This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3153653, IEEE Internet of Things Journal

Sustainable Blockchain-based Digital Twin Management Architecture for IoT Devices

Chenyu Wang, *Graduate Student Member, IEEE*, Zhipeng Cai, *Senior Member, IEEE*, Yingshu Li, *Senior Member, IEEE*

***Abstract*—As the number of IoT devices increases, sustainability is becoming a bottleneck of the production process in industrial systems. As a matter of fact, inefficient management and scarce resources significantly impeded the development of sustainability. In recent years, it has been observed that the digital twin technology plays a promising role in facilitating the interaction between the Internet of Things (IoT) assets and digital services. However, high-fidelity models of digital twins raise the requirement of efficient data flows, which is limited by realistic constraints such as data collection strategy and energy supply. We propose a sustainable data collection and management approach to construct digital twins for physical assets. With this approach, data packets are uploaded to the data brokers, namely agents, by a large number of IoT devices. The challenge lies in the balance between enduring data collection and the information loss associated with the stale data. In this article, we aim to optimize the metrics of data fidelity and reveal delay while guaranteeing both sustainable energy and sustainable information. Additionally, a shareable and sustainable blockchain-based digital twin management architecture is proposed, which does not rely on data exchanges with a single centralized server. Our analytical and simulation results demonstrate the applicability of our proposed architecture.**

***Index Terms*—digital twin, sustainable system, internet of things, blockchain, network optimization**

I. Introduction

In the past decade, people have witnessed the expeditious development of the Internet of Things (IoT), which is cohesively integrated with Artificial Intelligence (AI), the new generation wireless network technology (5G), and advanced system architectures to serve human beings. With the proliferation of IoT devices, numerous digital service applications have been devised to meet the demands of industrial production and social activities, such as smart grid [1], smart city [2], smart transportation [3] and smart healthcare [4]. Our everyday lives are immersed with a large number of IoT devices, which build up the foundation of a variety of services. The purpose is to collect data in the physical world, upload data to remote servers for further processing, and make decisions according to feedback.

In the traditional approaches, data are stored in backlogs of IoT devices for later diagnosis, and improvements are then incorporated into devices. Such pipelines not only lack timely feedback, but also risk on-device testing and unverified updates, which may result in severe malfunctions. As creating a digital avatar for a real-world object is highly in demand in most IoT services, the Digital Twin (DT) technology becomes a promising paradigm for IoT services. Often referred to as a virtual representation of a physical asset, DT empowers complicated modeling and immense data transmission, thus creating high-fidelity replicas of physical objects for further prediction, monitoring, controlling, and decision making. These digital proxies are often expected to provide virtualization and optimization functionalities by integrating domain knowledge from subject-matter experts as well as real-time data collected from IoT devices [5]. DT also makes remote testing on virtual environment possible, which is a cost-efficient and secure alternative comparing with on-device testing. Due to the aforementioned high-fidelity and flexibility advantages, DT has been widely adopted in many applications, *i.e.*, human DT [6], DT city [7], and DT automation [8].

However, unfortunately, most previous DT-related works have been restricted to the control of a limited number of IoT devices [9], [10]. In the works for large-scale DT platform deployment [11]–[13], it is not realistic to simultaneously upload data all the time for all IoT devices. Since these IoT devices are deeply integrated with various services, stringent requirements in terms of fresh data and energy supply are essential to robustness of an IoT system. To overcome the above limitations, we should take sustainability of IoT systems into consideration. Sustainability considered in this article is twofold, which can be categorized by sustainable information and sustainable energy.

On the one hand, data generated by an IoT device may play different roles in different DT services. As a result, at a certain time, the status of an IoT device might be crucial to DT but not relevant to another device. Prioritizing different data sources carefully is essential for DT services to receive the most crucial data from all physical devices. One should devise a feasible strategy of data collection to maintain information sustainability and achieve better synchronization [14] in a global perspective.

On the other hand, energy supply is another key issue for the services aided by physical assets. To provide adequate energy for devices in a region, an electricity storage system may be deployed where electricity is sustainably replenished and transported to IoT devices. The uninterrupted computation and data transmission of devices are enabled by such sustainable

This work is partly supported by the National Science Foundation (NSF) under grant NOs. 1912753, 1704287, 2011845 and 1829674.

C. Wang, Z. Cai and Y. Li are with the Department of Computer Science, Georgia State University, Atlanta, GA, 30303, USA (E-mail: cwang50@student.gsu.edu, zcai@gsu.edu, yili@gsu.edu) (*Corresponding author: Yingshu Li.*)

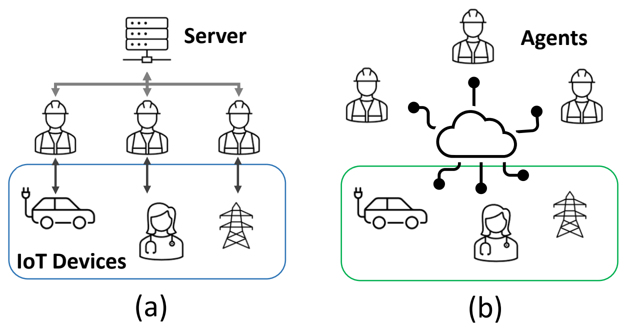

Fig. 1: Hierarchical and distributed data-sharing modes.

energy supply. However, frequent data updates at IoT devices may deplete energy quickly, thus hampering energy sustainability. Undoubtedly, a reasonable energy allocation strategy to provide energy sustainability is desired.

A common way to realize DT services is to establish constant connections and data exchanges between IoT devices and service provider, and to upload data to a centralized server similar to traditional IoT services. This hierarchical sensing mode/protocol [15]–[17] has been broadly studied and applied to many practical scenarios of dedicated services, which is illustrated in Fig.1a. However, this way of data collection is deemed to be unsustainable, since each service collects the needed data by itself with limited resources, and the barriers among untrustworthy services would cause unnecessary data collection and incomplete information. Data trading mechanisms [18], [19] are studied to expand the coverage of services while preserving the privacy requirements. Motivated by the deficiency of traditional services, we raise the question of whether the collected data can be shared among all the DT systems as shown in Fig.1b and the queries toward massive IoT data can be processed in a distributed manner [20]. In order to ensure service sustainability, we consider maintaining a trustable and shareable ledger, colloquially known as blockchain, for all the DT systems. The blockchain gathers distributed, secured and verifiable records of information collected from different entities, and links them in a single chain with multiple blocks [21]–[23]. The information is maintained by all the participants, and can be resistant to the failure of a single point, which leads to a secure and convergent industrial IoT environment [24]. As the core of blockchain, we should devise some consensus to make sure that data can be utilized by different DT systems securely and fairly.

In this article, we design a blockchain-based sustainable DT management system to ensure information sustainability in DT systems and energy sustainability in physical assets. The main contributions are summarized as follows:

- A DT framework, consisting of devices, agents, and requestors, is formalized in the scenario of IoT device-assisted services. In this framework, agents collect fresh data from physical devices, and feed them to requestors so as to create DT services for further uses.
- Information sustainability and energy sustainability are both considered to improve system performance. In details, we optimize delay of devices to ensure sustainable information and control the probability of energy depletion to guarantee data fidelity.
- To further enhance practicability and system performance, we introduce the blockchain technology to enable data sharing among agents, and improve the efficiency of data collection while not relying on data exchanges with a specific server.
- Our extensive analytical and experimental results show that our proposed blockchain-based DT management system can achieve both information sustainability and energy sustainability.

The remainder of this paper is organized as follows. Section II reviews some related works. In Section III, we introduce the components of our DT-based IoT service system. In Section IV, we show the optimization goal and collection method of DT system with the aid of a specific server. In Section V, we further propose a blockchain-based system that incorporates distributed and shareable attributes into our design. The experiment results are illustrated in Section VI. Finally, Section VII concludes the paper.

II. Related Works

A. Digital Twin Platform

The DT concept was first introduced in 2002 and has recently been implemented to solve different problems in the areas of aviation, supply chain, wireless networks, and many more. In [25], the authors introduced the DT concept, categorized the types of DT systems, and provided a framework for data exchanges between the physical twin and its DT. They also presented a vast of applications where system performance can be enhanced through the DT technology. In [26], the authors solved the mobile offloading problem in 6G networks with the assistance of DT, where DTs of edge servers and a mobile edge computing system are deployed to estimate servers' states and provide training data for offloading decisions. In [27], a DT bending bean test system is established by setting up DT components, a physical twin of two actuators, and a communication interface that connects the two. These works only considered the construction of DT with one or several physical twins and services, which cannot depict the status of many assets. Driven by this limitation, some researchers studied DT platforms for large-scale systems [28]–[31]. However, system sustainability has not been well investigated in the state-of-the-art.

B. Information Cost and Energy Cost

To model information sustainability, *Age of Information* (AoI) is an ideal performance metric that measures the loss of information at the destination [32]. Often defined as the elapsed time since the generation of the most recently received data, AoI characterizes the freshness of data for a service and suggests the potential utility that can be extracted from the data. However, AoI, determined by the information collecting strategy, cannot reflect the fundamental updating frequency of information source.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3153653, IEEE Internet of Things Journal

For energy cost, many works have studi
tion in extensive network paradigms. Hov
mation sustainability nor energy sustaina
addressed for IoT services with the aid
optimization problem of an IoT monitori
to minimize the average AoI while sat
energy cost constraint at the devices in [
sustainability of a multi-node wireless p
tion network, both energy and price-inc
studied aiming at optimizing the per-pack
[34]. The work in [35] investigates the c
for energy-harvesting monitoring nodes s
average AoI. Inspired by these prior stud
mize system performance in terms of sus
and sustainable energy for DT-aided IoT

C. Blockchain-based Digital Twin

Both DT and blockchain technologies
posed in the first decade of this cent
not received much attentions until rec
[36] presents a literature-review study for
and design of blockchain-based DT. Leveraging the strength of trustworthy blockchain, the data-driven product lifecycle events of physical assets can be efficiently utilized by multiple DT-based services. The study in [37] proposes a decentralized ownership-centric sharing model for protecting access control integrity and confidentiality based on DT components and lifecycle requirements. Besides, this work [38] proposes a DT model for additive manufacturing in the aircraft industry with the aid of blockchain. To collect the data from the physical devices, crowdsourcing has been widely adopted in blockchain-based IoT platform [39], [40], where agents are deployed to collect the data and contribute to the IoT services. However, none of them consider sustainability of the blockchain-based DT platforms.

