i, pk_i, h_{k,i})$, Algorithm 1 first verifies $ID_i \in V_{list}$, if true, it calculates

$$h'_{k,i} = H_4(pk_i, ID_i) \tag{33}$$

and verifies the following equation:

$$h'_{k,i} \stackrel{?}{=} h_{k,i} \tag{34}$$

If true, the tuple $(pk_i, h_{k,i})$ will be stored in PMC. Note that pk_i and $h_{k,i}$ are public data.

5.3 CS Registration

For CS ID_c, it calculates the hash value of pk_c

$$h_{k,c} = H_4(pk_c, ID_c) \tag{35}$$

then uploads $(ID_c, pk_c, h_{k,c})$ to the blockchain, PMC automatically executes **STORAGE_PK**.

STORAGE_PK is a public key storage contract deployed on PMC. When CS uploads the public key information $(ID_c, pk_c, h_{k,c})$, Algorithm 1 first verifies $ID_c \in V_{list}$, if true, it calculates

$$h'_{k,c} = H_4(pk_c, ID_c) \tag{36}$$

and verifies the following equation:

$$h'_{k,c} \stackrel{?}{=} h_{k,c} \tag{37}$$

If true, the tuple $(pk_c, h_{k,c})$ will be stored in PMC. Note that pk_c and $h_{k,c}$ also are public data.

Algorithm 1 STORAGE_PK

Input: ID_*, pk_*, $h_{k,*}$, $params$

Output: {0: *Unsuccessful*, 1: *Successful*}

```
RAV ID_* ← (pk_*, h_k,*)
if ID_i ∈ V_list then
    h'_k,* = H_4(pk_*, ID_*)
    if h'_k,* = h_k,* then
        pk_list.write(pk_*, h_k,*)
        //pk_list is a table of public keys stored on the PMC; write() is the function that inserts the data into the block.
        return 1
    else
        return 0
    end if
end if
```

5.4 Data Acquisition

This algorithm is the same as the data acquisition algorithm in Section 4, so it is omitted here.

5.5 Data Aggregation

AGV executes the data aggregation algorithm in Section 4 to verify all RAVs acquisition reports in a cluster, and generates an aggregation report, then uploads it to DMC. For the aggregation report Ω of each cluster, DMC automatically executes **STORAGE_DATA**.

STORAGE_DATA is a data storage contract deployed on DMC. DMC received the upload request from AGV, Algorithm 2 first verifies $ID_n \in A_{list}$, if true, it then verifies the source of the aggregated report Ω as follows:

$$\hat{v}P \stackrel{?}{=} \hat{U} + \sum_{i=1}^{n} h_{3,i}Q_i + \left(\sum_{i=1}^{n} h_{3,i}h_{1,i}\right) P_{pub} \tag{38}$$

If true, it means the aggregation report is valid and the tuple $(c_1, \ldots, c_n)$ will be stored in DMC. Note that $(c_1, \ldots, c_n)$ are encrypted data, and other users on the chain cannot know their real content.

Algorithm 2 STORAGE_DATA

Input: $\Omega, params', pk_i$
Output: {0: Unsuccessful; 1: Successful}

```
AGV ID_n ← Ω
if ID_n ∈ A_list then
    for each i ∈ [1, n] do
        h_{1,i} ← H_1(ID_i, R_i, P_pub)
        h_{3,i} ← H_3(ID_i, c_i)
    end for
    Σ = Û + Σ_{i=1}^{n} h_{3,i}Q_i + (Σ_{i=1}^{n} h_{3,i}h_{1,i}) P_pub
    if v̂P = Σ then
        data_list.write(c_1, ..., c_n)
          //data_list is a table of the aggregated data stored on the DMC.
        return 1
    else
        return 0
    end if
end if
```

5.6 CS Processing

CS queries the acquisition data of the specified cluster through the *ID* of AGV, CS decrypts n ciphertexts $c_1, c_2, \ldots, c_n$ one by one to obtain messages $m_1, m_2, \ldots, m_n$ acquired by all UAVs. That is, for each ciphertext $c_i = (c_{1,i}, c_{2,i})$ where $i = 1, \ldots, n$, CS decrypts it by the following equality:

$$m_i \| T_i = c_{1,i} \oplus H((d_c + h_{2,c}x_c)c_{2,i}) \tag{39}$$

where $h_{2,c} = H_2(ID_c, X_c)$.

