Sensor-Chain: A Lightweight Scalable Blockchain Framework for Internet of Things

Conference Paper · May 2019

DOI: 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00195

4 authors, including:

Abdur Rahman Bin Shahid
Florida International University
11 PUBLICATIONS **14** CITATIONS

SEE PROFILE

Sensor-Chain: A Lightweight Scalable Blockchain Framework for Internet of Things

Abdur R. Shahid*, Corey Staier†, Rain Kwan‡, Niki Pissinou*
*School of Computing and Information Sciences, Florida International University, Miami, Florida
†Department of Computer Science, University of Mary Washington, Fredericksburg, Virginia
‡Computer Science, New College of Florida, Sarasota, Florida
Email: *{ashah044, pissinou}@fiu.edu, †cstaier@mail.umw.edu,‡rain.kwan16@ncf.edu

***Abstract*—The Internet of Things (IoT), forming the foundation of Cyber Physical Systems (CPS), connects a huge number of ubiquitous sensing and mobile computing devices. The mobile IoT systems generate an enormous volume of a variety of dynamic context data and typically count on centralized architectures to process them. However, their inability to ensure security and decline in communication efficiency and response time with the increase in the size of IoT network are some of the many concerning weaknesses that are holding back the fast-paced growth of IoT. Realizing the limitations of centralized systems, recently blockchain-based decentralized architecture is being considered as the key to redesigning the IoT systems in a way that is designed to be secure, transparent, highly resistant to outages, auditable, and efficient. However, before realizing the new promise of blockchain for IoT, there are significant challenges to address. One fundamental challenge is the scale issue around data collection, storage, and analytic as IoT sensor devices possess limited computational power and storage capabilities. In particular, since the chain is always growing, IoT devices require more and more resources. Thus, an oversized chain poses storage and scalability problems. With this in mind, the overall goal of our research is to design a lightweight scalable blockchain framework for IoT of mobile devices. This framework, coined as "Sensor-Chain", promises a new generation of lightweight blockchain management with a superior reduction in resource consumption, and at the same time capable of retaining critical information about the IoT systems of mobile devices.**

***Index Terms*—IoT, Blockchain, Storage, Mobility.**

I. Introduction

Over the past several years, there has been a surge of interest on Internet of Things (IoT) that connect physical infrastructures with machine intelligence, information and communication technologies based on sensors and Wireless Sensor Networks (WSN); and form the foundation of Cyber Physical Systems (CPS) [1]. From transportation, environmental monitoring, healthcare, smart homes, public security, wearables, to wildfire mapping, IoT systems are expected to have high impact on "everything". However, before realizing the promise of IoT, there are significant challenges to address. First of all, current IoT systems rely on centralized structures which are incapable of handling fast paced growth of IoT. The frequent change in the mobility-based IoT network due to node mobility, node failure, damage, energy depletion, or channel fading only further exacerbate the problems of centralized model. To overcome the limitations of centralized model, the blockchain based Peer-to-Peer(P2P) network models have gained significant attention in recent years. Blockchain is a distributed P2P way of recording digital interactions in a way that it provides built-in integrity of information, and security of immutability by design, making it very useful to ensure trust, security, and transparency in P2P trustless networks of huge number of devices [2], [3]. Although blockchain is considered as the key to redesign IoT systems, they cannot be directly integrated into IoT systems. Since the chain is always growing, IoT nodes require more and more resources in order to manage it on their local spaces. Similarly, scalability with constrained computing power and battery also poses a challenge. With an integration of blockchain, each node needs to perform large amount of tasks at different stages of the blockchain with their constrained computing power and battery life. The growth of the network further aggravates the problem.

These two issues are rooted to the problem of managing the number of transactions required to be stored and processed by a single IoT node at any time instance, as transactions are the main building blocks of a blockchain. For better understanding of the problem, let us consider a conventional blockchain for a P2P network of n number of nodes. At any time instance, there could be at most $\frac{n(n-1)}{2}$ number of transactions in the the network. If we express the size of a blockchain as the number of transactions stored it, then $size(\text{BC}) = \text{TX}^1 + \ldots + \text{TX}^T = \text{O}(T \times \frac{n(n-1)}{2}) = \text{O}(Tn^2)$. Where, TX^i refers to the number of transactions happened in the network at i-th time. This upper bound of the required space to store a blockchain clearly indicates the IoT sensor nodes will be run out of space in a short period of time.

In this paper, we focus on certain mobility-related scenarios where a mobile node is not really required to have a "global" view of a blockchain. Let us consider an environment monitoring mobile crowdsensing application where aggregated data (e.g. temperature, humidity, air quality, and so on) from a small region at a certain time is more important than individual's data. In such a scenario, the mobile nodes at a location may contact each other in a P2P way to collect each others environmental sensor's value for some time. Then, one node is selected to send the aggregated information in a certain form (e.g. max, mean, average, median, etc.). As the nodes are mobile, the trust value computed for some nodes may not be important at a different location and time for a certain node.

Also, the environmental data varies from one region to another; thus instead of having a single network, region based multiple smaller networks, as well as blockchains, are more feasible.