III. System Model

We consider a system that supports the construction of DT services by collecting data from physical assets. Our proposed system consists of multiple participants in a given region, including *requestors*, *agents*, and *devices*, all of which play important roles in the process of data exchange and decision making. The detailed definitions of these participants are summarized as follows.

1) **Devices**: As physical assets, IoT devices are responsible for generating data for DT services. Each time an IoT device generates data, it consumes energy supplied by a central electricity storage system. In general, devices act as the foundations and actuators of services.
2) **Agents**: Due to the limited capacities of computation, communication and data storage, devices cannot be always online/connected for a single server. In a region, agents[1] are deployed to initiate communications to selected devices and collect data from them so as to support DT services. This means agents could strategically interact with physical devices to maintain service quality. In general, agents perform as data brokers which bridge DTs and physical assets and realize the expected DT services.
3) **Requestors**: As customer-oriented service providers, requestors subscribe to agents and incorporate their data in the DTs of physical assets. Data may be collected from a broad range of IoT devices with different functionalities (*e.g.*, speed of cars, quantity of fuel and air humidity). The DT services established by requestors can be used to serve target customers.

[1]The terminology of service-oriented agents may have different names in some other scenarios such as mobile workers or data sellers.

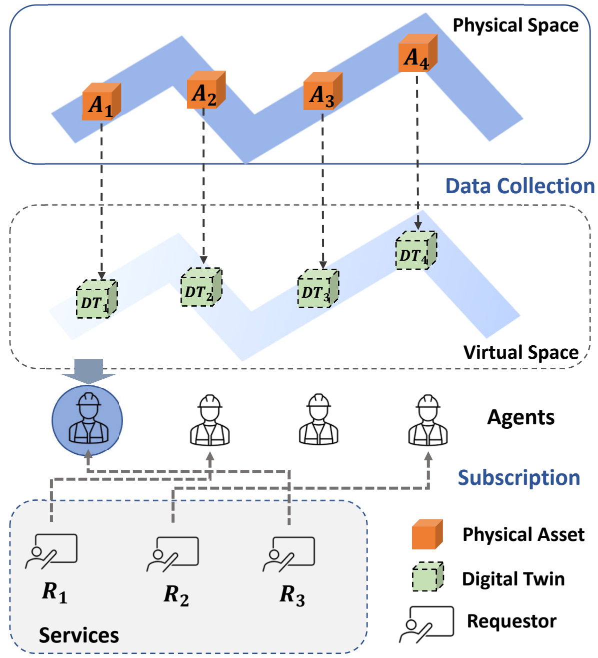

Fig. 2: DT-assisted paradigm for IoT devices.

A. Sustainable DT-based Paradigm

To maintain a sustainable system that provides high-quality and durable DT services, we deploy a set of IoT devices denoted by $\{1, 2, \cdots, N\}$, and a set of agents denoted by $\{1, 2, \cdots, M\}$. We simplify the requestors subscribing to agent m by a popularity vector $\boldsymbol{\alpha^m} = \{\alpha_1^m, \cdots, \alpha_N^m\}$. This representation reflects the data demand priorities for agent m. The popularity vectors of different agents may be significantly different as different agents may concentrate on different types of services, as a result, leading to distinctive data collection strategies of individual agents. We propose a DT-based framework that benefits the DT services, which is able to retrieve the last known status of the devices. The details about similar functionalities are available in [5]. We assume that the system can be divided into three components:

1) **Physical devices** which are deployed in a physical space in a distributed manner to ensure the basic system functionalities such as data sensing.
2) **Agents** which exert measurements on one of the devices at a time independently. DTs are created in a virtual

space (constructed by agents) to synchronize the status of physical devices.

3) **Requestors** who subscribe to the models of DTs maintained by agents, and run DT-based services to serve customers. Each subscription is made through one of the agents.

It is critical to synchronize the real-time status of physical twins. As illustrated in Fig.2, the mapping process between the physical space and virtual space is enabled through agents. In our system, we assume that each IoT device $n \in \{1, 2, \cdots, N\}$ performs the sensing task and generates a data log of the current state independently under a Bernoulli distribution with rate p_n in each time slot. Thus, the total number of data logs generated from the sensing tasks of device n in T time slots, *i.e.*, J_n, follows the following distribution,

$$\mathcal{P}\left(J_n = j, T, p_n\right) = \left(\begin{array}{c} T \\ j \end{array}\right) {p_n}^j \left(1 - p_n\right)^{T-j}. \tag{1}$$

With the statistic of sensing tasks, agents perform data collection according to their own popularity profiles of all the IoT devices. The popularity profiles are determined by the subscriptions made by requestors in the given range of time. Due to the trend of sustainability, we mainly focus on the optimization in terms of timeliness of data and endurable energy supply for DT systems.

B. Sustainable Energy Supply of Devices

We first formalize the sustainability of energy supply. We consider that all IoT devices are supported by a central electricity storage system. To describe our proposed system, the time horizon with stable energy supply is divided into multiple stages $\{1, 2, \cdots, K\}$ with T time slots. Note that T is a relatively large number. We define that the total amount of energy supply is S_c unit in each stage $k \in \{1, 2, \cdots, K\}$, and the energy replenishment at the beginning of any stage constantly supports the basic functionalities of all physical devices, thus ensuring the status mapping in the DTs of agents.

Traditional DT formulation typically concentrates on the status mapping between a single DT and its corresponding physical asset, thus neglecting the impact of energy supply for a large group of devices. We extend traditional DT cases to industrial DT platforms with a large-scale deployment of IoT devices (*i.e.*, a large N) and study the data sensing policy of devices in the system.

For any device ($\forall n \in \{1, 2, \cdots, N\}$), it consumes a unit energy each time to perform a sensing task and collect the status changes of the physical world. In any stage k, the energy consumed by device n is denoted by $a_n(k) = J_n(k)$, where $J_n(k)$ is the total times of sensing of device n in stage k. The distribution of J_n, denoted as $\mathcal{P}(J_n, T, p_n)$, tends to be the Poisson distribution with expectation $\lambda_n \to p_n T$ when $T \to \infty$ and $p_n \to 0$ [41]. In fact, λ_n can be treated as the expected energy consumed by device n in any stage with T time slots. Thus, the energy consumption process of all devices can be depicted by

$$W(k) = A(k) - kS_c, \tag{2}$$

where $A(k)$ is the accumulated energy consumed by all devices by the end of stage k, written as

$$A(k) = a^{(1)}(k) + a^{(2)}(k) + \cdots + a^{(N)}(k), \tag{3}$$

and $\forall n \in \{1, 2, \cdots, N\}$,

$$a^{(n)}(k) = a_n(1) + a_n(2) \cdots + a_n(k). \tag{4}$$

To characterize system properties with respect to the numbers of devices N, we denote $S_c = NC$, where $C > 0$ reflects the relationship between the energy supply rate and the number of devices. Thus, the maximum energy debt with respect to the energy consumption process of devices can be depicted by

$$Q = \sup_{k \geq 0} W(k). \tag{5}$$

We consider that the central electricity storage system stores a maximum energy backup with $B(0)$ units[2] in stage $k = 0$ to overcome the potential energy shortage in the future. It can be easily found that if Q is greater than B, the system will encounter energy depletion, which may cause severe operation problems, and devices would not be able to work correctly and provide the expected sensing data for DT services.

Large deviation theory [42] is a useful tool for analyzing rare or tail events with larger fluctuation, especially for energy depletion in a large-scale system with a large number of devices. We aim to design a robust system that yields sustainable energy supply for all devices.

Denote the probability of energy depletion as $\mathcal{P}(Q > B)$. As the number of sensing tasks in any stage could be approximated by a Poisson distribution, the expression of the cumulant generating function (CGF) of average energy consumption $a(k) = A(k)/N$ is given by

$$\Lambda_k(\theta) = \log \mathbb{E}\left[\mathrm{e}^{\theta a(k)}\right], \tag{6}$$

and

$$\begin{aligned} \Lambda(\theta) &= \lim_{k \to \infty} \frac{1}{k} \Lambda_k(\theta) \\ &= \frac{1}{N} \log \mathbb{E}\left[\mathrm{e}^{\theta \sum_{n=1}^{n} J_n}\right] \\ &= \frac{1}{N} \sum_{n=1}^{N} \lambda_n(\mathrm{e}^{\theta} - 1). \end{aligned} \tag{7}$$

To achieve sustainable energy, we focus on the behavior of $\mathcal{P}(Q > B)$. Since we study the system with a large N, for any $B = Nq > 0$, Q follows

$$\lim_{N \to \infty} \frac{1}{N} \log \mathcal{P}\left(Q/N > q\right) \approx -\mathcal{I}(q) \tag{8}$$

according to the Cramér's theorem [43], where $\mathcal{I}$ is the rate function of q that describes the probability decays of energy depletion with respect to the energy backup, and

$$\mathcal{I}(q) = \inf_{k \in \mathbb{N}} \Lambda_k^*\left(q + Ck\right), \tag{9}$$

where Λ_k^* is the convex conjugate of $\Lambda_k(\theta)$ defined by

$$\Lambda_k^*(x) = \sup_{\theta \in \mathbb{R}^+} \left\{\theta x - \Lambda_k\left(\theta\right)\right\}. \tag{10}$$

[2]We omit the time stage index 0 of $B_n(0)$ for simplicity.