5.7 Correctness

The correctness proof of the equations involved in the BDA construction has been completed in Section 4. Note that the correctness proof of Eq. (38) is the same as that of Eq. (26), so it is omitted here. In Eq. (37), since $h_{k,i}$ and $h'_{k,i}$ have the same input parameters, the two values must be equal.

6 Analysis

6.1 Security Analysis

Theorem 6.1. The proposed BDA construction can guarantee the confidentiality of the acquisition data. That is, any adversary cannot decrypt the ciphertext to obtain the real content of the acquisition data.

Proof. In the proposed BDA construction, UAV uses the CS's public key pk_c to signcrypt the acquisition data. When an adversary eavesdrops or intercepts the acquisition reports, it must obtain the private key sk_c of the CS to decrypt them, that is, calculate $m_i \| T_i = c_{1,i} \oplus H((d_c + h_{2,c} x_c) c_{2,i})$. However, the adversary would be unable to generate the private key sk_c without the partial private key ppk_c and the secret value x_c, and it also cannot obtain a valid private key sk_c from other channels. Thus, the proposed BDA construction can guarantee the confidentiality of the acquisition data from UAVs.

Theorem 6.2. The proposed BDA construction can guarantee the integrity of the acquisition data. That is, any adversary cannot tamper with the content of the acquisition reports.

Proof. In the proposed BDA construction, the certificateless signcryption technology is used to process the acquisition data, which is adapted from the PF-CLS scheme of Thumbur et al. [37]. The process of generating (U, v) in BDA construction is the same as that of generating σ in PF-CLS. According to Theorem 1 in [37], the PF-CLS is proved to be existentially unforgeable under the ECDLP assumption. Thus, the proposed BDA construction also enjoys existentially unforgeability and guarantees the integrity of the acquisition data.

Theorem 6.3. The proposed BDA construction can guarantee the authenticity of the acquisition data source.

Proof. As shown in Theorem 6.2, the proposed BDA construction has been proved to be unforgeable under the ECDLP assumption. Therefore, any adversary cannot impersonate a legal UAV to produce a valid acquisition report without being detected, which means that the authenticity of data source can be guaranteed.

Theorem 6.4. The proposed BDA construction can resist replay attacks.

Proof. When generating an acquisition report, a timestamp T will be introduced. If the adversary replays an expired message, CS can detect it by checking the freshness of each message. Also, according to Theorem 6.2 and Theorem 6.3, the acquisition reports have been proven to be unforgeable, which means any adversary cannot change the timestamp in the acquisition report. Thus, the proposed BDA construction can resist replay attacks.

Theorem 6.5. The proposed BDA construction can resist the malicious KGC. That is, KGC cannot obtain the private key of any entity or forge a valid acquisition report.

Proof. In the proposed BDA construction, The private key is jointly generated by the KGC and the entity. KGC only generates the partial private key for the entity, which means that KGC cannot know the private key. According to Theorem 6.2 and Theorem 6.3, without the UAV's private key, KGC is unable to forge a valid acquisition report. Hence, the proposed BDA construction can resist malicious KGC.

Theorem 6.6. The proposed BDA construction can resist the public-key substitution attacks. That is, the public key uploaded to the PMC will not be replaced with a fake public key by a malicious attacker.

Proof. The proposed BDA construction uploads the public key information $(ID, pk, h_{k,*})$ of RAVs, AGV, and CS. In the proposed BDA construction, all entities (RAVs, AGV, and CS) must upload their public key information to PMC after registration. The public key storage contract deployed on PMC compares $h_{k,*}$ with $h'_{k,*}$ can detect whether the public key information has been changed before uploading. Moreover, according to the tamper-proof features of blockchain, malicious adversaries cannot replace the uploaded public keys. That is, the proposed BDA construction can resist the public-key substitution attacks.

6.2 Theoretical Analysis

This section compares the proposed BDA construction with IBE-AggAuth [5], AAS [6], and CLAS [38]. As shown in Table 2, the IBE-AggAuth scheme lacks the aggregation authentication of AGVs, which makes it difficult to ensure the identity authenticity of RAVs. In addition, it lacks the privacy protection of acquisition data. Similarly, the AAS scheme and the CLAS scheme do not clarify the privacy protection of data, but they improve the aggregation authentication of AGVs and realize the identity authentication of RAVs. The proposed BDA construction solves the problem of data privacy protection while realizing the aggregation authentication of AGV (or CS).

