Secure Caching Scheme Using Blockchains for Unmanned Aerial Vehicle-Assisted Information-Centric Wireless Sensor Networks

Shintaro Mori

Department of Electronics Engineering and Computer Science, Faculty of Engineering, Fukuoka University
8-19-1 Nanakuma, Jonan-ku, Fukuoka 814-0180, Japan
E-mail: smori@fukuoka-u.ac.jp

Abstract This paper proposes a caching data protection scheme that uses blockchains and hashchains for unmanned aerial vehicle (UAV)-assisted information-centric wireless sensor networks in smart-city applications as a case study. In the consensus scheme of the proposed scheme, a new block is verified according to voting by validators without exhaustive mining computations. Therefore, the proposed scheme is suitable for resource-limited wireless and mobile devices. With the scheme, UAVs work as validators as well as data collectors, and the overhead of transaction-fee management can be eliminated. Validators are responsible for block verifications; thus, the proposed scheme maintains the consensus by detecting and removing validators that vote incorrectly by using hashchain-based signatures. Simulations were conducted to evaluate the proposed scheme, and the results revealed the requirements for implementing it in smart-city applications and its robustness, feasibility, and effectiveness.

Keywords: wireless sensor network, information-centric network, unmanned aerial vehicle, blockchain

1. Introduction

Urbanization and the rapid increase in population density present various challenges to researchers and developers. Smart cities have emerged as a hub of intelligent and sophisticated applications ranging from public infrastructure management to personal healthcare [1]. This paper focuses on smart-city applications and wireless sensor network (WSN) systems. There are various interconnected devices, such as sensors, actuators, and distributed storages; therefore, such network-linked nodes generate a massive amount of data called big data. These sensing data need to be effectively collected, processed, and analyzed with continuous and non-disruptive provisioning. In particular, the first step of gathering data at a central location is of interest, i.e., the sensing data collection mechanism.

In typical smart-city frameworks, sensor nodes (SNs) are directly connected to cloud servers to gather and centralize sensing data via application programming interfaces (APIs) on the basis of Internet protocol stacks. However, these frameworks have the following structural issue. Individual SN devices have different constraints, data formats, and specifications, which raise concerns about interoperability and heterogeneity. Address-based networks on standard protocols create a potential bottleneck for handling dynamic wireless data transmissions. In data retrieval, heavy address-based queries cause serious protocol overheads, and wireless forwarding may lead to extra network congestion and frame collisions.

An information-centric network (ICN) design [2] is a promising content-oriented network model to address the above issues. Instead of the “address,” ICNs give a name as named-data-object for sensing data; thus, ICNs can deliver, store, and retrieve sensing data independently from the current location. ICNs have an in-network caching scheme and name-based routing scheme among intermediate nodes, which can handle further responses. Note that ICNs copy and store the duplicated data in the cache memory at the individual network nodes without using exclusive content-delivery servers, which is different from the content-delivery network scheme in current Internet systems. The concept of ICNs has been introduced into wireless networks [3], in particular, information-centric wireless sensor networks (IC-WSNs). The in-network caching scheme of an ICN can efficiently handle...
the sensing data from unavailable nodes and minimize retrieval delay. A naming scheme can enable better data-information management and an easy data-retrieval mechanism. Owing to their address-free structure, ICNs can provide a better hands-off mechanism for mobile devices.

Practical smart-city applications require the collection, aggregation, and storage of massive and rich multimedia sensing data, such as images and videos, the quantity of which has been rapidly increasing. Such heavy network traffic is a significant challenge for content collection and delivery in WSNs. Moreover, SN devices are unevenly scattered depending on the ground surface. SNs are deployed in not only urban areas but also rural areas, i.e., SNs are used in environments with poor radio propagation. Unmanned aerial vehicles (UAVs), such as drones and small planes, are expected to be widely used in the future to enable the proliferation of various applications, e.g., aerial delivery, disaster rescue, and remote sensing [4][5]. In WSNs with UAV assistance, UAVs are dispatched to collect sensing data from an observation area and have superior flexibility, mobility, and robustness as aerial relay nodes for data collection and forwarding, which can play an essential role in air-ground integration networks. As a result, energy consumption can be reduced and transmission reliability increased due to short-range communication with ground nodes via an air-to-ground line-of-sight wireless link, even if SNs are sparsely deployed over a wide area. Consequently, UAV-assisted IC-WSNs (UAV-IC-WSNs) are believed to be suitable for smart cities.

When running practical applications, sensing data have been subjected to various illegal attacks caused by the wireless nature of the communication medium and inherent device limitations such as heterogeneity, scalability, location, provider, and ICN-specific features [6]. This paper focuses on an ICN model’s weakness, i.e., the in-network caching scheme is vulnerable to cache pollution attacks. Cache pollution causes ICN structural flaws; therefore, the integrity of duplicate copied data (replica) cannot be guaranteed. If the caching data are illegally rewritten, IC-WSNs cannot correctly execute data retrieval, repair malfunctioning actuators, or guarantee data reliability. A variety of solutions have been studied to prevent cache pollution, but all are aimed at traditional address-based networks. This paper focuses on caching-data integrity and ignores data confidentiality (preventing unauthorized leakage of caching data) and data availability (ensuring caching data can be accessed when needed by authorized users). This is because confidentiality and availability can be provided by a data encryption technique and access control technique, respectively. Note that IC-WSNs can be constructed on centralized (cloud) servers or decentralized blockchains, whose consideration is beyond the scope of this paper (i.e., the aim of this paper is to prevent cache pollution).