As shown in [44], rate function $\mathcal{I}(q)$ can be written as

$$\begin{aligned}\mathcal{I}(q) &= \inf_{k\in\mathbb{N}} \sup_{\theta\in\mathbb{R}^+} \theta(q+Ck) - k\Lambda(\theta) \\ &= q \sup\left\{\theta > 0 : \Lambda(\theta) \le \theta C\right\},\end{aligned} \tag{11}$$

and we could define δ as a function of the process of $a(k)$, which is

$$\delta\left(a\right) = \sup\left\{\theta : \Lambda(\theta) \le \theta C\right\}. \tag{12}$$

Hence, the probability of energy depletion of the central electricity storage system is approximated by [45]

$$\mathcal{P}\left(Q > B\right) \approx \mathrm{e}^{-\delta N q} \tag{13}$$

according to the large deviation theory. This approximation is of great importance to guide the sensing process so that the usage of energy in our system is controllable and sustainable.

C. Sustainable Information of Agents

To capture the time-varied information of the system and inject the data of physical assets into their DTs, agents exert data collection from the IoT devices.

In any stage with T time slots, each IoT device $n \in \{1, 2, \cdots, N\}$ performs the sensing task and generates logs of data sensing independently at time slots $\{s_1^n, s_2^n, \cdots\}$. To pursue a high-fidelity digital model of the physical assets, we expect to collect the logs as soon as possible. We assume that the time slots of data collection at device n performed by all agents, *i.e.*, $\{c_1^n, c_2^n, \cdots\}$, are randomly distributed among the time slots of sensing tasks. To capture the utility of collected data, we give the definition of *data fidelity*.

Definition 1 (Data Fidelity)**.** *The data fidelity of a given log in a DT is characterized by the expected time spanning from the most recent time of data generation to the delivery of data.*

For any time of data collection at device n, denoted as c_i^n, it splits the interval of any successive sensing tasks i' and $i'+1$ into two intervals $(s_{i'}^n, c_i^n]$ and $\left(c_i^n, s_{i'+1}^n\right]$. If data collections are randomly distributed between the two successive sensing tasks, data fidelity $\mathcal{F}_n$ of device n can be obtained [46], which is

$$\mathcal{F}_n = -\mathbb{E}\left[c_i^n - s_{i'}^n\right] = -\mathbb{E}\left[s_{i'+1}^n - c_i^n\right] = -\frac{T}{2\lambda_n}. \tag{14}$$

To ensure high quality of DT models, devices should perform adequate sensing tasks to fully capture the status changes, thus improving data fidelity. However, this goal is limited by the total energy supply.

To obtain sustainable information from the data records of agents, a system devotes to the optimization of the overall data fidelity of all devices and guarantees sufficient data collection for each device, which is closely related to the status of the current data records and the sensing statistics of devices. We will detail the optimization process in Section IV.

IV. Collection Methods for DT Services

In this section, we present a theoretical analysis for the multiple-agent data collection scheme with the coordination of a centralized server. Initially, the central electricity storage system keeps the maximum energy backup of $B = Nq$ units, and at the start of any stage k, each agent $m \in \{1, 2, \cdots, M\}$ directly uploads its individual popularity vector $\{\alpha_n^m(k)\}_{n=1}^N$ of all devices to the centralized server. The centralized server can set the sensing policy of all devices at the beginning of stage k. With the sensing policy of all devices, agents perform data collection to optimize the quality of collected data according to the popularity profiles. To ensure the sustainability of both energy supply and information, we will introduce the methods respectively.

A. Optimal Sensing Policy of Devices

Considering sensing policy $\boldsymbol{\lambda} = \{\lambda_n\}_{n=1}^N$, (12) can be written as a function $\boldsymbol{\lambda}$, *i.e.*, $\delta(\boldsymbol{\lambda})$. If the parameters of energy supply rate C and energy backup q of the central electricity storage system are given, our goal is to explore the feasibility of sensing policies of all devices. We explain the definition of *feasible sensing policy* as following.

Definition 2 (Feasible Sensing Policy)**.** *For given parameters of energy supply rate C and energy backup q of the central electricity storage system, a feasible sensing policy of all devices is bounded by the maximum average energy consumption rate of sensing tasks, which ensures that the depletion probability of the central electricity storage system is no greater than the depletion tolerance degree $\epsilon \in (0, 1]$.*

For simplicity, we denote κ as $\frac{1}{N}\sum_{n=1}^N \lambda_n$. According to this definition, the maximum average energy consumption rate of all feasible sensing policies, denoted as $\hat{\kappa}$, is given by

$$\hat{\kappa}(\epsilon) = \max\left\{\kappa : \mathcal{P}(Q/N \ge q) \le \epsilon\right\}, \tag{15}$$

and $\hat{\kappa}(\epsilon)$ can be derived from the following theorem.

Theorem 1. *Given $\kappa < C$ for stability, $q > 0$ and $\epsilon \in (0, 1]$, the maximum energy consumption rate of all feasible sensing policies, denoted as $\hat{\kappa}(\epsilon)$, follows*

$$\hat{\kappa}\left(\epsilon\right) = \frac{\theta^* C}{\mathrm{e}^{\theta^*} - 1}, \tag{16}$$

where $\theta^* = -\frac{\ln\epsilon}{Nq}$.

Proof. With (12), (13), and (15), we have

$$\hat{\kappa}(\epsilon) = \max\left\{\kappa : \mathrm{e}^{-\delta(\kappa)Nq} \le \epsilon\right\}. \tag{17}$$

According to (12), θ should satisfy the condition of

$$\kappa \le \frac{\theta C}{\mathrm{e}^{\theta} - 1}, \tag{18}$$

where $\frac{\theta C}{\mathrm{e}^{\theta}-1}$ is always decreasing with the increment of θ when $\theta > 0$. From (17), we could easily find that

$$\delta\left(\kappa\right) \ge -\frac{\ln\epsilon}{Nq}. \tag{19}$$

This means that $\theta \ge -\frac{\ln\epsilon}{Nq}$, and the maximum κ is obtained for $\theta^* = -\frac{\ln\epsilon}{Nq}$, and $\hat{\kappa}(\epsilon) = \frac{\theta^* C}{\mathrm{e}^{\theta^*}-1}$. ☐

With Theorem 1, we know that the energy supply of the system significantly impacts the data fidelity of DT models. We aim to derive an energy policy that can maximize the weighted expected data fidelity while guaranteeing the stability of the system for $q > 0$.

To reduce the overall information loss, the goal of the system is to adapt the rate of sensing tasks to maximize the data fidelity of all devices when considering both energy supply and popularity of devices, which yields the optimization problem $\mathbb{P}1$ given as follows:

$$\mathbb{P}1: \quad \min_{\boldsymbol{\lambda}} \quad -\sum_{m=1}^{M}\sum_{n=1}^{N} \alpha_n^m \mathcal{F}_n(\lambda_n) \tag{20a}$$

$$\text{s.t.} \quad \kappa(\boldsymbol{\lambda}) \leq \hat{\kappa}(\epsilon) \tag{20b}$$

In $\mathbb{P}1$, the optimization goal (20a) is to minimize the negative weighted-sum data fidelity of all devices for the given popularity profiles of all agents, and the constraint (20b) infers that the average energy consumed during the sensing tasks cannot exceed the maximum average energy consumption derived from Theorem 1, which ensures that the probability of energy depletion would not exceed ϵ. The unique optimal solution for $\mathbb{P}1$ could be derived according to Theorem 2.

Theorem 2. *The optimal sensing strategy of all devices $\boldsymbol{\lambda}^*$ from Algorithm 1 is the unique solution to $\mathbb{P}1$.*

Proof. Let $\beta_n = \sum_{m=1}^{M} \alpha_n^m, \forall n$. The optimal sensing strategy of $\mathbb{P}1$ could be solved by constructing the Lagrangian function by introducing the multiplier $\gamma \geq 0$ associated with the energy constraint, that is

$$\mathcal{L}^{\mathbb{P}1}(\boldsymbol{\lambda}, \gamma) = \sum_{n=1}^{N} \frac{\beta_n T}{2\lambda_n} + \gamma \left(\sum_{n=1}^{N} \lambda_n - \hat{\kappa}(\epsilon) \right). \tag{21}$$

The optimal solution $\boldsymbol{\lambda}$ to $\mathbb{P}1$ should satisfy the following KKT conditions:

1) $\partial \mathcal{L}^{\mathbb{P}1} / \partial \boldsymbol{\lambda} \in \mathbf{0}$ (stationarity);
2) $\gamma \left(\sum_{n=1}^{N} \lambda_n - \hat{\kappa}(\epsilon) \right) = 0$ (complementary slackness);
3) $\sum_{n=1}^{N} \lambda_n - \hat{\kappa}(\epsilon) \leq 0$ (primal feasibility);
4) $\gamma \geq 0$ (dual feasibility).

From the KKT conditions, the derivative of stationarity condition yields that

$$\frac{\partial \mathcal{L}^{\mathbb{P}1}}{\partial \lambda_n} = -\frac{\beta_n T}{2\lambda_n^2} + \gamma = 0, \ \forall n, \tag{22}$$

and the optimal non-negative γ^* and $\boldsymbol{\lambda}^*$ should satisfy the following conditions

$$\begin{cases} \gamma^* = 0, \ \text{if} \ \sum_{n=1}^{N} \lambda_n^* - \hat{\kappa}(\epsilon) < 0, \\ \sum_{n=1}^{N} \lambda_n^* - \hat{\kappa}(\epsilon) = 0, \ \text{if} \ \gamma^* > 0, \end{cases} \tag{23}$$

along with the complementary slackness and feasibility conditions.