Against this backdrop, this research aims to address the blockchain management problem by designing a lightweight blockchain framework, named as "Sensor-Chain", for mobile IoT. We show that breaking down a traditional global blockchain into smaller "local" blockchains in spatial domain and limiting their size through a temporal constraint will allow us to design scalable blockchain for mobile IoT systems. The highlights of our contribution are as follows: 1) The Sensor-Chain blockchain framework consumes little storage space on the IoT sensor devices and is scalable with the increase in network size. We compare the performance of proof-of-concept implementation of sensor-chain with 3 other schemes and the results justify its superiority. 2) The proposed framework does not involve any fixed positioned powerful edge devices, which makes it more flexible with a variety of mobility-based IoT applications. 3) Sensor-Chain is independent of any particular ledger platform. Thus, it can be implemented with any platform (e.g. Ethereum, hyperledger, and so on) for IoT.

The rest of the paper is organized as follows. A discussion on the existing works is presented in section II. Afterwards, a gentle introduction on blockchains, and the fundamental concepts related to Sensor-Chain are discussed in section III. Then, section IV details the Sensor-Chain framework. Section V verifies the framework in terms of experimentation with synthetic data. Finally, section VI concludes the paper. Last but not least, important symbols used in this paper are described in table I.

II. Related Works

Understanding the limitations of the centralized model of IoT, recent focus has been shifted to developing decentralized architecture based on blockchain. A large number of the research works [4], [5] discussed the impact of blockchain on IoT and important research issues that are required to be addressed to fully realize the benefit of blockchain. The existing research efforts can be categorized into devising approach to integrate blockchain with IoT [6]–[9], node authentication and access control [10]–[13], trust management [5], [14]–[16], and security and privacy [17]–[19]. These different research works have one thing in common: they either simply considered that IoT devices are equipped with enough storage and computing resources to hold and process blockchains, or utilized high end edge computing devices to manage the blockchain. The assumptions of having enough resources is hard to get on with IoT devices, making the applicability of the research works based on such assumptions questionable. For instance, trust and authentication management for wireless sensor networks using blockchain was proposed in [14] without hinting how the sensors will manage the blockchain on their own local space. Likewise, the Block-VN architecture for distributed transport management system [20], based on a permissioned blockchain, considered that at least some portion of the vehicles are capable of storing and processing an ever-growing blockchain. Another example is the IoT-based Machine-to-Machine payment system, known as IOTA [21]. IOTA uses proof-of-work consensus protocol, which makes the new block creation task both computationally expensive and time consuming. Thus, in IOTA the hardware requirement is too high and it is hard to meet such requirement for IoT sensor nodes.

Realizing the resource issues of the IoT devices, many research works proposed to offload the blockchain onto edge computing devices. The SpeedyChain [22] data sharing framework for intelligent vehicles suggested to use roadside infrastructure units (RSIs) and service providers (SPs) to maintain blockchain. Here, RSIs are responsible for trust and authentication management, and trusted vehicles, verified by the RSIs, can append block to the blockchain. In a similar way, a Roadside Units (RSU) based blockchain trust management for vehicular network was proposed in [23]. In this work, each vehicle generates a rating for its neighboring vehicles and share the rating with nearby RSU. With all the most recently received ratings, RSUs calculate the trust value offsets of involved vehicles and gather these data into a block. In order to insert the new block into the blockchain, the authors proposed a combination of proof-of-work and proof-of-stake, improving each other. In contemporary works, Xiong et al. [6], [24] proposed to deploy multiple access mobile edge computing devices to carryout the computationally expensive proof-of-work and introduced game theoretic approach for edge computing resource management. In these works, the sensors are considered as ordinary nodes, and the edge devices are responsible for the blockchain operations. The "EdgeChain" framework [7] extended this idea by introducing credit-based resource management system to control the edge server resource consumption by an individual IoT device. In [10], a smart contract-based access mechanism was put forward with the aim of simplifying the process of blockchain management and reducing the communication overhead between the nodes. In this mechanism, the IoT devices are kept out of the blockchain as they cannot hold a large blockchain. Rather, a special node called management hub is proposed to put as a link between IoT devices and blockchain. A blockchain framework was proposed for smart homes [25], where the information produced by smart home devices are stored in the blockchain. In this architecture, the blockchain is maintained in the gateways and is isolated from the devices. Similar to the other works on blockchain based Internet of Vehicles, kang et al. [22] also considered RSUs as edge computing infrastructures for blockchain management. This approach utilized a modified high-efficiency Delegated Proof-of-Stake (DPoS) consensus scheme where instead of stake-based voting, reputation is used for miner RSU selection.

Through a careful observation of all these approaches, one can figure it out that that they all tried to solve the storing and processing heavyweight blockchain problems by employing more powerful computing devices in the architecture. As discussed earlier in section I, such structured deployment is hardly achievable, as the network topology is prone to changes

TABLE I
TABLE OF SYMBOLS

Symbol	Description
TX	Abbreviation of transaction
BC	Abbreviation of blockchain
$\mathcal{C}$	Voronoi diagram or set of Voronoi cells
C_i	i-th Voronoi cell or simply cell
n	Total number of sensor nodes
m	Number of sensors in a single cell
G_i^t	Local network in i-th cell at time t
V_i^t	Set of vertices of local network G_i^t
E_i^t	Set of edges between the nodes in V_i^t
S	A sensor node
B_i^t	Local blockchain generated by G_i^t
T_{chain}	Temporal constraint for blockchain
T_{block}	Block generation time constraint

very frequently in many IoT scenarios.