This paper proposes a secure caching scheme that uses blockchains. The objective of using blockchains is to protect the caching data stored in each node from cache pollution. Since the system can prevent the caching data from being rewritten once it has been registered, blockchains are reasonable for this purpose. In addition, blockchains do not require a central authority, a third party to provide a guarantee, an intermediary, Byzantine fault tolerance, or a chronological ledger maintained among distributed nodes. Therefore, we select blockchains as a suitable technology in an autonomous decentralized environment, such as ICNs and WSNs.

From the above perspective, in our previous study [7], we proposed a secure caching scheme using blockchains for ICN-WSNs. In our previous study, however, we adopted a proof-of-work (PoW) scheme as a block verification (to commit a new block to append a blockchain among all member nodes). In other words, if blockchains are introduced into WSNs, managing the entire blockchain-based ledger and verifying transactions require a large amount of computation resources, but WSN nodes do not have sufficient capability because of hardware limitations. In the verification process, typical verification tasks require exhaustive computer calculations, such as the PoW, which are also applicable in Bitcoin [8], Ethereum [9], and other cryptocurrencies. Unlike the PoW, the proof-of-stake (PoS) does not require exhaustive computer calculations but is not suitable in an equal peer relationship since richer nodes have more chances to verify, resulting in a bias.

To tackle the above technical issues, the proposed scheme uses a voting-based consensus scheme. Blocks can be verified among validators, i.e., a block is considered verified if almost all validators give approval. In the proposed scheme, UAVs are assigned the role of not only data collectors but also validators. Therefore, the proposed scheme has the advantage that the blocks can be verified during usual data transferring in ad hoc and multi-hop wireless networking among UAVs. As a result, there is no extra overhead required to introduce the scheme. There is also no need to exchange transaction fees, e.g., a reward in Bitcoin and gas in Ethereum, for block-verification jobs. The proposed scheme thus mitigates network traffic congestion and eliminates transaction-fee management.

Validators play an essential role in providing a sufficient consensus for each block. Since the proposed scheme approves a block based on the validator’s votes, the scheme maintains the consensus by detecting and removing validators that vote incorrectly. To realize the mechanism, the proposed scheme uses hashchains with traceability and integrity.
The contributions of this paper are as follows.

- A secure caching scheme that uses blockchains for UAV-IC-WSNs is proposed. In particular, we consider the scenario of smart city applications and demonstrate the feasibility.

- The proposed scheme uses a novel voting-based consensus scheme suitable for UAV-IC-WSNs. Note that the proposed consensus scheme replaces the PoW scheme in our previous studies.

- The proposed scheme includes a novel mechanism to maintain the consensus by detecting and removing validators that vote incorrectly using hashchain-based signatures.

The remainder of this paper is organized as follows. Section 2 presents related work. Section 3 describes the proposed scheme. Section 4 presents the simulations conducted to evaluate the proposed scheme and the results. Finally, in Section 5, findings are summarized and conclusions are provided.

2. Related Work

In autonomous and distributed networks such as ICN-WSNs, blockchains have been introduced into various IoT systems. For example, Hamdaoui et al. [10] proposed the blockchain-based authentication, registration, and management of participatory IoT devices. Tahir et al. [11] surveyed state-of-the-art applications of blockchain technologies in fifth-generation (5G) cellular networks and beyond. Directed acyclic graph (DAG)-based blockchains, such as IOTA [12], have been developed as a new generation of blockchains suitable for IoT networks. Yang et al. [13] addressed the capacity limitation in vehicular nodes, and Hassija et al. [14] proposed a lightweight DAG-based blockchain protocol for vehicular networks.

Regarding blockchain technologies for smart city applications, Hakak et al. [15] investigated a conceptual architecture for securing smart cities using blockchains, and Hamdaoui et al. [16] proposed an abstraction for virtualized cloud-enabled IoT infrastructure. For individual solutions in smart cities, Guo et al. [17] proposed a blockchain-based electricity-trading ecosystem. Asamoah et al. [18] proposed several digital city management methods to secure individual residents’ identifications with a blockchain for key management. Regarding UAV-assisted wireless networks with blockchains, Jiao et al. [19] proposed a stability-aware consensus protocol for providing resource-sharing services in mobile ad hoc networks, Khan et al. [20] investigated outsourcing network coverage of a specific area using blockchains and auction mechanisms, and Lei et al. [21] proposed a systematic framework that integrates interest-key-content binding, forwarding strategy, and on-demand verification to discover poisoned content for UAV-IC ad hoc networks.