When $\gamma = 0$, (22) cannot be satisfied since $\lambda_n \ll T$. Consequently, we have the unique solution

$$\lambda_n(\gamma^*) = \sqrt{\frac{\beta_n T}{2\gamma^*}}, \ \gamma^* > 0, \tag{24}$$

and

$$\sum_{n=1}^{N} \lambda_n(\gamma^*) = \hat{\kappa}(\epsilon). \tag{25}$$

Since $\sum_{n=1}^{N} \sqrt{\frac{\beta_n T}{2\gamma}}$ is monotonically increasing with the decrements of γ, we can first set γ as a relatively large value and check whether Equation (25) holds. If not, the optimal unique solution of $\boldsymbol{\lambda}^*$ can be derived from (24) by gradually updating $\gamma > 0$ using the sub-gradient method until (25) approximately holds. We summarize the process of finding the optimal unique solution $(\boldsymbol{\lambda}^*, \gamma^*)$ of $\mathbb{P}1$ in Algorithm 1. □

Algorithm 1 Finding unique optimal sensing policy $(\boldsymbol{\lambda}^*, \gamma^*)$ of $\mathbb{P}1$ with KKT conditions.

1: Set η as a real number close to 0
2: Set $\gamma^{(0)}$ as a relatively large positive number
3: $\lambda_n^{(0)} \leftarrow \sqrt{\frac{\beta_n T}{2\gamma^{(0)}}}, \ \forall n$
4: **repeat**
5: $\quad \gamma^{(l+1)} \leftarrow \gamma^{(l)} + \eta \left(\sum_{n=1}^{N} \lambda_n^{(l)} - \hat{\kappa}(\epsilon) \right)$
6: $\quad \lambda_n^{(l+1)} \leftarrow \sqrt{\frac{\beta_n T}{2\gamma^{(l+1)}}}$
7: **until** $\gamma^{(l+1)}$ converges
8: **return** $\left(\boldsymbol{\lambda}^{(l+1)}, \gamma^{(l+1)} \right)$

The solution of $\boldsymbol{\lambda}$ shows the statistics of sensing policy of all devices, which is calculated using the public information such as the parameters of energy supply rate C, maximum energy backup q, depletion tolerance degree ϵ, and the popularity vectors $\{\boldsymbol{\alpha}^m(k)\}_{m=1}^{M}$. The deterministic property of $\boldsymbol{\lambda}$ provides an view of data collection for agents, which is introduced in Section IV-B.

B. Optimal Collection Strategy of Agents

In a centralized system, the agents only concentrate on their own benefits from the data collection. In stage k, the optimal sensing policies $\{\lambda_n^*(k)\}_{n=1}^{N}$ of all devices are deterministic when popularity vectors $\{\boldsymbol{\alpha}^{\boldsymbol{m}}(k)\}_{m=1}^{M}$ of all agents and other system settings are given. Each agent should perform data collection according to their own data records. Besides the sensing policies of devices, data collection strategies of agents also affect the quality of a DT model. To reveal the logs of sensing tasks timely, we give the definition of *reveal delay*.

Definition 3 (Reveal Delay)**.** *The reveal delay of a time slot is characterized by the time spanning from the generation time of the most recent collected log to this time slot at any agent.*

For any time of data collection c_i^n at device n, it splits the interval of any successive sensing tasks i' and $i'+1$ into two intervals $(s_{i'}^n, c_i^n]$ and $\left(c_i^n, s_{i'+1}^n\right]$. If the data collection is randomly distributed between any successive sensing tasks, the expected reveal delay $\mathcal{D}_n$ in time slot c_i^n can be obtained, which is

$$\mathbb{E}[\mathcal{D}_n(c_i^n)] = \mathbb{E}\left[c_i^n - s_{i'}^n\right] = \mathbb{E}\left[s_{i'+1}^n - c_i^n\right] = \frac{T}{2\lambda_n}. \tag{26}$$

We could easily find that (26) is similar to (14), and the length of each time interval (c_i^n, c_{i+1}^n) with a mean of $\frac{T}{u_n}$ will have an impact on the overall reveal delay of the DT system, where u_n is the expected times of data collection[3] at device n.

The best method for reducing the expected reveal delay of each time slot is to increase the number of data collections at a device. To capture the reveal delay in each time slot more precisely, we denote $d_n(t) \in \{0,1\}$ as the transmission indicator that shows whether a data collection is performed in time slot t ($d_n(t) = 1, \forall t \in \{c_1^n, c_2^n, \cdots\}$). The evolution of reveal delay regarding device n in any time slot is

$$\mathcal{D}_n(t+1) = \begin{cases} t - \max\{s_i^n | s_i^n \leq t\} + 1, & \text{if } d_n(t) = 1, \\ \mathcal{D}_n(t) + 1, & o.w.. \end{cases} \tag{27}$$

This expression shows that if the log of device n is collected by an agent in time slot t, the reveal delay of this device decreases if the log of a new sensing task is generated since the last data log has been captured; otherwise, it increases by one. To minimize the reveal delay of a DT system, the optimization problem of any agent can be characterized by

$$\mathbb{P}2: \quad \min_{\boldsymbol{u}} \quad \frac{1}{T}\sum_{t=1}^{T}\sum_{n=1}^{N} \alpha_n \mathcal{D}_n(u_n, t), \tag{28a}$$

$$\text{s.t.} \quad u_n \geq \lambda_n^*, \ \forall n. \tag{28b}$$

In $\mathbb{P}2$, the optimization goal (28a) aims to minimize the weighted-sum reveal delay of all devices in a stage, and (28b) shows the collection throughput constraints for all devices, where the expected collection times should be greater than the expected number of sensing tasks in any stage. In other words, to fully capture the log of sensing tasks, the expected collection interval $\frac{T}{u_n}$ should be set no greater than the expected information change interval $\frac{T}{\lambda_n}$.

$\mathbb{P}2$ imposes the requirement on the data log throughput. The optimization of AoI with the constraint of throughput has been studied in [32], and a Max-Weight-AoI (MWA) policy was used to adapt the data collection selection of devices to minimize the weighted-sum AoI of all nodes while guaranteeing the throughput.[4] However, with a fixed arrival delay, the AoI of a data packet in their formulation is completely determined by the information collecting strategy of the agent, which cannot reflect the fundamental updating frequency of information source, *i.e.*, the data generation process of sensing tasks.

To minimize the reveal delay, we next propose several properties of the reveal delay:

Lemma 1. *For any device n, the lower bound of the expected reveal delay is $\frac{T}{2\lambda_n}$, which is obtained when the data collection at this device is performed in every time slot.*

Proof. Since data collection is performed in every time slot, the reveal delay of each slot between any successive sensing tasks, *i.e.*, $[s_i^n, s_{i+1}^n - 1]$, is $\{1, 2, \cdots, s_{i+1}^n - s_i^n\}$, which yields an average value of $(s_{i+1}^n - s_i^n + 1)/2$. Consequently, we have the expected reveal delay of device n, which is

$$\mathbb{E}[\mathcal{D}_n] = \mathbb{E}\left[\frac{s_{i+1}^n - s_i^n + 1}{2}\right] = \frac{T}{2\lambda_n}. \tag{29}$$

□

Lemma 2. *For any device n, when the expected collection time interval $\frac{T}{u_n}$ is no more than the expected information change interval $\frac{T}{\lambda_n}$, the expected reveal delay is no more than $\frac{3T}{2\lambda_n}$.*

Proof. For any device n, when the expected collection time interval is set below the expected information change interval, the expected collection times within any successive sensing tasks should be no less than one. We assume that the only collection at device n between sensing tasks of s_{i-1}^n and s_i^n, *i.e.*, $[s_{i-1}^n + 1, s_i^n]$, is performed in time slot $s_i^n - 1$. Thus, the reveal delay in time interval $[s_i^n, s_{i+1}^n - 1]$ is $\{s_i^n - s_{i-1}^n + 1, s_i^n - s_{i-1}^n + 2, \cdots, s_{i+1}^n - s_{i-1}^n\}$ if the next collection is performed in time slot $s_{i+1}^n - 1$. Consequently, we have the expected reveal delay of device n

$$\mathbb{E}[\mathcal{D}_n] = \mathbb{E}\left[s_i^n - s_{i-1}^n + \frac{s_{i+1}^n - s_i^n + 1}{2}\right] = \frac{3T}{2\lambda_n} \tag{30}$$

This can be treated as an extreme case of data collection, and with more times of data collections, the expected reveal delay can be reduced. Hence, the expected reveal delay is at most $\frac{3T}{2\lambda_n}$ when the expected collection time interval $\frac{T}{u_n}$ is no more than the expected information change interval $\frac{T}{\lambda_n}$. □

To minimize the expected reveal delay with the requirement on the times of data collections ($\sum_{t=1}^{T} d_n(t) \geq \lambda_n^*$), we leverage the Max-Weight-Delay (MWD) Policy to solve $\mathbb{P}2$, and the performance should satisfy Lemma 1 and Lemma 2. Specifically, any agent constructs the Lyapunov Function of the data record status $\mathcal{S}_t$ according to the view of any agent, which is

$$\phi(\mathcal{S}_t) = \sum_{n=1}^{N} \left[x_n^+(t)\right]^2, \tag{31}$$

where

$$x_n^+(t) = \max\left\{p_n t - \sum_{t=1}^{T} d_n(t), \ 0\right\} \tag{32}$$

is the throughput debt associated with device n. The agent tends to reduce the Lyapunov Drift $\Delta(\mathcal{S}_t) = \mathbb{E}[\phi(\mathcal{S}_{t+1}) - \phi(\mathcal{S}_t)]$ between any successive time slots, which can be written by

$$\Delta(\mathcal{S}_t) = \sum_{n=1}^{N} \mathbb{E}\left\{\left[x_n^+(t+1)\right]^2 - \left[x_n^+(t)\right]^2\right\}. \tag{33}$$

According to [32], the upper bound associated with the throughput debt is

$$\begin{aligned} \mathbb{E}\Big\{ \left[x_n^+(t+1)\right]^2 - \left[x_n^+(t)\right]^2 \Big\} \leq \\ - 2x_n^+(t)\big(\mathbb{E}\{d_n(t)|\mathcal{S}_t\} - p_n\big) + 1, \end{aligned} \tag{34}$$

and the throughput requirements of all devices can be satisfied, which is $\mathbb{E}\left\{\frac{1}{T}\sum_{t=1}^{T} d_n(t)\right\} \geq \lambda_n / T = p_n, \ \forall n$. According

[3] We omit the subscript of the agent index m for simplicity.