Table 2: Comparision with related technologies

Scheme	Aggregate verification of AGV	Aggregate verification of CS	Privacy protection of acquisition data
IBE-AggAuth [5]	–	√	–
AAS [6]	√	√	–
CLAS [38]	√	√	–
BDA	√	√	√

We summarize the computational complexity of the algorithms of the IBE-AggAuth scheme [5], the AAS scheme [6], the CLAS scheme [38], and the BDA construction in Table 3. We only focus on the time-consuming operations, where T_{SM} is the scalar point multiplication in G, T_{EA} is the elliptic curve point addition in G, and T_{PA} is the bilinear pairing operation.

Table 3: Computational complexity of each algorithm in BDA

Scheme		IBE-AggAuth [5]	AAS [6]	CLAS [38]	BDA
Setup		–	–	T_{SM}	T_{SM}
UAVReg		T_{SM}	$2T_{SM}$	$5T_{SM} + 2T_{EA}$	$5T_{SM} + 2T_{EA}$
CSReg		T_{SM}	$2T_{SM}$	$5T_{SM} + 2T_{EA}$	$5T_{SM} + 2T_{EA}$
DataAcq	Authentication	$3T_{SM} + T_{EA}$	$3T_{SM} + 2T_{EA}$	T_{SM}	T_{SM}
	Total	–	–	–	$4T_{SM} + T_{EA}$
DataAgg	Authentication	$2(n-1)T_{EA}$	$(n+2)T_{SM} + 4(n-1)T_{EA} + 3T_{PA}$	$(n+2)T_{SM} + (2n-1)T_{EA}$	$(n+2)T_{SM} + (2n-1)T_{EA}$
	Total	–	–	–	$(n+5)T_{SM} + 2nT_{EA}$

(Continued)

Table 3 (continued)

Scheme		IBE-AggAuth [5]	AAS [6]	CLAS [38]	BDA
CSPro	Authentication	$nT_{SM} + 3T_{PA} + (n-1)T_{EA}$	$nT_{SM} + (2n-1)T_{EA} + 3T_{PA}$	$(n+2)T_{SM} + (n+1)T_{EA}$	$(n+2)T_{SM} + (n+1)T_{EA}$
	Total	–	–	–	$(n+3)T_{SM} + (n+1)T_{EA}$

The computational complexity of the **Setup** algorithm, **UAVReg** algorithm, **CSReg** algorithm, and **DataAcq** algorithm in Table 3 is one execution. The IBE-AggAuth scheme [5] does not perform time-consuming operations in the **Setup** algorithm, and both the **UAVReg** algorithm and the **CSReg** algorithm only need 1 scalar multiplication in G. Each UAV performs 3 scalar point multiplications in G and 1 elliptic curve point addition in G to generate a signature, and $2(n-1)$ elliptic curve point additions in G for n signature aggregation. In **CSPro**, CS verifies the aggregated data needs to perform n scalar point multiplications in G, $n-1$ elliptic curve point additions in G, and 3 bilinear pairing operations. Same as the IBE-AggAuth scheme [5], the AAS scheme [6] also does not perform time-consuming operations in the **Setup** algorithm, and it needs 2 scalar point multiplications in G to generate the public-private key pairs of UAV (or CS). In **DataAcq**, Each RAV generates a signature by performing 3 scalar point multiplications in G and 2 elliptic curve point additions in G. The AGV performs $n+2$ scalar point multiplications in G, $4(n-1)$ elliptic curve point additions in G, and 3 bilinear pairing operations to complete aggregation authentication and secondary aggregation in **DataAgg**. And CS performs n scalar point multiplications in G, $2n-1$ elliptic curve point additions in G, and 3 bilinear pairing operations to verify the aggregated data.