The above studies built blockchain networks on cloud servers or intelligent and rich edge networks. Edge-side nodes with sufficient computational capability to use a consensus algorithm are deployed, which is a reasonable solution. However, the goal of the current study cannot be achieved by simply introducing these blockchain technologies. The most significant challenge in deploying blockchains in WSNs is reducing the computing resources and delays for the verification process. There are two types of consensus algorithms: proof-based and vote-based. The former is adopted on almost all current frameworks; nevertheless, competitive transactions are not applicable for resource-limited WSN devices. The proposed scheme, however, can be used when computer resources are scarce without a transaction fee by adopting a consensus algorithm. Since the consensus algorithm in the proposed scheme is based on a voting scheme, it resembles proof-of-authority (PoA) and proof-of-reputation (PoR), which are used in payment systems of IoT devices [22]. PoA and PoR are stricter for recruiting validators than our consensus scheme. After the certified third-party organization examines validators in terms of names, affiliations, reliable records, and so on, the validators can be selected. Therefore, PoA and PoR are not suitable for UAV-IC-WSNs because of their centralized validator management.

3. Proposed Scheme

3.1 System description

SNs are massively deployed in the observation field and divided into several regions. The segmented area has one base station (BS) for comprehensive management, as shown in Fig. 1. There are four network components, and
the entities that logically play roles in sensing-data collection and dissemination are as follows.

• An SN behaves as a publisher in an ICN system, i.e., SNs periodically measure sensing data as atomic data. Atomic data are protocol data units in an ICN system.

• A UAV collects atomic data from SNs and summarizes them into a block. A block is a protocol data unit in a blockchain. UAVs also work as validators, and a local blockchain is constructed on them. Note that UAVs have fewer hardware limitations than SNs.

• A BS manages the WSNs where its territory, i.e., BS, works as a coordinator to determine which block to append into the blockchain. A BS works as a gateway to mediate between local blockchains and blockchains located at the upper tier for scalability.

• A user behaves as a subscriber in an ICN system, i.e., users send a data-retrieval request such as an interest packet to UAVs, BSs, or cloud servers. Users then obtain sensing data from the original data or the caches in the blockchain networks.

In the proposed scheme, sensing data are packed into an atomic data packet, including meta-tag information, and users can find their required data using the meta-tag information as clues. Regarding a decision rule for a publisher when data are being retrieved, if the required data are located in the same region, i.e., if the publisher and subscriber are in the same city, the nearest publisher responds to the subscriber. However, if the publisher is in city $\mathbb{A}$ and the subscriber is in city $\mathbb{B}$, city $\mathbb{A}$’s BS responds to the subscriber via city $\mathbb{B}$’s BS. The proposed scheme uses the flat-type naming rule within the same region as the addressing formula. If the data are transferred via outer regions, the scheme uses the hierarchical-type naming rule, and the vertically structured relationship is assigned through BSs.

Atomic data are collected by UAVs in the proposed scheme, which has two advantages. UAVs can work as movable coverage cells for load balancing (offloading). Furthermore, UAVs can communicate with SNs (line-of-sight) anywhere, even if there are poor radio-reception areas (blind zones). Note that, if the desirable bandwidth of network traffic is chronically and continuously required in the hotspot area and in the cell-edge area, it is assumed that the UAV’s task is replaced with BSs for cost-effectiveness. Regarding connectivity between SNs and UAVs, a start topology based on low-rate and long-range (LPWA) communications is assumed.

The battery of a UAV is limited; thus, it is necessary to return to the charging spot periodically. It was assumed in this study that UAVs fly along a predefined flight path. Regarding wireless connectivity between UAVs or between UAVs and BSs, high-speed and wideband transmission is required; thus, microwave radio transmissions are assumed, such as 5G cellular networks and beyond. As mentioned above, UAVs fly on a circular route; therefore, one UAV is determined as a head UAV and can transfer blocks to a BS. In addition, the number of UAVs to be deployed should be decided on the basis of the environment of the individual observation area, the communication specifications, the deployment of SNs on the ground, radio propagation, and the ground surface. In Sec. 4, we will provide numerical examples from the perspective of feasibility of deployment in a smart city.

3.2 Scalability of proposed blockchain scheme

For scalability, blockchains in the proposed scheme can be functionally systematized as a hierarchical blockchain structure, and we define the blockchain at the bottom (edge-side) tier as a local blockchain, as shown in Fig. 2. A local blockchain will be deployed in UAV-IC-WSNs as metropolitan WSNs (this study is primarily focused on the local blockchain scheme), and the local blockchain stores duplicate copies of blocks as caching data on individual UAVs. The local blockchain that each UAV manages is commonly shared in the same UAV network. The reason why UAVs also store caching data is for the purpose of ICN data retrieval. Namely, as flying BSs, the UAVs store the data as caching data similarly to cloud servers. The blockchains located at higher tiers than the local blockchains, such as regional blockchains, county blockchains, state blockchains, and national blockchains,

![Fig. 2 Structure of proposed scheme](image-url)
manage the inventory of local blockchain data. With this hierarchical structure, the proposed scheme is scalable.

### 3.3 Structure of proposed scheme

The verification procedure of the proposed scheme uses a voting-based consensus in which a new block is verified according to voting by validators. Namely, if a block obtains sufficient validator agreements, it can be regarded as a correctly verified block. As shown in Fig. 2, UAVs are assigned as validators; thus, these UAVs should verify the blocks. There are always flying UAVs as active validators by switching the role of validator to an alternative UAV when a UAV lands for charging and repairing is assumed. Note that the basic idea behind this study is not to invent a theory for maintaining Byzantine fault tolerance but to solve this problem with an engineering approach to ensure that the scheme works as desired. The traceable signature of the candidate blocks can highlight malfunctioning UAVs to help sustain the consensus mechanism. With the proposed scheme, no further verification is required if verified blocks are appended to the blockchain, no key management is necessary for verification of every user, and the overhead of key exchange can be reduced regardless of the locality of the region.