[4] The AoI in [32] is different from the reveal delay in this article, where AoI is the time interval since the last data collection while reveal delay is the time interval since the latest sensing task that has been collected by the agent.

to Lemma 2, we have the upper bound of $\mathbb{E}\left\{\mathcal{D}_n(t)|\mathcal{S}_t\right\}$, which is

$$\mathbb{E}\left\{\mathcal{D}_n(t)|\mathcal{S}_t\right\} \leq \frac{3T}{2\lambda_n}. \tag{35}$$

By substituting (34) into (33), we have an upper bound for the value change of the Lyapunov function, which is

$$\Delta(\mathcal{S}_t) \leq -\sum_{n=1}^{N} \mathbb{E}\left\{d_n(t)|\mathcal{S}_t\right\} \Omega_n(t) + \Phi(t), \tag{36}$$

where $\Omega_n(t)$ and $\Phi(t)$ are given as follows

$$\Omega_n(t) = 2x_n^+(t), \tag{37}$$

$$\text{and} \quad \Phi(t) = \sum_{n=1}^{N} \left[2p_n x_n^+(t) + 1\right]. \tag{38}$$

Since $\Phi(t)$ is not impacted by the choice of data collection as shown in (36), we can derive that the selection strategy to minimize the overall weighted-sum reveal delay is collecting the data log from the device yielding the maximum $\Omega_n(k)$ so as to minimize the upper bound of $\Delta(\mathcal{S}_t)$ in (36), *i.e.*,

$$n^* = \arg\max_n \Omega_n(t). \tag{39}$$

However, data collection is performed by each individual agent, and each agent only concentrates on minimizing their own reveal delay. Next, we will state our motivation for incorporating the optimization introduced in this section into a blockchain-based platform and elaborate the detailed architecture.

V. Sustainable Blockchain-based Twin Management

The method introduced in Section IV provides a direction to realize a DT-based system that guides devices and agents to perform the sensing task and data collection optimizing both weighted-sum data fidelity and reveal delay. A naive architecture of such a DT-based system can be that each agent owns a single server that records data logs of devices, and performs data collection independently. However, this approach is not appropriate for the multi-agent and large-scale-device deployment case. The reasons are explained as follows:

- Firstly, DT on a single server is acceptable for the IoT ecosystem with a limited number of devices and requestors. However, the communication of all devices and requestors heavily relies on a single server, which incurs significant risks to the security of the DT-based system. The high-fidelity property of services may not be satisfied once the server has undermined faulty.
- Secondly, in the aforementioned architecture, all data logs of devices at a server are collected by a single agent. In fact, in industrial environment, agents are not always available to collect data from all devices, *e.g.*, an agent is out of the communication range of a specific device. With the contributions of more agents, the status of devices can be likely renewed more frequently, thus leading to less information loss.
- Finally, the resources of wireless channels are limited to be allocated for the communication between agents and devices. The simultaneous data collection at a device by multiple agents may trigger severe problems of interference.

Considering the previously mentioned disadvantages, we are motivated to incorporate more agents to participate in a shareable architecture for DT services. We will introduce the blockchain technology, widely acknowledged for trustworthy and shareable properties, as the backbone of our proposed architecture.

A. Architecture Overview

In our blockchain design, we use the consensus of Practical Byzantine Fault Tolerance (PBFT) [47] to process the requests of subscriptions and a modified consensus of PBFT, namely Max-Weight-Delay (MWD) consensus, to derive the sensing policy of devices and motivate the data collection of agents. A commonly used assumption in a distributed system is that any message can be received with a bounded delay, which guarantees weak synchrony. With the weak synchrony assumption and incorporating the functionalities of the original PBFT system, we characterize the agents as the following different roles for data collection:

- **Task Leader**: An agent whose local ledger serves as the primary view of the system is the task leader. Once a change needs to be applied to the system, it should be initiated by the task leader, and the records of the other agents serve as the backups of the primary view. All other agents vote for the assignments of data collection and acceptances of the data record changes, and then synchronize the ledger with all others.
- **Task Executor**: A task executor is an agent who performs the assigned data collection task. A data collection executor should be chosen from the participants who prefer to collect data of the selected device (an agent may not be able to perform data collection considering the availability). The agent who contributes more data logs to the open ledger (blockchain) should be awarded with more credits, which encourages participations and contributions of all the agents.

The ledger that records the subscription of requestors and the parameters of the system (e.g., depletion tolerance, energy supply rate) is the *request chain*, and the ledger that records the shared data of agents is the *data chain*. The implementation of the request chain simply follows the consensus of PBFT. For PBFT-based systems, a fundamental assumption is that the system should contains at least $M = 3f + 1$ agents to tolerate f faulty agents. Thus, the blocks in both the request chain and data chain can be only accepted if receiving at least $2f + 1$ confirmations from agents, where $f = \lfloor (M-1)/3 \rfloor$.

All the system changes in a stage will only take effect in the next stage. That means the popularity profiles of all agents and some other policies in any stage are fixed and open for all agents, and they will not change until the end the current stage. For example, the requests of any popularity change records in stage $k-1$ only updates at the beginning of stage k, *i.e.*, $t = 0, \ \forall k \in \{1, 2, \cdots, K\}$.

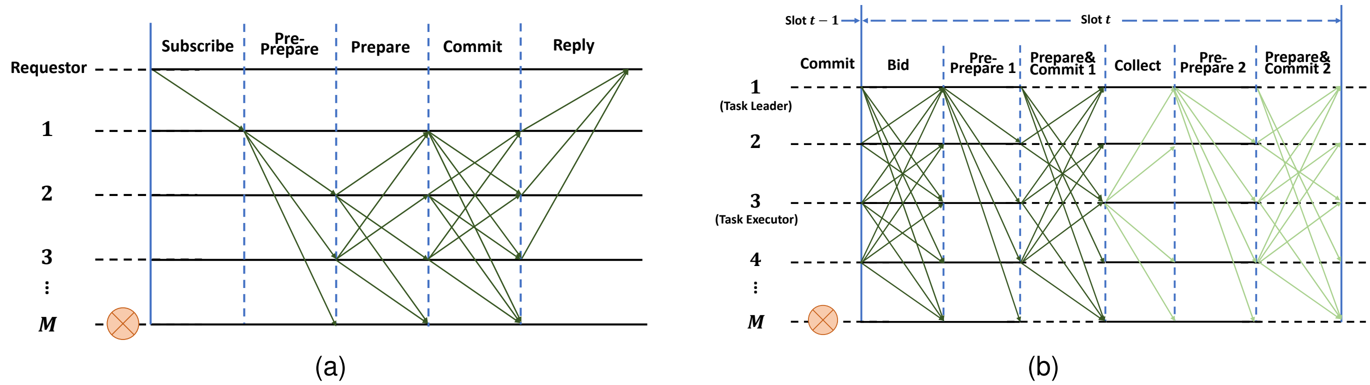

Fig. 3: Phases of a request chain [47] and a data chain.

In time slot $t = 0$, according to the popularity profiles and the energy policy of the system from the request chain, the sensing policies of devices are set to improve the weighted data fidelity while guaranteeing the sustainable energy supply. The maintenance of the request blocks has multi-phases of consensus including *pre-prepare*, *prepare*, *commit* and *reply* as shown in Fig.3a.

Based on the sensing policies of devices from the request chain and historical data collection records, in each time slot of a stage, an agent is chosen to execute the data collection task via the MWD consensus and updates the data records in the data chain. The selection of device is to optimize the weighted reveal delay and improve the information sustainability of DT-based models. We will next introduce the data updates of data chain under the MWD consensus.

B. Data Updates of Data Chain

Considering the subscription information is continuously updated in the request chain, as we declared in the system formulation, the popularity profiles are fixed until the stage ends. Since the goal of the MWD consensus is to optimize the sum-valuation of all devices, we use $\boldsymbol{\beta}(k) = \{\beta_1(k), \cdots, \beta_N(k)\}$ as the new weights of devices in stage k. In a stage, the consensus of data chain is based on the consensus obtained from the request chain. The execution steps for maintaining blocks in the data chain are characterized as follows:

1) According to agents' historical data collection performance shown in the current committed ledger, the agent with the highest reputation $\mathcal{R}$ is selected as the **task leader**. Without loss of generality, we can let the task leader be any agent with good reputation at the beginning.
2) In any time slot t, based on the state of the current open ledger $\mathcal{S}_t$, the task of data collection should be deterministic and associated with the decision in (39). Since the views of agents are consistent under the consensus, each agent could calculate the same best device selection n^*, and decide whether to bid the data collection for device n^* considering the availability. If an agent m decides to bid for collection task of n^*, it broadcasts the biding price $\mathcal{C}_b^m$ to all the other agents.
3) By receiving the biding prices of all the biding agents within a bounded delay, the **task leader** packs the biding information into a bidding block that indicates the agent m^* to perform the task with the consideration of both biding price and reputation, *i.e.*,
$$m^* = \arg\min_{m} \mathcal{N}(\mathcal{C}_b^m, \mathcal{R}_m), \tag{40}$$
where $\mathcal{N}$ can be any normalization function such as Min-Max technology [48]. The **task leader** then broadcasts the biding block to all other agents.
4) The biding block in previous step that has been committed by the system shows the next **task executor**. This **task executor** starts to perform the data collection task, and broadcasts the transaction including the collected data log to all the other agents after the confirmation of the biding block is acknowledged.
5) Upon receiving the collected data from the **task executor**, the **task leader** creates a data block, packs the transaction into the block, and further broadcasts this block to be confirmed by all the other agents. If the **task leader** does not receive the data from the **task executor** within a tolerant delay, she will issue a transaction that indicates a failure for this time of data collection instead of the transaction of the collected data.
6) Any agent who receives a biding/data block from the **task leader** should broadcast the *prepare* message of this block to all the other agents if the validation has passed; any agent who receives more than $2f$ *prepare* messages of the block with the same result from other agents, should broadcast the *commit* message to all the other agents. An agent, who is in the *prepare* state and receives more than $2f$ *commit* messages of the block, should formally commit this block and add this block to its local ledger.
7) Once both the biding block and data block are formally committed, the **task leader** receives fixed credits $\mathcal{C}_{cr}/(M-1)$ from each of the other agents for creating the committed block, where $\mathcal{C}_{cr}$ is the total credit reward for creating a biding block or a data block. If the data log of device n^* collected by **task executor** m^* has

been successfully written into the open ledger, agent m^* receives credits $(1+\tau)\mathcal{C}_b^{m^*}\alpha_n^m/\beta_n$ from agent m, where τ is the ratio of extra credit reward for the data collection; otherwise, agent m^* transfers $\mathcal{C}_b^{m^*}\alpha_n^m/\beta_n$ credits to agent m as a penalty. An agent whose credit is lower than a threshold will be excluded from the system in the next stage.