Since the CLAS scheme [38] and the BDA construction need to generate the master public key, they perform 1 scalar point multiplication in G, respectively. Both the CLAS scheme [38] and the BDA scheme perform 5 scalar point multiplications in G and 2 elliptic curve point additions in G to complete the UAV (or CS) registration. Because they use certificateless cryptography technology, the generation and verification of the partial private key increase the computational complexity. Since authentication and data privacy protection are implemented simultaneously in BDA scheme, Table 3 divides **DataAcq** into **Authentication** and **total**. In **Authentication**, both the CLAS scheme [38] and the BDA scheme require 1 scalar point multiplication in G to generate an authentication message. The AGV performs $n+2$ scalar multiplication operations in G and $2n-1$ elliptic curve point additions in G for aggregation verification and secondary aggregation. In addition, the proposed BDA scheme also achieves data privacy protection, which requires $n+5$ scalar point multiplications in G and $2n$ elliptic curve point additions in G in the complete **DataAgg** algorithm. In **CSPro**, both the CLAS scheme [38] and the BDA scheme need to perform $n+2$ scalar point multiplications in G and $n+1$ elliptic curve point additions in G. The difference is that the BDA scheme also needs to decrypt the signcrypted data. Therefore, a total of $n+3$ scalar point multiplications in G and $n+1$ elliptic curve point additions in G are required.

6.3 Experimental Analysis

In this section, we evaluate the experimental performance of the proposed BDA scheme and compare it with AAS construction [6] and CLAS construction [38], The Golang language is used to simulate the above scheme under the windows 10 platform intel(R) core(TM) i5-9500 @ 3.00 GHz. Since the AAS scheme is based on bilinear pairing, it is simulated in the Pairing-Based Cryptography (PBC) library, where the elliptic curve is of Type E ($y^2 = x^3+ax+b$) such that q and the element size in G are all 256 bits. Both the CLAS construction and the proposed scheme do not require bilinear pairing

operations, so they are simulated in the Elliptic Curve Digital Signature Algorithm library, where the elliptic curve is NISTP-256 ($y^2 = x^3 - 3x + b$), q and the element size in G are the same as in AAS scheme, both are 256 bits. Blockchain related experiments using Solidity programming languages, and the FISCO BCOS 2.0 is adopted as the underlying framework of the blockchain.

Fig. 5 shows the experimental performance of the proposed BDA scheme, AAS scheme [6], and CLAS scheme [38] in **Setup**, **UAVReg**, **CSReg**, and **DataAcq** (authentication only). The experimental results show that in all the above algorithms, the time-consuming of CLAS construction and BDA construction is significantly lower than AAS construction, and the time of CLAS construction and BDA construction is similar and with a small gap. In addition, regardless of the scheme, the time cost of the **Setup** algorithm is significantly higher than other algorithms. The system initialization process in AAS scheme [6] takes about 12.7 msec, while CLAS scheme [38] and the proposed scheme require about 0.6 msec. The UAV (or CS) generates its own public-private key pair to complete the registration process, which requires about 3.0 msec in AAS scheme [6], and about 0.2 msec for both CLAS scheme [38] and BDA scheme. In *DataAcq* (authentication only), the AAS scheme [6] takes about 4.0 msec for UAV to generate a signature, while the CLAS scheme [38] only takes about 0.1 msec. Fig. 5 only shows the time required for BDA to generate the authentication part of a signcryption, which is about 0.1 msec.

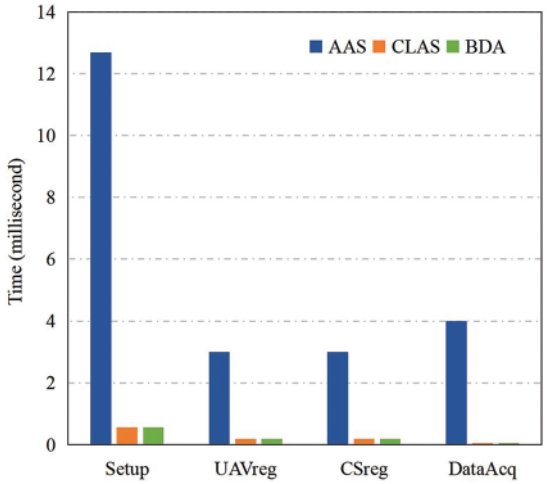

Figure 5: Performance comparison with AAS of [6] and CLAS of [38] in *Setup*, *UAVReg*, *CSReg* and *DataAcq* (authentication only)

To evaluate the experimental performance of generating multiple signatures or authentications. We test the time cost when generating 20, 40, 60, 80, 100 signatures or authentications, respectively, and the results are shown in Fig. 6. The authentication time of AAS scheme [6], CLAS scheme [38] and the proposed scheme all have a linear growth trend with the increase of the number of UAVs. And the time growth of AAS scheme [6] increases rapidly, while the growth of CLAS scheme [38] and BDA construction are relatively gentle.