Regarding data format, the proposed scheme uses two protocol data units, i.e., atomic data packets and block packets, as shown in Fig. 2. An atomic data packet is used for ICN systems and a block packet is used for blockchains. Atomic data consist of meta-tag information and sensing data. Meta-tag information includes an SN’s identification number, the timestamp when atomic data were generated, an SN’s location, and a description (attribute) of the sensing data. There are two types of blocks used as protocol data units of a blockchain: candidate and chained. These blocks have different data structures, i.e., different data types are used depending on the block status. A candidate block has not yet been verified and is first generated by a UAV that collects atomic data.

Let the nth UAV (\( n \in \{1,2,\cdots,N\} \)) collect the atomic data \( x_q = \{x_q^1,x_q^2,\cdots,x_q^K\} \), which are packed into a candidate block \( B_q \). Then the packet \( B_q \) is constructed from

\[
B_q = q \oplus x_q
\]

where \( \oplus \) denotes any bit-level combining function.

To distinguish one block from another, \( B_q \) has a unique fingerprint \( q \). \( q \) is calculated using a hash-function and is expressed as

\[
q = h_n(x_q)
\]

where \( h_n \) denotes the nth UAV’s hash function, i.e., \( h_n \) is defined on the basis of Eq. (3) for any bit sequence \( b \);

\[
h_n(b) \triangleq h(c_n \oplus b)
\]

where \( h(\cdot) \) denotes a general hash function

In Eq. (3), \( c_n \) denotes a unique key value for an individual UAV, and the pair of node id \( n \) and \( c_n \) is managed on the BS as a database \( T \), which is expressed as

\[
T = \{(1,c_1),(2,c_2),\cdots,(n,c_n),\cdots,(N,c_N)\}
\]

### 3.4 Procedure of data collection, verification, and appending to blockchain in proposed scheme

As shown in Fig. 3, the proposed scheme involves three steps: collecting sensing data, conducting block verification, and appending the verified blocks to the blockchains. In the data-collection step, a large amount of atomic data is collected by UAVs and summarized as a candidate block (\( \bigodot \)) at the UAV assigned to the data collection. In the verification step (\( \bigodot \)), candidate blocks are multi-hop forwarded among UAVs and reach cloud servers via the BS. The proposed scheme has three different roles during ad hoc wireless data transfer: data forwarding, block verification, and data caching. Note that they correspond to the relevant tasks of a WSN, blockchain, and ICN, respectively. Block verification is an essential
component in blockchains. In the proposed scheme, the validator verifies the candidate block, and the validators vote for the block, and then the BS approves the block based on the voting result. If the validator does not approve a candidate block during transferring, the block will be rejected and dropped without further forwarding. Note that typical blockchains have not only a method to select which blocks to append to the blockchain, but also another system to verify the correctness of the blocks. This study focuses on the former as a reasonable way to gain a consensus to deploy blockchains in UAV-IC-WSN. Thus, we assume that the UAV can validate the block.

The consensus procedure ensures the tamper-proof security of chained blocks and validation. In the proposed scheme, a new block can be regarded as a verified block when it receives an agreement of at least two-thirds among validators according to Byzantine fault tolerance. In practice, the threshold of the consensus should have redundancy because a few UAVs cannot be properly signed, which aims to eliminate the effects of illegal voting.

Before the approval of a block linked next to the blockchain, the BS selects a head UAV among the UAVs (Ω). To avoid system complexity, the head UAV is determined as that nearest to the BS in this study. After the head UAV is chosen, the BS selects a new chained block (that might not necessarily be appended to the blockchain). Namely, the BS receives candidate blocks signed by validators from the head UAV. Candidate blocks arrive from multiple routes; thus, their voting results are merged at the BS (Ω). When the BS can ensure that the block has a set of active validators’ signatures based on the aggregated voting results, the BS approves the block as a chained block. On the other hand, if a candidate block does not obtain the consensus, the candidate block cannot be verified and joined to the blockchain. In addition, when the decoding process of the hashchain-based signatures fails, the decoded signatures before the failure are valid, otherwise they are invalid. Note that the hashchain can be decoded and traced in the backward direction along the path of block forwarding. When the BS fixes a chained block to an appended block, the BS calculates a new hash value of the final and fixed chained block on the basis of the atomic data, the consensus information, and the previous block’s hash value instead of the fingerprint (Ω).

When updating the local blockchain, the renewed chained block is duplicated and distributed among UAVs to provide the IC-WSN’s caching scheme (Ω). The BS shares the inventory information extracted from the renewed chained block to the blockchain located at an upper tier of the local blockchain (Ω); thus, users can obtain sensing data from a cross section of cities. When the current head UAV leaves the area of the dominant BS’s coverage, the current head UAV swaps with a new one that the BS reselects.