One viable solution to make blockchain "manageable" for sensors without using any edge or other devices is limiting the size of the blockchain within the resource capacity of sensors. The "temporal blockchain" framework proposed a solution based on such concept [26]. It was proposed to delete all the blocks older than a preset period (e.g. 30 days old) from the blockchain. While this approach can reduce the size of the blockchain, it still lacks in guaranteeing limited storage capacity with the growth of the network in the long-run in IoT scenario. Moreover, how to deal with the loss of information due to the deletion of blocks was not addressed.

This study highlights that existing blockchain frameworks lack a clear understanding of the resource management issues for blockchain in IoT scenario. Lack of such understanding makes the frameworks highly impractical for IoT. The research on blockchain and IoT has a long way to go, and we emphasize that before taking further steps, we must have an efficient approach to make blockchain lightweight and scalable for IoT sensors. In light of this, this research proposes the first lightweight scalable blockchain framework for IoT.

III. PRELIMINARIES

In this section we present a formal introduction to blockchain, the proposed system model and the assumptions made about it.

Blockchain, in a sense, is a sequence of ledgers that are connected by hash values. These ledgers contain transactions that have taken place over time in the form of blocks of data [2]. Each block has two components: block header and transaction list. The block header contains a cryptographic hash of the previous (parent) block and the transaction list stores the up to date transactions. The hash values are generated based on the contents of the data that has been stored thus far in the chain. By design, the chain is therefore immutable and any change to data in the system will result in a change in hash values, and thus it will be detected. The first block in the chain is called the genesis block and it has no parent. Each node in the network uses two keys: public key and private key to perform transactions. The public key is used as an address of a node and the private key is used to sign a transaction. Each signed transaction is broadcasted to all the peers who check the validity of the transaction. An invalid transaction is discarded and only a valid transaction is eventually reached to all the nodes in the network. At a fixed time interval, a special node, selected through a consensus, known as *miner*, collects all the transactions, orders them, and packs them into a block and broadcast in over the network. Upon successful verification of the block, it is included into the chain. There exist different consensus mechanisms for blockchain. Bitcoin and Ethereum use computationally very expensive Proof-of-Work (PoW) [27]. On the other hand, Proof-of-stake (PoS) is a scalable and lightweight alternative of PoW. Instead of computational energy, nodes' longevity and stake of cryptocurrencies are considered for miner selection, which yields far little required amount of power than PoW [27].

A. System Model and Important Assumptions

The proposed system model has two major entities: 1) a region, divided into a set of smaller cells, and 2) a set of sensor nodes. Some of the sensors are static and others are mobile. The mobile nodes are moving over the region based on Random Waypoint Mobility model [28]. Each sensor node is capable of performing lightweight aggregate operations, such as e.g. max, mean, min, weighted average [29] and so on. Furthermore, the proposed system does not require any additional resources. We assume that the distribution of the sensed data within a cell is approximately same. The proposed blockchain is a permissioned-blockchain. That is, the authority of the blockchain assigns each IoT Node a private key and a private key and to join a network a node needs to reveal its identity to all the other nodes in the network. In order to achieve conditional privacy from the peers, an IoT node can anytime request the authority for new key pairs. Devising mechanisms for Key management and authentication are beyond the scope of this paper. In the blockchain, the nodes use Proof-of-stake (PoS) consensus mechanism.

IV. SENSOR-CHAIN FRAMEWORK

This section presents the Sensor-Chain framework. We first discuss 3 different frameworks: Conventional and our proposed improved temporal, and spatial blockchains. We analyze their strength and limitations to highlight the motivation behind the design of Sensor-Chain framework.

A. Naive Approach: Conventional Blockchain

In the conventional blockchain frameworks [6], [14] a blockchain is managed by all the nodes in the network and continues to grow with the lifespan of the network. Thus, with a $T = \infty$ lifespan, according to our discussion in section I, the size of a conventional blockchain becomes,

$$size(\text{conventional}) = \sum_{t=1}^{\infty} \frac{n(n-1)}{2} \tag{1}$$

Obviously, this blockchain will impose a high storage requirement which cannot be met by sensor nodes. To improve this, we then design an improved version of temporal blockchain [26] in the context of mobile IoT.