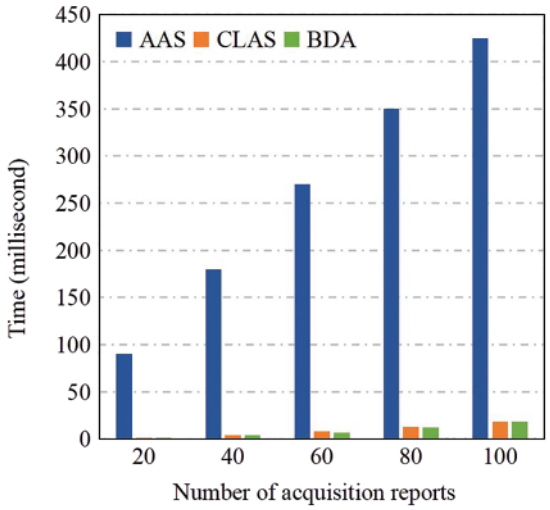

Figure 6: Performance comparison with AAS of [6] and CLAS of [38] in *DataAcq* (authentication only)

Similarly, in Fig. 7, we also test the aggregation time cost of 20, 40, 60, 80, 100 signatures, respectively. For example, in the AAS scheme [6], aggregating 20 signatures cost about 78 msec, the CLAS scheme [38] needs about 3.0 msec, and the BDA scheme only needs about 2.9 msec. Compared with the AAS scheme, the CLAS scheme and the BDA scheme save roughly 75 msec. In addition, the AAS scheme has a linear growth trend with the increase in the number of UAVs, and the growth trend is relatively rapid, while the growth trends of CLAS scheme and BDA scheme are relatively gentle.

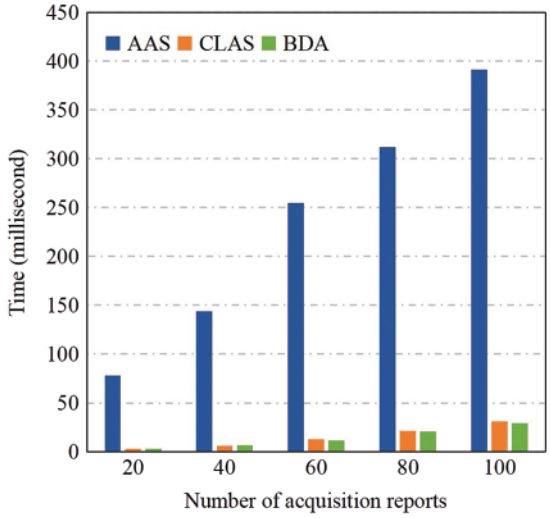

Figure 7: Performance comparison with AAS of [6] and CLAS of [38] in *DataAgg* (authentication only)

Fig. 8 shows that the performance of CS to perform aggregate verification of 20, 40, 60, 80, 100 signatures in AAS scheme [6], the CLAS scheme [38], and the proposed BDA scheme. Similar to Figs. 6 and 7, the time complexity of aggregate verification also increases linearly with the number of UAVs,

and the time cost of the AAS scheme is much higher than that of the CLAS scheme and the proposed BDA scheme. In Figs. 6–8, we only compare the authentication time of CLAS and BDA, and the time consumed is similar. However, the proposed BDA scheme not only realizes authentication, but also protects data privacy.

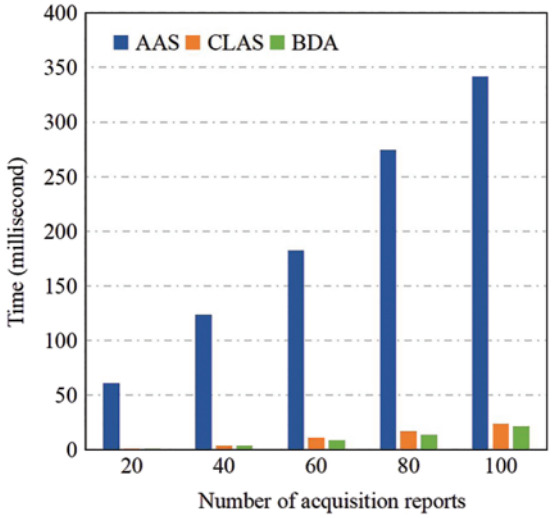

Figure 8: Performance comparison with AAS of [6] and CLAS of [38] in *CSPro* (authentication only)