3.5 Hashchain-based signature for detecting UAVs with fraudulent voting

In the verification procedure of the proposed scheme, a validator records a signature on a candidate block, and a candidate block can be regarded as a verified block when a sufficient consensus of validators is confirmed. To approve the block, the proposed scheme uses hashchains when their one-directionality. In our scheme, a hashchain uses the block-signature mechanism as well as detects illegal UAVs. When each block is ad hoc forwarded among validators and a validator approves the block, the validator appends a hashchain-based signature to the block. If the hashchain-based signature cannot be properly decoded, it can be determined that one or more validators along the route where the block is transferred do not proceed with a correct ballot. A failure to decode may be due to hijacking by malicious attackers, device failures, and human errors.

A hashchain is generally a type of blockchain where a hash function is used to connect blocks of information sequentially. By letting any hash function \( H() \) be iteratively applied \( \kappa \) times to any bit sequence \( b \), a general hashchain is defined as \( \{ H^1(\cdots), H^2(\cdots), \ldots, H^\kappa(\cdots) \} \), where \( H^\kappa(b) \) is given by

\[
H^\kappa(b) = H^\kappa ( H^{\kappa-1}( \cdots ( H^1 (b) ) \cdots ) )
\]  

The hashchain-based signature mechanism of the proposed scheme is as follows. It is assumed that any candidate block \( B_{q,\ell} \) is forwarded via the \( \ell \)th wireless transmission path ( \( \ell = \{1,2,\cdots,L\} \) ), and the group of validators \( U_{q,\ell} = \{ u^{\ell}_{q,1}, u^{\ell}_{q,2}, \ldots, u^{\ell}_{q,M} \} \) gives the signatures for the candidate block \( B_{q,\ell} \). For \( U_{q,\ell} \), any \( u^{\ell}_{q,m} \) \((m = \{1,2,\cdots,M\}) \) represents UAV identifications \( n \) \((n = \{1,2,\cdots,N\}) \), i.e., \( \forall u^{\ell}_{q,m}, \exists n \).

In the signature procedure of the proposed scheme, the first validator’s signature is a hash value for the entire header and payload, and the second and following signatures are the hash values of the previous signature up to the number of validator groups \( M \). Therefore, for \( B_{q,\ell} \), \( u^{\ell}_{q,m} \)'s signature \( s^{\ell}_{q,m} \) is calculated using Eq. (6). In addition, \( B_{q,\ell} \)'s signatures are given by \( S_{q,\ell} = \{ s^{1}_{q,\ell}, s^{2}_{q,\ell}, \ldots, s^{M}_{q,\ell} \} \).

\[
\begin{align*}
S^{1}_{q,\ell} &= H^{1}_{\ell} ( B_{q,\ell} ) \\
S^{2}_{q,\ell} &= H^{2}_{\ell} ( s^{1}_{q,\ell} ) = H^{2}_{\ell} ( H^{1}_{\ell} ( B_{q,\ell} ) ) \\
&\vdots \\
S^{M}_{q,\ell} &= H^{M}_{\ell} ( H^{M-1}_{\ell} \cdots ( H^{1}_{\ell} ( B_{q,\ell} ) ) )
\end{align*}
\]
where $H^\pi_m(\cdot)$ denotes the hash function of the UAV of $u^\pi_m$. Consequently, the chained block $\tilde{B}_{q,\ell}$, which corresponds to $B_{q,\ell}$, is calculated as

$$\tilde{B}_{q,\ell} = B_{q,\ell} \oplus U_{q,\ell} \oplus S_{q,\ell}$$ (7)

Finally, the procedure of detecting UAVs with fraudulent voting based on hashchain-based signatures is as follows. The blocks are multi-hop propagated across the UAV network in multiple paths. Hashchains guarantee that the UAVs on that path have been properly voted for if the hashchain is correctly decoded; otherwise, one of the UAVs along the path can be regarded as possibly having been voted for fraudulently. Therefore, the BS can summarize the decoding results of these multiple passes to identify the UAV with fraudulent voting. In the BS, the group of signatures along $L$ paths during the verification process for $B_{q,\ell}$ is summarized as $S_q = \{S_{q,1}, S_{q,2}, \ldots, S_{q,\ell}, \ldots, S_{q,L}\}$. On the other hand, the BS can examine the validity of each signature group using the managed table regarding the pair of the validator identification number and its key value, which is given by Eq. (4). Thus, the decision function can be defined as $\varphi(T, S_{q,\ell})$. If any hashchain is correctly traced, the UAVs can be considered uncompromised, $U^+$; otherwise, UAVs with fraudulent voting, $U^-$, are suspected:

$$U_{q,\ell}^+ = \{S_{q,\ell} \in S_q \text{ if correctly as } \varphi(T, S_{q,\ell})\}$$
$$U_{q,\ell}^- = \{S_{q,\ell} \in S_q \text{ otherwise as } \varphi(T, S_{q,\ell})\}$$ (8)

Therefore, the current UAVs with fraudulent voting $V^\tau$ can be determined on the basis of $U^+, U^-$, and the latest (previous) $V^{\tau-1}$, as

$$V^\tau = V^{\tau-1} \cap U_{q,\ell}^+ \cup V^{\tau-1} \cup U_{q,\ell}^-$$ (9)

When UAVs are grounded and recovered (for recharging and maintenance), they can be replaced with alternative UAVs or repaired on the basis of the alert information.