8) With the confirmation of a new data block with validated data, the reputation $\mathcal{R}$ of the **task executor** m^* increases according to its contribution of data collection. The reputation decays with time according to a widely-used exponential moving average (EMA) technique that highlights their most recent effort of task execution [48], *e.g.*,

$$\mathcal{R}_m(t+1) = \begin{cases} \omega\mathcal{R}_m(t), & m \neq m^* \\ \omega\mathcal{R}_m(t) + (1-\omega)\mathcal{C}_b^{m*}, & m = m^* \end{cases} \tag{41}$$

where $\omega \in [0,1]$ is a decay parameter and $\mathcal{R}_m(0) = 0$. The sustainable blockchain-based digital twin management system will turn to Step 1 for a new **task leader** election process.

The different consensus phases of the data chain are explained in Fig.3b. As shown in Fig.3b, in any time slot t, all the available agents bid for the collection task according to the committed ledger at the end of time slot $t-1$. As the primary view of all agents, the task leader performs the block creation tasks twice to assure all the other agents can receive a same referenced view of the biding and collecting results, respectively. The agent with the lowest data collection cost is selected to perform the data collection task. By receiving the referenced view of biding price and the collected data, all the agents broadcast two rounds of *prepare* and *commit* messages in the phases of *prepare & commit*, respectively. After the confirmation, the same process is repeated in time slot $t+1$.

C. Complexity Analysis

For the request chain, according to the change request made by a requestor, a task leader should broadcast $M-1$ messages to other agents to indicate the primary view. After that, it requires two round-trip decisions of *prepare* and *commit* with $\mathcal{O}(M^2)$ communication complexity to achieve consensus.

For the data chain, in phases of *prepare* and *commit*, the communication complexities are the same as those in the request chain. In the phase of biding, all biding agents should broadcast their own prices to other agents, thus needing at most $M*(M-1)$ messages. In the phase of collection, the task executor should send the collected data to all the other agents, which requires at most $M-1$ messages. Therefore, the overall communication complexity of both the request chain and data chain are $\mathcal{O}(M^2)$.

VI. Experimental Results

In this section, we demonstrate the extensive experiment results to verify our proposed architecture.

A. Energy Sustainability

We first verify the sustainability of energy supply of our system. Considering a DT-based IoT platform with N devices, we explore the sensing strategies of devices with different pre-defined parameters depletion tolerance degree ϵ, energy backup q and energy supply rate C.

Firstly, we explore the maximum energy consumption rate $\hat{\kappa}$ of all feasible sensing policies in terms of both ϵ and q for a given N and C. Fig.4 and Fig.5 plot the impacts of ϵ and q on the feasible sensing policy that yields the maximum κ when $N=100, C=10$ and $N=50, C=20$. It can be found that the increase of both ϵ and q triggers a more aggressive sensing rate. As can be seen in Fig.4 and 5, when q is large enough, we can employ a more stringent depletion tolerance setting, which still yields a relatively larger maximum energy consumption rate. However, when q is small, the maximum average energy consumption rate is sensitive to the change of ϵ as shown in Fig.4a and Fig.5a. As a result, for a DT system, it is important to precisely regulate the sensing policies of devices based on the conditions of energy backup and energy supply so as to meet the requirement of depletion tolerance degree.

To verify our analytical results in terms of depletion probability, we perform the experiments under a variety of depletion tolerance degrees. With $q=1$ and $K=300$, we run the energy consumption experiments 2000 times under different ϵ's with $N=50$, $C=20$, $N=100$, and $C=10$. As shown in Fig.6, the depletion rate is lower than the corresponding ϵ shown by the black dash line. We can find that the changes of depletion rate basically consist with the changes of ϵ. In practice, we can slightly relax the maximum average energy consumption rate constraint to achieve better system performance for a given ϵ.

In addition, we explore the relationship between the number of devices N and the overall maximum energy consumption $N\hat{\kappa}$. From Fig.7, we observe that for a fixed amount of energy supply, *e.g.*, $NC=1000$, in any stage, the maximum overall energy consumption for any feasible sensing policy increases with the increase of N. As shown in both Fig.7a and Fig.7b, it is clear that $N\hat{\kappa}$ is the largest one in the case when $N=200$ and $C=5$. Since the total incoming energy is fixed, the energy to be allocated to each device decreases with an increasing N. As a result, the fluctuation of energy consumption of each device is restrained, and the average fluctuation of the overall energy consumption is reduced. That means the system can exert more aggressive sensing policy on each device without depleting the energy storage when N is large. Moreover, as shown in Fig.7a, the differences of $N\hat{\kappa}$ in the three cases gradually diminish with the increment of ϵ when q equals to 1. Similarly, when ϵ is 0.01, $N\hat{\kappa}$ is limited when q is small as shown in Fig.7b. As q grows, $N\hat{\kappa}$ in all the cases finally converges to NC, which means nearly all the incoming energy can be allocated to all the devices for performing sensing tasks.

B. Weighted Data Fidelity

To verify the optimality of data fidelity of the system, we first establish a system with N IoT devices and $M=1$ agent. Without loss of generality, we normalize the popularity values

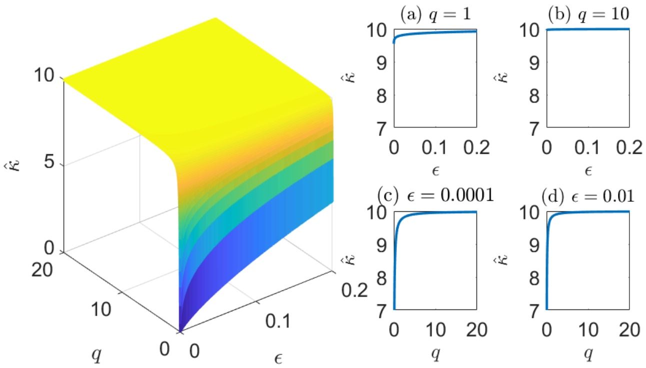

Fig. 4: Maximum energy consumption rate of all the feasible collection policies for different q's and ϵ's when $C = 10$ and $N = 100$.

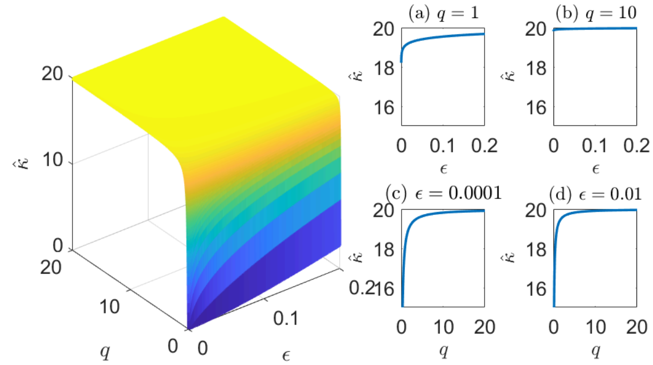

Fig. 5: Maximum energy consumption rate of all the feasible collection policies for different q's and ϵ's when $C = 20$ and $N = 50$.

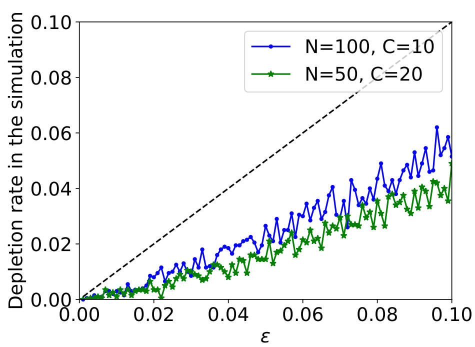

Fig. 6: The comparison of average depletion rate with the setting of $\hat{\kappa}(\epsilon)$, where $\epsilon \in [0.001, 0.1]$ with step size 0.001 and $q = 1$; N and C are chosen differently.

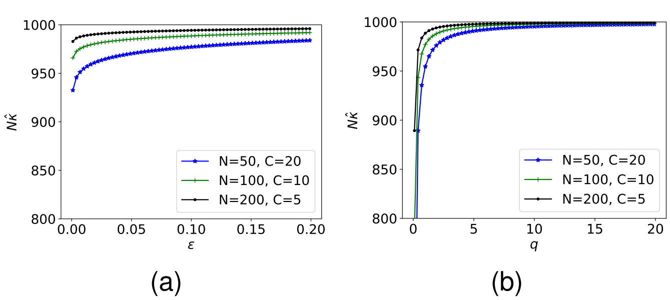

Fig. 7: The impact of N on the overall energy consumption $N\hat{\kappa}$: (a) $q = 1$; (b) $\epsilon = 0.01$.