B. Our 1^{st} Approach: Improved Temporal Blockchain

In the original temporal blockchain [26], it was proposed to keep a portion of the blockchain after certain time period. However, we propose to replace the blockchain with an aggregated version of it after certain a time period. In detail, in the preprocessing step of our scheme, we consider a specific time at the "genesis time", and a time period is set as the temporal constraint for blockchain deletion. For example, if 00:00 in 24-hour format is taken as the genesis time and the temporal constraint is 2 hours, then the deletion operation will take place at 02:00, 04:00, 06:00, ... of each day. This genesis time information and temporal constraint are preset onto the IoT devices. Another way to set this information is to have smart contract on the blockchain. We leave this for our future research. Every time the lifetime of the blockchain meets the temporal constraints, through the consensus mechanism,a node will be selected as an aggregator node which performs aggregation over the whole blockchain and creates an aggregated block. This block includes the ID of the aggregator node. This block is then broadcasted over the network by the aggregator. This aggregation could be anything lightweight for IoT sensor devices to perform (e.g. min, max, mean, weight average [29]).

Upon receiving this block, the nodes in the network replaces the whole existing blockchain with this block on their local storage. That is, it will considered as the genesis block of a new blockchain. Even though as a consequence the newly restarted blockchain's size becomes relatively small, we still need to look into the size of the blockchain between two consecutive restarts so as to ensure that it is withing the storage space capacity of the IoT sensor node. If the temporal constraint is T_{chain}, then in the the worst case scenario, the maximum size of the blockchain can be,

$$size(\text{improved-temporal}) = \sum_{t=1}^{t=T_{\text{chain}}} \frac{n(n-1)}{2} \tag{2}$$

Clearly this scheme outperforms the conventional blockchain schemes. However, with higher T_{chain} and a large number of nodes in a network, the nodes still need to hold a large blockchain, making it quite impractical for IoT devices. Thus, despite the fact that a temporal blockchain can reduce the size of a chain, the size of a chain must be further improved when dealing with IoT nodes. This is done using the following spatial blockchain technique.

C. Our 2^{nd} Approach: Spatial Blockchain

In our spatial blockchain framework, a global blockchain is broken down into smaller disjoint *local blockchains* with the aim of reducing the number of transactions performed by a node at any given time than in conventional blockchain frameworks. To achieve this objective, we translate a region into a Voronoi diagram [30]. Voronoi diagram $\mathcal{C}$, is a partitioning of a plane into non-overlapping smaller convex regions, called Voronoi cells C. Based on this partitioning of the plane, we define two different structures: local networks and local blockchains (figure 1 depicts these structures). A *local network*

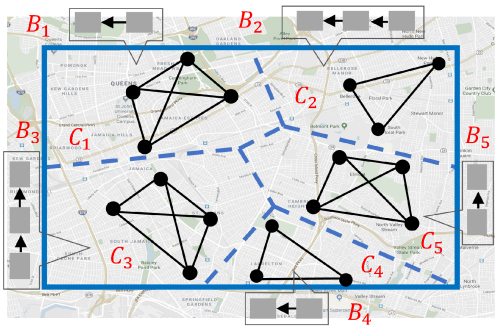

Fig. 1. Proposed blockchain-based IoT architecture: A region is transformed into a Voronoi diagram where C_i is the i-th Voronoi cell and the graph inside of it is a local network G_i. '•'s and'−'s represent nodes and edges between the nodes, respectively. B_i refers to the local blockchain in cell C_i.

refers to the graph $G_i^t = (V_i^t, E_i^t)$ formed by the nodes in the cell $C_i \in \mathcal{C}$ at time t. Here, V_i^t and E_i^t are the set of the nodes and the edges between them. Any two local networks of two different cells at the same time are disjoint. That is,

$$V_i^t \cap V_j^t = \emptyset, \quad E_i^t \cap E_j^t = \emptyset \tag{3}$$

A *local blockchain* B_i, is the blockchain managed by the nodes in cell C_i and B_i^t is the snapshot of B_i at time t. Any two local blockchains from two different cells have the following property: a block of a local blockchain in a cell is neither a parent nor a child of a block of another local blockchain in another cell at any time instance. That is,

$$\begin{aligned}(\exists b_i^x \in B_i | b_i^x \text{ is a parent of a block in } B_j) \cup \\ (\exists b_j^y \in B_j | b_j^y \text{ is a parent of a block in } B_i) = \emptyset; \forall t\end{aligned} \tag{4}$$

The two properties imply that a sensor node in G_i works only on local blockchain B_i. Hence, it needs to store only the copy of B_i at any given time as long as it remains in G_i. While this definitely improves the storage issue than in conventional blockchain, this scheme further enhance its efficacy by considering mobility of the nodes. In case of mobility, if a node moves from cell C_i to C_j, at first it deletes the copy of local blockchain B_i from its memory and then, after joining G_j, it downloads the copy of B_j from its peers. Thus, a node is required to store only one local blockchain at any time instance, which significantly reduces the required space to store a blockchain. We quantify the storage requirement of this scheme as follows. Let us consider that at any time instance, there could be at most m number of nodes in a cell, where $m < n$ and the time difference between the creation of genesis block and current time is $\approx \infty$. Let us also assume that a mobile node's permanence in a cell is at most T_{per}. At the first glance, it seems $size(\text{spatial}) = \sum_{t=1}^{t=T_{per}} \frac{m(m-1)}{2}$. However, consider the worst case scenario where there exists at least one node in a particular cell C_i all the time (if some nodes are static or the cell is never empty). That is, the local blockchain continues to expand forever. In that case,

$$size(\text{spatial}) = \sum_{t=1}^{t=\infty} \frac{m(m-1)}{2}; \quad m < n \tag{5}$$

From the analysis of temporal and spatial blockchains, it is not clear which one offers the best solution. For static nodes, the temporal blockchain with a small temporal constraint could be the better solution in the long run. On the other hand, in mobile environment, the spatial blockchain will be the winner. To address the limitations of both approaches, we propose Sensor-Chain approach.