4. Simulations and Results

A preliminary evaluation was conducted involving computer simulations to reveal the requirements for the proposed scheme when implementing it in smart-city applications and its feasibility. These simulations were also conducted to estimate the parameter settings and conditions and clarify technical requirements for our scheme.

4.1 Simulation environments

The computer simulations were implemented in C++ language on a PC (Windows 10 OS, Core i5 2.9 GHz, and 16 GB RAM) to provide preliminary numerical results. SNs were deployed in a 100 km² area, and the simulation parameters were determined on the basis of Japan’s LPWA specifications [23]. It was assumed that there are always UAVs flying by changing to alternative UAVs and that BSs have sufficient power supplies. The received signal strength was calculated using a typical link budget formula [24]. For the link budget calculation, the model of Erceg et al. [25] was selected for the link between SNs and BSs, and the model of Amorim et al. [26] was selected for the link between SNs and UAVs as radio propagation models. These models are formulated on the basis of the experimental measurement for respective practical scenarios.

4.2 Deployment requirements of proposed scheme

The implementation of the proposed scheme in a smart-city application was simulated. Figure 4 shows the generated traffic of sensing data during environmental monitoring, i.e., the amount of atomic data, as a function of the number of SNs controlled by one UAV. The three colored curves correspond to the intervals of atomic data generation of 600 s (assuming a maximum situation in
typical WSN systems), 1,002 s (assuming an average situation in current WSN systems), and 3,600 s (assuming periodic service). As a result, the arrival probability degraded with an increase in the number of SNs because of collisions caused by congested wireless communication channels. Regarding arrival-rate degradation, collisions were more dominant than packet errors due to poor wireless communication channels. Figure 4 shows the number of SNs each UA could control, i.e., 60, 100, and 1,000, when the SN’s transmission intervals were 600 s, 1,002 s, and 3,600 s, respectively, at the required 5% packet error ratio (PER). When the same packet was repeatedly transmitted three times, the number of SNs each UA could control was 375, 700, and 2,430 when the SN’s transmission intervals were 600 s, 1,002 s, and 3,600 s, respectively.

Figure 5 shows the communication coverage area required for UAVs under dense SN device deployment, such as \( \rho = 10,000 \text{ /km}^2 \). A circular coverage area was assumed, i.e., the effects of shadowing and fading were ignored, and a received signal strength depending on the distance between nodes was considered as a feature of specific radio propagations. For example, when a UAV controlled 700 SNs, the radius of the UAV’s coverage on the ground was obtained as \( r = 150 \text{ m} \). Note that the above scenario was assumed when 5% PER was achieved through three iterations of transmission and when the atomic data transmission interval was set to 16.7 min.

Figure 6 shows the number of UAVs required to enable sufficient wireless communications with SNs scattered in the observation area, in other words, 98% of SNs are located within the communication range when the UAVs are deployed. The solid line indicates the case in which the UAVs were placed in a hexagonal cell with the maximum density layout, and the dashed line indicates the case in which the UAVs were randomly placed in a uniform distribution. Note that the solid and dashed curves represent the theoretical upper limit and worst lower bound, respectively; therefore, the curves for a realistic environment lie between both curves. As shown in Fig. 6, the number of UAVs per square kilometer was determined to be 16 (for an ideal case) and 80 (for the worst case) when the radius of the required communication coverage was set to 150 m on the basis of Figs. 4 and 5.

### 4.3 Requirements, robustness, and feasibility of proposed scheme

In the previous section, it was found that the radius of the circular radio communication area in which a UAV should provide coverage is 150 m when the average transmission interval of the atomic data is set as 16.7 min and the required packet loss is set as 13.5% (in the case of three transmissions via a wireless link with 5% PER). In this case, the number of UAVs required to cover the observation area (10 km² square surface) was 1,600 (for an ideal case) and 80,000 (for the worst case).

Figure 7 shows the number of multi-hops via ad hoc wireless networks until candidate blocks obtained the consensus. In the simulation, as the threshold of consensus, we decide on 80%, which is based on realistic cryptocurrencies, such as Ripple and other cryptocurrencies that have a similar consensus mechanism to that of the proposed scheme [27][28]. We believe that the
parameter setting is reasonable since Ripple is used in interbank transactions. Let $R$ denote the communication range between UAVs and $r$ denote the coverage radius of UAVs; therefore, the parameter $R/r$ represents the ratio between the available wireless UAV-UAV and UAV-SN communication distances. Note that $R$ and $r$ are considered separately because UAVs in the sky are in line-of-sight communication with each other, i.e., $R$ is larger than $r$ even if the wireless link is under the same conditions, such as communication protocols and frequency bands. For example, when the number of UAVs was 1,600 and $R/r = 4$ (i.e., $R = 2$ km and $r = 0.5$ km), it was found that the consensus can be achieved with an average of 10 hops.