In order to evaluate the performance of uploading the acquisition reports to the blockchain, we tested the time cost and gas cost of 20, 40, 60, 80, 100 reports, respectively. As shown in Fig. 9, it takes about 0.45 s to upload 20 reports, while uploading 100 reports takes about 0.62 s. That is the more reports are uploaded, the higher upload efficiency of a single report. The gas cost of uploading 20 reports is 6523810 gas, so the gas cost of a single report is about 326191 gas. Similarly, the gas cost of uploading 100 reports is 32451136 gas, so the gas cost of a single report is about 324511 gas. Therefore, the gas cost is related to the number of uploaded reports and increases linearly.

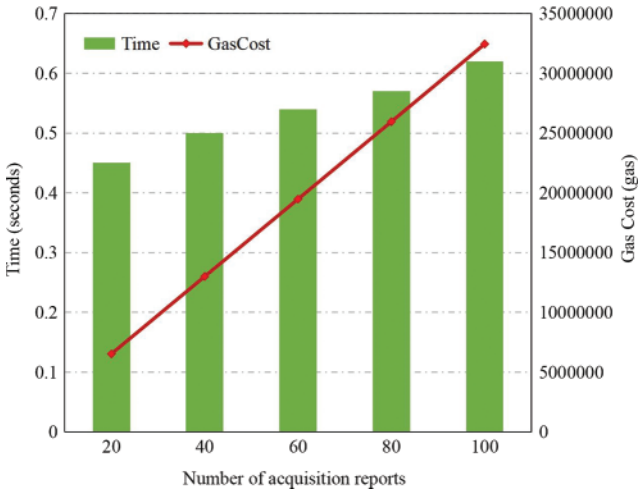

Figure 9: Performance and gas cost of uploading the acquisition reports

7 Conclusions

To address data privacy and identity authentication problems in the resource-constrained UAV cluster network, this paper proposed a Blockchain-based Data Acquisition (BDA) scheme with privacy protection. In a UAV cluster, RAVs signcrypted the acquisition data and sent the data to the administrative AGV for aggregation and verification. If no invalid report was found, the AGV would further add its own signcryption to the aggregated data and upload it to the blockchain. With BDA, all entities uploaded their public key information to the PMC of the blockchain to resist public-key substitution attacks. Due to the characteristics of the blockchain, the acquisition data stored in the DMC would not be tampered with and forged by malicious adversaries, and the CS only needed to decrypt it without verification operations. Security analysis showed that the proposed BDA construction could protect the privacy and authenticity of the acquisition data, and resist replay attacks, the public-key substitution attack, and malicious KGC. The theroetical and experimental analysis demonstrated that the proposed BDA construction is suitable for UAV cluster networks.

Funding Statement: This article is supported in part by the National Key R&D Program of China under Project 2020YFB1006004, the Guangxi Natural Science Foundation under Grants 2019GXNSFFA245015 and 2019GXNSFGA245004, the National Natural Science Foundation of China under Projects 62162017, 61862012, 61962012, and 62172119, the Major Key Project of PCL under Grants PCL2021A09, PCL2021A02 and PCL2022A03, and the Innovation Project of Guangxi Graduate Education YCSW2021175.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Rana, T., Shankar, A., Sultan, M. K., Patan, R., Balusamy, B. (2019). An intelligent approach for UAV and drone privacy security using blockchain methodology. *2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, Noida, India.
2. Arafat, M. Y., Moh, S. (2019). A survey on cluster-based routing protocols for unmanned aerial vehicle networks. *IEEE Access, 7,* 498–516. https://doi.org/10.1109/ACCESS.2018.2885539
3. He, D., Chan, S., Guizani, M. (2017). Communication security of unmanned aerial vehicles. *IEEE Wireless Communications, 24(4),* 134–139. https://doi.org/10.1109/MWC.2016.1600073WC
4. Rodrigues, M., Amaro, J., Osório, F. S., Kalinka, R. L. J. C. B. (2019). Authentication methods for UAV communication. *2019 IEEE Symposium on Computers and Communications (ISCC)*, Barcelona, Spain.
5. Wang, H., Li, J., Lai, C., Wang, Z. (2020). A provably secure aggregate authentication scheme for unmanned aerial vehicle cluster networks. *Peer-to-Peer Networking and Applications, 13(1),* 53–63. https://doi.org/10.1007/s12083-019-0718-9
6. Li, J., Zhao, M., Ding, Y., Dennis, Y. W. L., Wang, Y. et al. (2020). An aggregate authentication framework for unmanned aerial vehicle cluster network. *2020 International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, Exeter, UK.
7. Wang, Y., Ding, Y., Wu, Q., Wei, Y., Qin, B. et al. (2019). Privacy-preserving cloud-based road condition monitoring with source authentication in vanets. *IEEE Transactions on Information Forensics and Security, 14(7),* 1779–1790. https://doi.org/10.1109/TIFS.2018.2885277
8. Da, L., Wang, Y., Ding, Y., Xiong, W., Wang, H. et al. (2021). An efficient certificateless signcryption scheme for secure communication in UAV cluster network. *2021 IEEE International Conference on Parallel*

& Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), New York, USA.

9. Fu, Z., Mao, Y., He, D., Yu, J., Xie, G. (2019). Secure multi-UAV collaborative task allocation. *IEEE Access, 7,* 35579–35587. https://doi.org/10.1109/ACCESS.2019.2902221
10. Noguchi, T., Komiya, Y. (2019). Persistent cooperative monitoring system of disaster areas using UAV networks. *2019 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Leicester, UK.
11. Qu, H., Zhang, W., Zhao, J., Luan, Z., Chang, C. (2020). Rapid deployment of UAVs based on bandwidth resources in emergency scenarios. *2020 Information Communication Technologies Conference (ICTC)*, Nanjing, China.
12. Liang, J., Li, P., Zhao, H., Han, L., Qu, M. (2020). Forest species classification of UAV hyperspectral image using deep learning. *2020 Chinese Automation Congress (CAC)*, Shanghai, China.
13. Huang, Z., Zhang, T., Liu, P., Lu, X. (2020). Outdoor independent charging platform system for power patrol UAV. *2020 12th IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, Nanjing, China.
14. Gao, X., Jia, H., Chen, Z., Yuan, G., Yang, S. (2020). UAV security situation awareness method based on semantic analysis. *2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, Shenyang, China.
15. Omri, A., Hasna, M. O. (2018). Physical layer security analysis of UAV based communication networks. *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Chicago, USA.
16. Kim, K., Kang, Y. (2020). Drone security module for UAV data encryption. *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, Korea (South).
17. Liu, L., Qian, H., Hu, F. (2019). Random label based security authentication mechanism for large-scale UAV swarm. *2019 IEEE International Conference on Parallel Distributed Processing with Applications, Big Data Cloud Computing, Sustainable Computing Communications, Social Computing Networking (ISPA/BDCloud/SocialCom/SustainCom)*, Xiamen, China.
18. Tian, Y., Yuan, J., Song, H. (2019). Efficient privacy-preserving authentication framework for edge-assisted internet of drones. *Journal of Information Security and Applications, 48,* 102354. https://doi.org/10.1016/j.jisa.2019.06.010
19. Yao, A. C. C., Zhao, Y. (2013). Online/offline signatures for low-power devices. *IEEE Transactions on Information Forensics and Security, 8(2),* 283–294. https://doi.org/10.1109/TIFS.2012.2232653
20. Khan, M. A., Ullah, I., Alkhalifah, A., Rehman, S. U., Shah, J. A. et al. (2022). A provable and privacy-preserving authentication scheme for UAV-enabled intelligent transportation systems. *IEEE Transactions on Industrial Informatics, 18(5),* 3416–3425. https://doi.org/10.1109/TII.2021.3101651
21. Li, T., Ma, J., Feng, P., Meng, Y., Ma, X. et al. (2019). Lightweight security authentication mechanism towards UAV networks. *2019 International Conference on Networking and Network Applications (NaNA)*, Daegu, Korea (South).
22. Alladi, T., Naren, Bansal, G., Chamola, V., Guizani, M. (2020). SecAuthUAV: A novel authentication scheme for UAV-ground station and UAV-UAV communication. *IEEE Transactions on Vehicular Technology, 69(12),* 15068–15077. https://doi.org/10.1109/TVT.2020.3033060
23. Xiong, W., Wang, R., Wang, Y., Zhou, F., Luo, X. (2021). CPPA-D: Efficient conditional privacy-preserving authentication scheme with double-insurance in VANETs. *IEEE Transactions on Vehicular Technology, 70(4),* 3456–3468. https://doi.org/10.1109/TVT.2021.3064337
24. Ding, Y., Luo, D., Xiang, H., Liu, W., Wang, Y. (2021). Design and implementation of blockchain-based digital advertising media promotion system. *Peer-to-Peer Networking and Applications, 14(2),* 482–496. https://doi.org/10.1007/s12083-020-00984-5