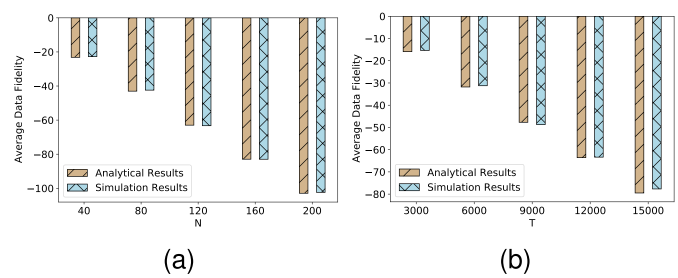

Fig. 8: The impacts of N and T on the weighted data fidelity: (a) $T = 10000$; (b) $N = 100$.

of devices between 0 to 1. Considering any stage with T slots, the parameters of the system are set as follows: $\epsilon = 0.0001$ and $q = 1$. According to the status of the DT system, the sensing policies of IoT devices $\boldsymbol{\lambda}$ are calculated by Algorithm 1. The analytical and experiment results of the optimal weighted data fidelity are explored with different settings of N and T.

First, we elaborate the relationship between the weighted data fidelity and the number of devices N. We set $T = 10000$ and $NC = 5000$. Fig.8a plots both the analytical and the experiment results of the averaged weighted data fidelity of all devices when we vary $N = \{40, 80, 120, 160, 200\}$. We assume that all devices have the same popularity, which is randomly sampled. As the energy supply is fixed in all the cases, the energy to be allocated to each device decreases with the increment of N. Thus, the sensing intervals also increase for energy saving, as a result, leading to the decrease of data fidelity of all devices.

Next, we discuss the relationship between the weighted data fidelity and the number of time slots T. We set $N = 100$ and $C = 50$. As shown in Fig.8b, the average weighted data fidelity of all devices also decreases with the increase of T, where T is chosen from $\{3000, 6000, 9000, 12000, 15000\}$. In this case, since the optimality of $\mathbb{P}1$ does not change with T, the optimal sensing policy of each device, *i.e.*, λ_n^*, $\forall n$, remains unchanged. That means the total number of sensing tasks performed by a device is fixed in any T. However, the increasing T would lead to an increasing expected intervals between any two successive sensing tasks, *i.e.*, T/λ_n, which leads to the decrements of weighted data fidelity of all devices. From both Fig.8a and Fig.8b, we can observe that the analytical results are consistent with the experiment results.

C. Weighted Reveal Delay

To test the weighted reveal delay performance of our proposed MWD policy, a simulation system consisting of M agents and N devices is constructed. The settings are the same as those in Section VI-B. The following data collection schemes are adopted to compare with the MWD

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3153653, IEEE Internet of Things Journal

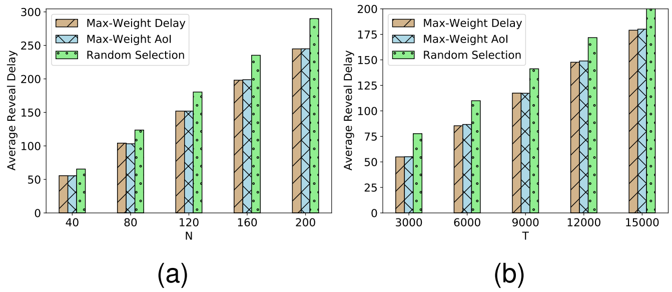

Fig. 9: The impacts of N and T on the weighted reveal delay: (a) $T = 10000$; (b) $N = 100$.

policy: 1) MWA policy [32], a max-weight policy aiming at minimizing the weighted AoI of all devices while considering the throughput; 2) Random Selection, a randomized data collection strategy to choose devices.

Fig.9a plots the results of the average weighted reveal delay of all devices when we vary the number of devices $N = \{40, 80, 120, 160, 200\}$ under the three collection schemes. As the energy supply is fixed in all the cases, the energy to be allocated to each device decreases with the increase of N, thus increasing the reveal delay. As shown in Fig.9b, the average weighted reveal delay of all devices also increases with an increasing T, where T is chosen from $\{3000, 6000, 9000, 12000, 15000\}$, since the average interval of any two successive sensing tasks increases when T is large. Notably, the performance of MWD and MWA are compatible, which reflects that the AoI metric does not have an impact on reveal delay once the throughput constraint is satisfied. The reason is that the AoI neglecting the execution of sensing tasks does not reflect the intrinsic property of the information source, and it does not have a direct relationship with the sustainability of the information.

D. MWD consensus

We evaluate MWD consensus with $N = 100$, $M = 20$, $C = 50$, $\epsilon = 0.0001$, $q = 1$ and $T = 10000$. The popularity of each agent is generated randomly from $[0, 1]$. We consider two system architectures, *MWD Consensus* and *Centralized MWD*. For *MWD Consensus*, the system is built based on the PBFT-blockchain, where all the agents observe the collection history, bid for the assignment, and share the collected data with all the other agents. For *Centralized MWD*, the agents' collection behaviors are coordinated by a centralized server, and they are randomly authorized to collect data with the MWD policy according to their own collection history. To ensure fairness, the centralized server can simply adopt Round-robin scheduling where each agent performs data collection in a circular manner. Supposing that each time of collection costs the executing agent with normalized credits sampled from a uniform distribution $[0, 1]$. Fig.10 plots the performance comparison between *MWD Consensus* and *Centralized MWD* in terms of average reveal delay and spent credits.

As can be seen in Fig.10a, the average reveal delay for all the agents in *MWD Consensus* is significantly reduced due to data sharing compared with those in *Centralized MWD*. With distributed data sharing, the time span for receiving two successive data logs is shortened, which leads to a smaller average reveal delay. In addition, the spent credits by the agents in *MWD consensus* are much fewer than those in Fig.10b compared with *Centralized MWD* due to the biding process. For a sharing mode, except for the collected information, the lower prices of collection are also shared by all participants, thus reducing the cost in a global view.

We also introduce the reward mechanism of data collection and creating blocks which aims to motivate the data sharing behaviors of agents. We randomly set the availability probability of all the agents from $[0, 1]$, which is the probability that an agent could provide a biding price (cost) for collecting data. The ratio of extra credit reward for the data collection τ is set as 0.1, and the reputation decay factor ω is set as 0.95. As shown in Section V-B, the income source of each agent is the reward of data collection and block creation. In Fig.11, both the earned credits and the number of generated blocks of all the agents with respect to their availability probabilities are plotted. It is clear that the credits for data collection and creating blocks of an agent increase with its increasing availability probability. In fact, an agent can earn more and improve reputation by actively collecting data. To do this, agents must continuously improve their connections with devices to reduce the cost of data collection. With a higher reputation, an agent can also gain more chances for serving as the primary view of all the agents and earn more credits from creating blocks in a blockchain-based DT system. The incentive effectiveness is therefore validated.

VII. Conclusion

In this article, we consider a sustainable blockchain-based digital twin management architecture for IoT devices. This is to address the sustainability issue for large-scale DT service management. The blockchain technology, as a shareable and distributed paradigm, is promising to improve and reform future DT systems. We expect to explore more practical applications of blockchain-based DT design. For example, DT visualization for real-time monitoring is critical to guiding industrial processes and improving production efficiency. We will explore the visualization of DTs in different entities using shared data from a blockchain. The method presented in this article will be beneficial to the construction of DT visualization for multiple entities with customized needs.

References

[1] F. Al-Turjman and M. Abujubbeh, "Iot-enabled smart grid via sm: An overview," *Future Generation Computer Systems*, vol. 96, pp. 579–590, 2019.

[2] T. hoon Kim, C. Ramos, and S. Mohammed, "Smart city and iot," *Future Generation Computer Systems*, vol. 76, pp. 159–162, 2017.

[3] B. Jan, H. Farman, M. Khan, M. Talha, and I. U. Din, "Designing a smart transportation system: An internet of things and big data approach," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 73–79, 2019.

[4] A. Ahad, M. Tahir, M. Aman Sheikh, K. I. Ahmed, A. Mughees, and A. Numani, "Technologies trend towards 5g network for smart healthcare using iot: A review," *Sensors*, vol. 20, no. 14, p. 4047, 2020.

[5] Oracle, "About the oracle iot digital twin implementation," Sep. 15, 2021. [Online]. Available: https://docs.oracle.com/en/cloud/paas/iot-cloud/iotgs/oracle-iot-digital-twin-implementation.html

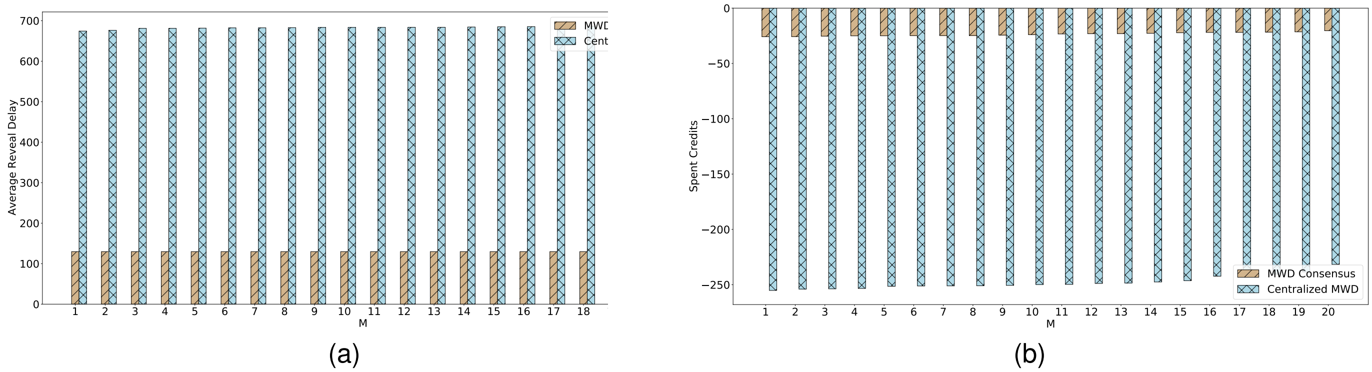

Fig. 10: The performance comparison between *MWD Consensus* and *Centralized MWD* in terms of (a) Average Reveal Delay and (b) Spent Credits.