Figure 8 shows whether UAVs with fraudulent voting can be detected from a mixture of hashchain-based signatures of the proposed scheme via multiple routes. It was assumed that ten signatures are given for one block on the basis of the results in Fig. 7. When 5% of the total UAVs did not correctly vote, the proposed scheme detected them with 80% accuracy. When more than 10% did not correctly vote, the scheme could detect all of them. The reason the detection accuracy increases as the percentage of UAVs with fraudulent voting increases depends on the number of verified signatures and their cross-checking. In addition, in Fig. 8, the probability of successfully detecting UAVs with fraudulent voting is not always 1. This is because the proposed scheme cannot detect the UAVs with fraudulent voting if it generates a candidate block and gives the first signature. However, we believe that we need not seriously consider this drawback since the threshold of the consensus has a certain redundancy. Note that, in Ripple [28], which is an interbank cryptocurrency network that has adopted a voting-based consensus scheme, there should be no more than 20% failed nodes, which satisfies Byzantine fault tolerance. Therefore, the proposed scheme is feasible since it can satisfy the requirements of such schemes.

### 4.4 Comparison with conventional schemes

In this section, we will demonstrate that the proposed scheme can overcome the remaining problems outlined in our previous study [7]. As far as we know, the PoW scheme (including Ethereum) is utilized as the consensus scheme for typical related systems. Compared with these schemes, the proposed scheme has the advantages of reducing the transmission of messages by one round trip and eliminating the exhaustive hash calculations.

Regarding the forwarding messages, the conventional scheme should transmit messages for two round trips since the data collection process and the verification process are separately conducted. Namely, the data collectors gather the sensing data and forward the data to the coordinator, and the coordinator sends the mining request message to the miner nodes. After the coordinator receives the mining result from winner nodes, the information of the verified block is broadcasted. On the other hand, the proposed scheme requires only one round-trip message transmission since the data collection process and the verification process can be done at the same time. As a result, the
proposed scheme can reduce the number of message transmissions by half.

In the verification procedure, the conventional scheme conducts mining jobs, i.e., it requires a large amount of hash calculations depending on the parameter of the difficulty in the cryptographic puzzles. On the other hand, the proposed scheme is based on voting; thus, the scheme requires only one hash calculation for the hashchain-based signature. As a result, the proposed scheme can reduce the number of computer calculations, i.e., it can reduce energy consumption, making it suitable for deployment on resource-limited wireless nodes. To demonstrate the above advantage, we developed a hardware-based testbed device using the familiar single-board computer of a Raspberry Pi 3 with the default Raspbian kernel version 10. As in our previous study [7], for a hash function, we used the first-generation secured hash algorithm (SHA-1) method [29], which is widely utilized for general purposes. We implemented the prototype program code using C++ language with the CLX C++ library [30] and we compiled and linked an executable binary program by using the GNU GCC compiler.

Figure 9 shows the energy consumption, which is obtained by integrating the consumed power (wattage) of the testbed measured at 2 ms intervals. The verification task begins 1 min. after the start of the monitoring, and the result represents a 10 min. period. Note that the difficulty of the conventional PoW method should be adjusted according to the individual environment; in particular, we set it as \( k = 17 \) for the experiment.\(^1\) As shown in Fig. 9, after the mining process has started, in the conventional scheme, with that for the gradient of energy consumption increases compared with that for the idle state. On the other hand, the proposed method had a short-term peak in power consumption; however, it had a negligible effect on the curve of energy consumption. As a result, the proposed scheme can reduce the energy consumption by 19.1% compared with that of the conventional scheme for 10 min. of measurement for one node.

5. Conclusion

A secure caching scheme using lightweight voting-based blockchains was proposed. To maintain the consensus, the proposed scheme uses hashchain-based signatures. Simulations were conducted to evaluate the scheme, and the results revealed the requirements of implementing the proposed scheme in smart-city applications and its robustness, feasibility, and effectiveness. As future work, it is necessary to improve the scheme to detect when only a few UAVs do not correctly vote, and the scheme should be evaluated in realistic situations, including by hardware experiments.

Acknowledgment

Part of this work was supported by JSPS KAKENHI Grant Number JP19K20261.

References

[1] A. Kirimtat, O. Krejcar, A. Kertesz and M. F. Tasgetiren: Future trends and current state of smart city concepts: A survey, IEEE Access, Vol.8, pp.86448-86467, 2020.
[2] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher and B. Ohlman: A survey of information-centric networking, IEEE Commun. Mag., Vol.50, No.7, pp.26-36, 2012.
[3] S. Arshad, M. A. Azam, M. H. Rehmani and J. Luo: Recent advances in information-centric networking-based Internet of things (ICN-IoT), IEEE Internet Things J., Vol.6, No.2, pp.2128-2158, 2019.
[4] F. Qi, X. Zhu, G. Mang, M. Kadoch and W. Li: UAV network and IoT in the sky for future smart cities, IEEE Network, Vol.33, No.2, pp.96-101, 2019.
[5] R. A. Nazib and S. Moh: Energy-efficient and fast data collection in UAV-aided wireless sensor networks for hilly terrains, IEEE Access, Vol.9, pp.23168-23190, 2021.
[6] B. Nour, K. Sharif, F. Li and Y. Wang: Security and privacy challenges in information-centric wireless