D. Our Best Approach: Sensor-Chain

Sensor-Chain is a fusion of both temporal and spatial blockchain approaches. Similar to spatial blockchain, in this framework, a complete region is first divided into a number of Voronoi cells. Using those cells, the nodes in a cell form a local network and maintain a local blockchain. All the local networks and local blockchains follow the properties defined for spatial blockchain. Among different information, each nodes holds the following tuple: {*current cell id* C_{cur}, *copy of the local blockchain* B^t_{cur}}. In order to manage the size of a blockchain, this framework has two important constraints: temporal constraint T_{chain} and block creation time constraint T_{block}. The storage management of blockchain is done in two ways: spatiotemporal and mobility-based.

Spatiotemporal-based blockchain management is detailed in algorithm 1. In this framework, the block creation and insertion are done at a fixed time interval (lines 1-6), a similar approach of bitcoin. At first, in each local network G^t_i a $Miner$ is selected through consensus. Then the $Miner$ gathers all the recent transactions and creates $NewBlock$. Upon verification, the new block is inserted into B^t_i. The temporal constraint is used to reset the local blockchains at a fixed time interval. Every time the temporal constraint is met (line 8), an $Aggregator$ node is selected from each local network. This $Aggregator$ node computes aggregation of its local blockchain, creates an $AggregatedBlock$, and broadcasts it over its local network (lines 9-13). Upon receiving the $AggregatedBlock$, the nodes in the local network first delete their copy of the existing local blockchain (line 14) and then regenerate the local blockchain using the aggregated block as the genesis block (line 15).

Algorithm 2 presents the mobility-based blockchain management. Every time a node moves from one cell C_{cur} to another C_{new_cell} (line 1), it deletes the copy of the local blockchain B_{cur} of previous cell from its memory. Then it joins the The work flow of Sensor-Chain is illustrated in figure 2.

*1) **Analysis**:* We argue that, with such spatiotemporal and mobility-based blockchain management, Sensor-Chain provides the best solution. To prove its validity, we now analyze the space requirement to store a blockchain in this scheme. Referring to the discussion on spatial blockchain, with the space partitioning, the size of a local blockchain in Sensor-Chain can be at most,

$$size(\text{Sensor-Chain}) = \sum_{t=1}^{t=\infty} \frac{m(m-1)}{2} \tag{6}$$

However, as the temporal constraint T_{chain} is applied to all the local blockchains, according to the discussion on temporal

Algorithm 1: Spatiotemporal Blockchain Management

Input : Current time T_t, set of all local networks $\mathcal{G}$ at T_t, set of all local blockchains $\mathcal{B}^t$, genesis time T_{gen}, temporal constraint T_{chain}, block creation time constraint T_{block}

Output: Updated local blockchains $\mathcal{B}^t$

1 **if** $(T_{gen} - T_t)\%T_{block} == 0$ **then**
2 **for** *each* $G^t_i \in \mathcal{G}^t$ **do**
3 $Miner \leftarrow$ Select-Miner(V^t_i)
4 $NewBlock \leftarrow$ Create-Block$(Miner)$
5 Insert-Block$(B^t_i, NewBlock)$
6 **end**
7 **end**
8 **if** $(T_{gen} - T_t)\%T_{chain} == 0$ **then**
9 **for** *each* $G^t_i \in \mathcal{G}^t$ **do**
10 $Aggregator \leftarrow$ Select-Aggregator(V^t_i)
11 $AggregatedBlock \leftarrow$ Compute-Aggregation$(B^t_i, Aggregator)$
12 Broadcast$(AggregatedBlock)$
13 **for** *each node* $v \in V^t_i$ **do**
14 Delete(B^t_i) from local storage
15 $B^t_i \leftarrow$ Re-generate$(B^t_i, AggregatedBlock)$
16 **end**
17 **end**
18 **end**

Algorithm 2: Mobility-Based Blockchain Management

Input : Voronoi diagram $\mathcal{C}$, sensor node S

Output: Updated node S

1 **if** $S.C_{cur} \neq C_{new\_cell}$ **then**
2 Delete(B_{cur}) from local storage
3 $S.C_{cur} \leftarrow C_{new\_cell}$
4 Join(G^t_{cur})
5 Download(B^t_{cur}) from peers in V^t_{cur}
6 **end**

blockchain, the size of a local blockchain can be further reduced as follows,

$$size(\text{Sensor-Chain}) = \sum_{t=1}^{t=T_{chain}} \frac{m(m-1)}{2} \tag{7}$$

This analysis gives us the required storage space in Sensor-Chain. Next, we analyze the scheme case by case and draw comparison with our proposed improved temporal and spatial blockchain frameworks.