25. Zhao, M., Ding, Y., Tang, S., Liang, H., Wang, H. (2022). A blockchain-based framework for privacy-preserving and verifiable billing in smart grid. *Peer-to-Peer Networking and Applications,* 1–14. https://doi.org/10.1007/s12083-022-01379-4
26. Zhang, T., Wang, Y., Ding, Y., Wu, Q., Liang, H. et al. (2022). Multi-party electronic contract signing protocol based on blockchain. *IEICE Transactions on Information and Systems, 105(2),* 264–271. https://doi.org/10.1587/transinf.2021BCP0011
27. Ali, I., Gervais, M., Ahene, E., Li, F. (2019). A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in vanets. *Journal of Systems Architecture, 99,* 101636. https://doi.org/10.1016/j.sysarc.2019.101636
28. Islam, A., Young Shin, S. (2020). A blockchain-based secure healthcare scheme with the assistance of unmanned aerial vehicle in internet of things. *Computers & Electrical Engineering, 84,* 106627. https://doi.org/10.1016/j.compeleceng.2020.106627
29. Masuduzzaman, M., Islam, A., Sadia, K., Shin, S. Y. (2022). UAV-based UAVs-assisted automated traffic management scheme using blockchain. *Future Generation Computer Systems, 134,* 256–270. https://doi.org/10.1016/j.future.2022.04.018
30. Ghribi, E., Khoei, T. T., Gorji, H. T., Ranganathan, P., Kaabouch, N. (2020). A secure blockchain-based communication approach for UAV networks. *2020 IEEE International Conference on Electro Information Technology (EIT)*, Chicago, USA.
31. Gai, K., Wu, Y., Zhu, L., Choo, K. K. R., Xiao, B. (2021). Blockchain-enabled trustworthy group communications in UAV networks. *IEEE Transactions on Intelligent Transportation Systems, 22(7),* 4118–4130. https://doi.org/10.1109/TITS.2020.3015862
32. Ossamah, A. (2020). Blockchain as a solution to drone cybersecurity. *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, New Orleans, USA.
33. Lu, Y. (2019). The blockchain: State-of-the-art and research challenges. *Journal of Industrial Information Integration, 15,* 80–90. https://doi.org/10.1016/j.jii.2019.04.002
34. Wen, B., Wang, Y., Ding, Y., Zheng, H., Liang, H. et al. (2021). A privacy-preserving blockchain supervision framework in the multiparty setting. *Wireless Communications and Mobile Computing, 2021,* 5236579. https://doi.org/10.1155/2021/5236579
35. Liang, W., Wang, Y., Ding, Y., Zheng, H., Liang, H. et al. (2022). An efficient anonymous authentication and supervision system based on blockchain. *2022 7th IEEE International Conference on Data Science in Cyberspace (DSC)*, Guilin, China.
36. Zheng, Z., Xie, S., Dai, H. N., Chen, W., Chen, X. et al. (2020). An overview on smart contracts: Challenges, advances and platforms. *Future Generation Computer Systems, 105,* 475–491. https://doi.org/10.1016/j.future.2019.12.019
37. Thumbur, G., Rao, G. S., Reddy, P. V., Gayathri, N. B., Reddy, D. V. R. K. (2020). Efficient pairing-free certificateless signature scheme for secure communication in resource-constrained devices. *IEEE Communications Letters, 24(8),* 1641–1645. https://doi.org/10.1109/COML.4234
38. Li, J., Wang, Y., Ding, Y., Wu, W., Li, C. et al. (2021). A certificateless pairing-free authentication scheme for unmanned aerial vehicle networks. *Security and Communication Networks, 2021,* 9463606:1–9463606:10. https://doi.org/10.1155/2021/9463606