---

\(^1\) In the mining job, the hash values must satisfy the condition that the bit sequence is followed by a long string of zeros with \( k \) bits from the most significant bit.
Internet of things networks, IEEE Secur. Privacy, Vol.18, No.2, pp.35-45, 2020.
[7] S. Mori: Secured caching scheme by using blockchain for information-centric network-based wireless sensor network, J. Signal Process., Vol.22, No.3, pp.97-108, 2018.
[8] S. Nakamoto: Bitcoin: A peer-to-peer electronic cash system, http://Bitcoin.org/Bitcoin.pdf, 2008. (retrieved: July 2021).
[9] Ethereum: https://www.ethereum.org/ (retrieved: July 2021).
[10] B. Hamdaoui, M. Alkalbani, A. Rayes and N. Zorba: IoTShare: A blockchain-enabled IoT resource sharing on-demand protocol for smart city situation-awareness applications, IEEE Internet Things J., Vol.7, No.10, pp.10548-10561, 2020.
[11] M. Tahir, M. H. Habaebi, M. Dabbagh, A. Mugees, A. Ahad and K. I. Ahmed: A review on application of blockchain in 5G and beyond networks: Taxonomy, field-trials, challenges and opportunities, IEEE Access, Vol.8, pp.115876-115904, 2020.
[12] IOTA: https://www.iota.org/ (retrieved: July 2021).
[13] W. Yang, X. Dai, J. Xiao and H. Jin: LDV: A lightweight DAG-based blockchain for vehicular social networks, IEEE Trans. Veh. Technol., Vol.69, No.6, pp.5749-5759, 2020.
[14] V. Hassija, V. Chamola, S. Garg, D. N. G. Krishna, G. Kaddoum and D. N. K. Jayakody: A blockchain-based framework for lightweight data sharing and energy trading in V2G network, IEEE Trans. Veh. Technol., Vol.69, No.6, pp.5799-5812, 2020.
[15] S. Hakak, W. Z. Khan, G. A. Gilkar, M. Imran and N. Guizani: Securing smart cities through blockchain technology: Architecture, requirements, and challenges, IEEE Network, Vol.34, No.1, pp.8-14, 2020.
[16] B. Hamdaoui, M. Alkalbani, T. Znati and A. Rayes: Unleashing the power of participatory IoT with blockchains for increased safety and situation awareness of smart cities, IEEE Network, Vol.34, No.2, pp.202-209, 2020.
[17] J. Guo, X. Ding and W. Wu: A blockchain-enabled ecosystem for distributed electricity trading in smart city, IEEE Internet Things J., Vol.8, No.3, pp.2040-2050, 2021.
[18] K. O. Asamoah, H. Xia, S. Amofa, O. I. Amankona, K. Luo, Q. Xia, J. Gao, X. Du and M. Guizani: Zero-Chain: A blockchain-based identity for digital city operating system, IEEE Internet Things J., Vol.7, No.10, pp.10336-10346, 2020.
[19] Z. Jiao, B. Zhang, L. Zhang, M. Liu, W. Gong and C. Li: A blockchain-based computing architecture for mobile ad hoc cloud, IEEE Network, Vol.34, No.4, pp.140-149, 2020.
[20] S. Khan, G. Chen, Y. Rahulamathavan, G. Zheng, B. Assadhan and S. Lambotharan: Trusted UAV network coverage using blockchain, machine learning, and auction mechanisms, IEEE Access, Vol.8, pp.118219-118234, 2020.
[21] K. Lei, Q. Zhang, J. Lou, B. Bai and K. Xu: Securing ICN-based UAV ad hoc networks with blockchain, IEEE Commun. Mag., Vol.57, No.6, pp.26-32, 2019.
[22] T. A. Alghamdi, I. Ali, N. Javaid and M. Shafiq: Secure service provisioning scheme for lightweight IoT devices with a fair payment system and an incentive mechanism based on blockchain, IEEE Access, Vol.8, pp.1048-1061, 2020.
[23] ARIB STD-T108: https://www.arib.or.jp/english/ (retrieved: July 2021).
[24] J. G. Proakis: Digital Communications, McGraw-Hill, 2008.
[25] V. Erceg, L. J. Greenstein, S. Y. Tjandra, S. R. Parkoff, A. Gupta, B. Kulic, A. A. Julius and R. Bianchi: An empirically based path loss model for wireless channels in suburban environments, IEEE J. Sel. Areas Commun., Vol.17, No.7, pp.1205-1211, 1999.
[26] R. Amorim, H. Nguyen, P. Mogensen, I. Z. Kovács, J. Vágard and T. B. Sørensen: Radio channel modeling for UAV communication over cellular networks, IEEE Wireless Commun. Lett., Vol.6, No.4, pp.514-517, 2017.
[27] B. Lashkari and P. Musilek: A comprehensive review of blockchain consensus mechanisms, IEEE Access, Vol.9, pp.43620-43652, 2021.
[28] Ripple: http://ripple.com/ (retrieved: July 2021).
[29] SHA-1 algorithm: IETF RFC 3174.
[30] CLX C++ library: https://github.com/clown/clx (retrieved: July 2021).

Shintaro Mori received his B.S., M.S., and Ph.D. degrees from Kagawa University in 2007, 2009, and 2014, respectively. Since April 2014, he has been with the Department of Electronics Engineering and Computer Science, Faculty of Engineering, Fukuoka University, Japan, where he is currently an assistant professor. His research interests include cross-layer design and wireless sensor networks. He is a member of IEEE, IEICE, ISSJ, and RISP.

(Received April 22, 2021; revised July 28, 2021)