Case 1: In the first case, all the nodes are assumed as static. Also, the partitioning of the region is such that all the nodes reside in a single cell. In such a case, $m = n$.

$$size(\text{Sensor-Chain}) = size(\text{improved-temporal}) = \sum_{t=1}^{t=T_{chain}} \frac{n(n-1)}{2} < size(\text{spatial}) \tag{8}$$

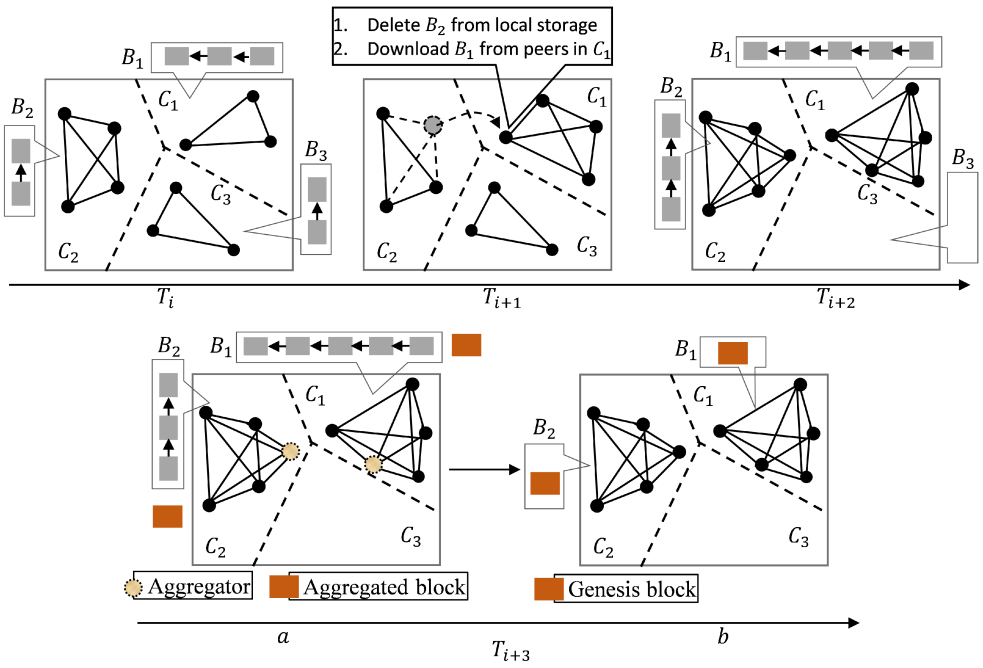

Fig. 2. Illustrated Sensor-Chain:-T_{i+1}: A mobile node moves from cell C_2 to C_1. First, it deletes the copy of local blockchain B_2 from its memory and then downloads B_1 from its peers in G_1^{i+1}. T_{i+2}: local blockchain B_3 does not exist anymore as C_3 is empty. T_{i+3}: as temporal constraint is met, (a) aggregator node from each local network is selected. The selected nodes compute aggregation over their respective local blockchains and generate aggregated blocks. (b) using the aggregated blocks as the genesis, the local blockchains are regenerated.

Where $size(\text{spatial}) = \sum_{t=1}^{t=\infty} \frac{n(n-1)}{2}$.

Case 2: All the nodes are moving in such a way that each local blockchain becomes empty (more correctly, it doesn't exist anymore) every time before the temporal constraint is satisfied. This case is depicted in figure 2(T_{i+2}) where cell C_3 is empty so that B_3 does not exist anymore. In such a case,

$$size(\text{Sensor-Chain}) = size(\text{spatial}) = \sum_{t=1}^{t<T_{chain}} \frac{m(m-1)}{2} < size(\text{improved-temporal}) \quad (9)$$

Where $m < n$ and $size(\text{improved-temporal}) = \sum_{t=1}^{t=T_{chain}} \frac{n(n-1)}{2}$.

All other cases: In all other cases,

$$(size(\text{Sensor-Chain}) < size(\text{spatial})) \; \& (size(\text{Sensor-Chain}) < size(\text{improved-temporal})) \quad (10)$$

V. Experimental Results

This section presents the experimental results. To carry out the experiment we use synthetic data. The parameters and their different values used in the experiment are presented in table II. We implemented all the four (conventional, improved-temporal, spatial, and Sensor-Chain) approaches. We ran the simulation for 6 hours and generated statistics for all the approaches. Specifically, we compared the approaches in terms of number transactions needed to be stored on a single IoT sensor device, as it defines the size of a blockchain. The evaluation is done from three different points of view: 1) duration of the simulation, 2) number of cells, and 3) number of sensors to analyze the benefit of Sensor-Chain in the long-run and scalability. The detail of the evaluation results are discussed below.

TABLE II
Parameters

Parameter	Values
Area of the region	$5000m \times 5000m$
Number of Voronoi cells	$50, 100, 150, 200, 1000$
Number of sensor nodes	$1000, 3000, 5000, 7000$
Speed of the nodes	$[0, 50]$ km/h
Temporal constraint T_{chain}	1 hour
Block creation time constraint T_{block}	10 minute

Figure 3(a) shows the result of the simulation for Sensor-Chain. In every hour, the curve moves upward. As $T_{chain} = 1$ hour, the size of the blockchain becomes 1 (with the aggregated block) at the end of each hour. It is also clear that in Sensor-Chain, using the temporal constraint, it is possible to keep the size of the blockchain within a limit. Figure 3(b) shows the comparison between Sensor-Chain and conventional approaches. From nearly the beginning of the simulation, the required storage space in Sensor-Chain is far less than in conventional approach. Next, we evaluate how Sensor-Chain, with the fusion of spatiotemporal and mobility-based blockchain management, outperforms the improved temporal and spatial schemes. For both of the improved temporal and Sensor-Chain, we used the same temporal constraint. Although the improved temporal blockchain shows a trend similar to Sensor-Chain, its required storage space is much higher than

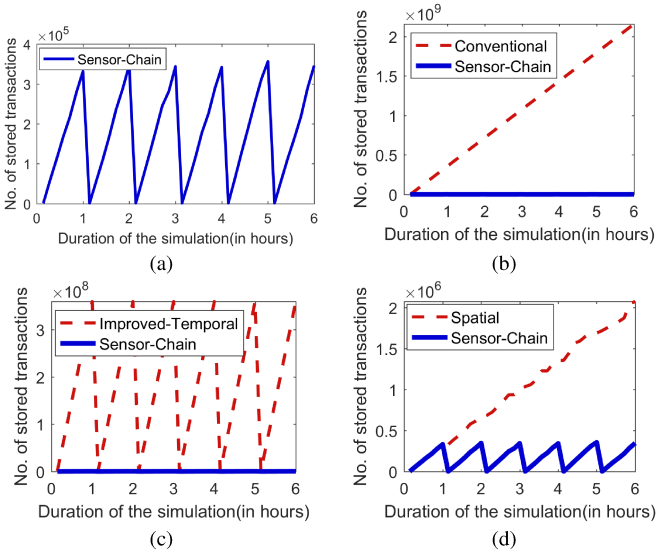

Fig. 3. Evaluation results in terms of performance in the long-run: (a) Sensor-Chain, (b) conventional, (c) improved-temporal, and (d) spatial blockchains (experiment Settings: number of cells = 50, number of sensors = 1000).

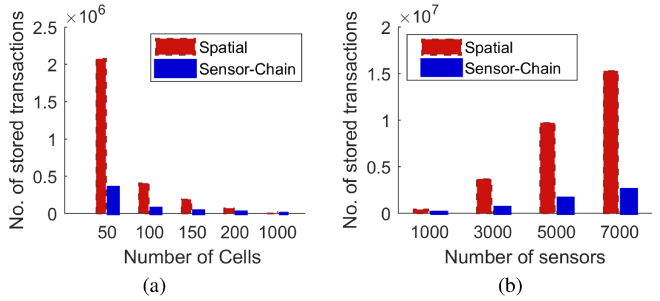

Fig. 4. Comparison between Sensor-Chain and spatial approaches in terms of number of (a) cells and (b) sensors.

Sensor-Chain. Figure 3(d) shows more interesting results on the comparison with spatial blockchain. In the 1^{st} hour, both spatial and Sensor-Chain approaches go toe-to-toe. However, just after the 1^{st} hour (as $T_{chain} = 1$ hour), the local blockchains in Sensor-Chain restore to genesis block, while spatial blockchain continues to grow over the time.

Then, we analyze the impact of number of cells and sensors on the size of the blockchain. As only spatial and Sensor-Chain use cell-based partitioning, here we analyze their comparison. Figure 4(a) presents the comparison result in terms of number of cells. It is understandable that with the increase in the number of cells, the size of a local blockchain decreases. Furthermore, it seems that when this number is relatively high (e.g. 1000 in the figure), both approaches require similar storage capacity. However, it is the number of sensors that makes the difference in such a particular case. With the increase in the number of sensors, the required storage space increases rapidly in spatial approach than in Sensor-Chain. Figure 4(b) shows the results for 1000 cells with different number of sensors.

VI. Conclusion and Future Work

In this paper, we proposed "Sensor-Chain", a lightweight scalable blockchain framework for resource-constrained IoT sensor devices. In this framework, a conventional blockchain is made lightweight in three steps. First, a global blockchain is divided into smaller disjoint local blockchains in spatial domain such that the required storage space to hold a local blockchain for an IoT device is always smaller than that in conventional blockchain. Second, a temporal constraint is imposed on the life span of the local blockchains to limit their size in temporal domain. Finally, a sensor node is required to keep at most one local blockchain in its memory at any time instance. We analyzed and tested Sensor-Chain in terms of both long-run performance and scalability; and compared with other approaches. Experimental results show that it consumes far little storage space than other approaches. As part of the future work, we are exploring ways to extend Sensor-Chain with an integration of smart contract. We are also working to devise a mechanism to deal with data loss when number of nodes in a cell become relatively low such that no trusted node is present to compute the aggregated block.

References

[1] K. Ashton *et al.*, "That internet of things thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *bitcoin.org*, 2008.

[3] P. Franco, *Understanding Bitcoin: Cryptography, engineering and economics*. John Wiley & Sons, 2014.

[4] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *Ieee Access*, vol. 4, pp. 2292–2303, 2016.

[5] B. Yu, J. Wright, S. Nepal, L. Zhu, J. Liu, and R. Ranjan, "Iotchain: Establishing trust in the internet of things ecosystem using blockchain," *IEEE Cloud Computing*, vol. 5, no. 4, pp. 12–23, Jul./Aug. 2018. [Online]. Available: doi.ieeecomputersociety.org/10.1109/MCC.2018.043221010

[6] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing: challenges and applications," *arXiv preprint arXiv:1711.05938*, 2017.

[7] J. Pan, J. Wang, A. Hester, I. Alqerm, Y. Liu, and Y. Zhao, "Edgechain: An edge-iot framework and prototype based on blockchain and smart contracts," *arXiv preprint arXiv:1806.06185*, 2018.

[8] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang, and Z. Han, "Cloud/fog computing resource management and pricing for blockchain networks," *IEEE Internet of Things Journal*, pp. 1–1, 2018.

[9] D. Wörner and T. von Bomhard, "When your sensor earns money: exchanging data for cash with bitcoin," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. ACM, 2014, pp. 295–298.

[10] O. Novo, "Blockchain meets iot: An architecture for scalable access management in iot," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1184–1195, April 2018.

[11] A. Z. Ourad, B. Belgacem, and K. Salah, "Using blockchain for iot access control and authentication management," in *Internet of Things – ICIOT 2018*. Cham: Springer International Publishing, 2018, pp. 150–164.

[12] G. Zyskind, O. Nathan *et al.*, "Decentralizing privacy: Using blockchain to protect personal data," in *Security and Privacy Workshops (SPW), 2015 IEEE*. IEEE, 2015, pp. 180–184.

[13] L. Axon, "Privacy-awareness in blockchain-based pki," 2015.

[14] A. Moinet, B. Darties, and J.-L. Baril, "Blockchain based trust & authentication for decentralized sensor networks," *arXiv preprint arXiv:1706.01730*, 2017.

[15] A. Durand, P. Gremaud, and J. Pasquier, "Decentralized web of trust and authentication for the internet of things," in *Proceedings of the Seventh International Conference on the Internet of Things*, ser. IoT '17. New York, NY, USA: ACM, 2017, pp. 27:1–27:2. [Online]. Available: http://doi.acm.org/10.1145/3131542.3140263

[16] G. Ayoade, V. Karande, L. Khan, and K. Hamlen, "Decentralized iot data management using blockchain and trusted execution environment," in *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, July 2018, pp. 15–22.

[17] P. Angin, M. B. Mert, O. Mete, A. Ramazanli, K. Sarica, and B. Gungoren, "A blockchain-based decentralized security architecture for iot," in *Internet of Things – ICIOT 2018*. Cham: Springer International Publishing, 2018, pp. 3–18.

[18] R. Casado-Vara, J. Prieto, and J. M. Corchado, "How blockchain could improve fraud detection in power distribution grid," in *International Joint Conference SOCO'18-CISIS'18-ICEUTE'18*. Cham: Springer International Publishing, 2019, pp. 67–76.

[19] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on*. IEEE, 2017, pp. 618–623.

[20] P. K. Sharma, S. Y. Moon, and J. H. Park, "Block-vn: A distributed blockchain based vehicular network architecture in smart city," *Journal of Information Processing Systems*, vol. 13, no. 1, p. 84, 2017.

[21] I. Foundation. (2018) Iota. [Online]. Available: https://www.iota.org/

[22] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Towards secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *arXiv preprint arXiv:1809.08387*, 2018.

[23] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things Journal*, 2018.

[24] Z. Xiong, S. Feng, D. Niyato, P. Wang, and Z. Han, "Edge computing resource management and pricing for mobile blockchain," *CoRR*, vol. abs/1710.01567, 2017.

[25] R. C. Lunardi, R. A. Michelin, C. V. Neu, and A. F. Zorzo, "Distributed access control on iot ledger-based architecture," in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2018, pp. 1–7.

[26] R. Dennis, G. Owenson, and B. Aziz, "A temporal blockchain: a formal analysis," in *Collaboration Technologies and Systems (CTS), 2016 International Conference on*. IEEE, 2016, pp. 430–437.

[27] C. Cachin and M. Vukolić, "Blockchains consensus protocols in the wild," *arXiv preprint arXiv:1707.01873*, 2017.

[28] E. Hyytiä and J. Virtamo, "Random waypoint mobility model in cellular networks," *Wireless Networks*, vol. 13, no. 2, pp. 177–188, 2007.

[29] S. Pumpichet, N. Pissinou, X. Jin, and D. Pan, "Belief-based cleaning in trajectory sensor streams," in *Communications (ICC), 2012 IEEE International Conference on*. IEEE, 2012, pp. 208–212.

[30] W. Alsalih, K. Islam, Y. N. Rodríguez, and H. Xiao, "Distributed voronoi diagram computation in wireless sensor networks." in *SPAA*, 2008, p. 364.