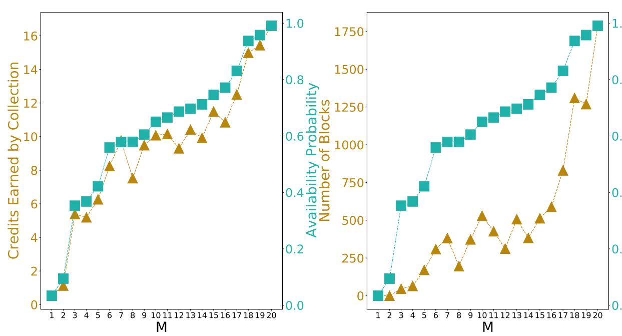

Fig. 11: Performance of the incentive mechanism.

[6] B. R. Barricelli, E. Casiraghi, J. Gliozzo, A. Petrini, and S. Valtolina, "Human digital twin for fitness management," *IEEE Access*, vol. 8, pp. 26 637–26 664, 2020.

[7] Q. Lu, A. K. Parlikad, P. Woodall, G. Don Ranasinghe, X. Xie, Z. Liang, E. Konstantinou, J. Heaton, and J. Schooling, "Developing a digital twin at building and city levels: Case study of west cambridge campus," *Journal of Management in Engineering*, vol. 36, no. 3, p. 05020004, 2020.

[8] Siemens, "Comos – making data work," Sep. 15, 2021. [Online]. Available: https://cache.industry.siemens.com/dl/files/354/109765354/att_978457/v1/COMOS_Imagebroschuere_EN.pdf

[9] M. Matulis and C. Harvey, "A robot arm digital twin utilising reinforcement learning," *Computers & Graphics*, vol. 95, pp. 106–114, 2021.

[10] M. G. Kapteyn, J. V. Pretorius, and K. E. Willcox, "A probabilistic graphical model foundation for enabling predictive digital twins at scale," *Nature Computational Science*, vol. 1, no. 5, pp. 337–347, 2021.

[11] A. Saad, S. Faddel, T. Youssef, and O. A. Mohammed, "On the implementation of iot-based digital twin for networked microgrids resiliency against cyber attacks," *IEEE transactions on smart grid*, vol. 11, no. 6, pp. 5138–5150, 2020.

[12] S. H. Khajavi, N. H. Motlagh, A. Jaribion, L. C. Werner, and J. Holmström, "Digital twin: vision, benefits, boundaries, and creation for buildings," *IEEE access*, vol. 7, pp. 147 406–147 419, 2019.

[13] H. Wang, Y. Wu, G. Min, and W. Miao, "A graph neural network-based digital twin for network slicing management," *IEEE Transactions on Industrial Informatics*, 2020.

[14] L. Corneo, C. Rohner, and P. Gunningberg, "Age of information-aware scheduling for timely and scalable internet of things applications," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 2476–2484.

[15] H. El Alami and A. Najid, "Ech: An enhanced clustering hierarchy approach to maximize lifetime of wireless sensor networks," *IEEE Access*, vol. 7, pp. 107 142–107 153, 2019.

[16] X. Jin, F. Kong, L. Kong, H. Wang, C. Xia, P. Zeng, and Q. Deng, "A hierarchical data transmission framework for industrial wireless sensor and actuator networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2019–2029, 2017.

[17] Y. Wu, "Cloud-edge orchestration for the internet-of-things: Architecture and ai-powered data processing," *IEEE Internet of Things Journal*, 2020.

[18] Z. Cai and Z. He, "Trading private range counting over big iot data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 144–153.

[19] J. Du, C. Jiang, E. Gelenbe, L. Xu, J. Li, and Y. Ren, "Distributed data privacy preservation in iot applications," *IEEE Wireless Communications*, vol. 25, no. 6, pp. 68–76, 2018.

[20] Z. Cai and T. Shi, "Distributed query processing in the edge assisted iot data monitoring system," *IEEE Internet of Things Journal*, 2020.

[21] S. Suhail, R. Hussain, R. Jurdak, and C. S. Hong, "Trustworthy digital twins in the industrial internet of things with blockchain," *IEEE Internet Computing*, 2021.

[22] Y. Wu, H.-N. Dai, H. Wang, and K.-K. R. Choo, "Blockchain-based privacy preservation for 5g-enabled drone communications," *IEEE Network*, vol. 35, no. 1, pp. 50–56, 2021.

[23] S. Zhu, W. Li, H. Li, L. Tian, G. Luo, and Z. Cai, "Coin hopping attack in blockchain-based iot," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4614–4626, 2018.

[24] Y. Wu, H.-N. Dai, and H. Wang, "Convergence of blockchain and edge computing for secure and scalable iiot critical infrastructures in industry 4.0," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2300–2317, 2020.

[25] A. M. Madni, C. C. Madni, and S. D. Lucero, "Leveraging digital twin technology in model-based systems engineering," *Systems*, vol. 7, no. 1, p. 7, 2019.

[26] W. Sun, H. Zhang, R. Wang, and Y. Zhang, "Reducing offloading latency for digital twin edge networks in 6g," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 12 240–12 251, 2020.

[27] S. Haag and R. Anderl, "Digital twin–proof of concept," *Manufacturing Letters*, vol. 15, pp. 64–66, 2018.

[28] J. Leng, Q. Liu, S. Ye, J. Jing, Y. Wang, C. Zhang, D. Zhang, and X. Chen, "Digital twin-driven rapid reconfiguration of the automated manufacturing system via an open architecture model," *Robotics and Computer-Integrated Manufacturing*, vol. 63, p. 101895, 2020.

[29] X. Tong, Q. Liu, S. Pi, and Y. Xiao, "Real-time machining data application and service based on imt digital twin," *Journal of Intelligent Manufacturing*, vol. 31, no. 5, pp. 1113–1132, 2020.

[30] Q. Qi, F. Tao, Y. Zuo, and D. Zhao, "Digital twin service towards smart manufacturing," *Procedia Cirp*, vol. 72, pp. 237–242, 2018.

[31] J. Leng, H. Zhang, D. Yan, Q. Liu, X. Chen, and D. Zhang, "Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop," *Journal of ambient intelligence and humanized computing*, vol. 10, no. 3, pp. 1155–1166, 2019.

[32] I. Kadota, A. Sinha, and E. Modiano, "Optimizing age of information in wireless networks with throughput constraints," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1844–1852.

[33] B. Zhou and W. Saad, "Optimal sampling and updating for minimizing age of information in the internet of things," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[34] H. Zheng, K. Xiong, P. Fan, Z. Zhong, and K. B. Letaief, "Age of information-based wireless powered communication networks with selfish charging nodes," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1393–1411, 2021.

[35] E. Gindullina, L. Badia, and D. Gündüz, "Age-of-information with information source diversity in an energy harvesting system," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1529–1540, 2021.

[36] S. Suhail, R. Hussain, R. Jurdak, A. Oracevic, K. Salah, and C. S. Hong, "Blockchain-based digital twins: Research trends, issues, and future challenges," *CoRR*, vol. abs/2103.11585, 2021. [Online]. Available: https://arxiv.org/abs/2103.11585

[37] B. Putz, M. Dietz, P. Empl, and G. Pernul, "Ethertwin: Blockchain-based secure digital twin information management," *Information Processing & Management*, vol. 58, no. 1, p. 102425, 2021.

[38] C. Mandolla, A. M. Petruzzelli, G. Percoco, and A. Urbinati, "Building a digital twin for additive manufacturing through the exploitation of blockchain: A case analysis of the aircraft industry," *Computers in Industry*, vol. 109, pp. 134–152, 2019.

[39] Z. Sun, Y. Wang, Z. Cai, T. Liu, X. Tong, and N. Jiang, "A two-stage privacy protection mechanism based on blockchain in mobile crowdsourcing," *International Journal of Intelligent Systems*, vol. 36, no. 5, pp. 2058–2080, 2021.

[40] S. Zhu, Z. Cai, H. Hu, Y. Li, and W. Li, "zkcrowd: a hybrid blockchain-based crowdsourcing platform," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4196–4205, 2019.

[41] J. L. Hodges and L. Le Cam, "The poisson approximation to the poisson binomial distribution," *The Annals of Mathematical Statistics*, vol. 31, no. 3, pp. 737–740, 1960.

[42] H. Touchette, "The large deviation approach to statistical mechanics," *Physics Reports*, vol. 478, no. 1-3, pp. 1–69, 2009.

[43] D. W. Stroock, *Probability theory: an analytic view*. Cambridge university press, 2010.

[44] A. J. Ganesh, N. O'Connell, and D. J. Wischik, *Big queues*. Springer, 2004.

[45] J. T. Lewis and R. Russell, "An introduction to large deviations for teletraffic engineers," *Dublin Institute for Advanced Studies*, pp. 1–45, 1997.

[46] A. Rovira-Sugranes and A. Razi, "Optimizing the age of information for blockchain technology with applications to iot sensors," *IEEE Communications Letters*, vol. 24, no. 1, pp. 183–187, 2019.

[47] M. Castro, B. Liskov *et al.*, "Practical byzantine fault tolerance," in *OSDI*, vol. 99, no. 1999, 1999, pp. 173–186.

[48] L. Yuan, Q. He, S. Tan, B. Li, J. Yu, F. Chen, H. Jin, and Y. Yang, "Coopedge: A decentralized blockchain-based platform for cooperative edge computing," in *Proceedings of the Web Conference 2021*, 2021, pp. 2245–2257.

Chenyu Wang received his B.S. degree from Xiangtan University, Hunan, China and M.S. degree from Beijing Normal University, Beijing, China. He is currently pursuing the Ph.D. degree in the Department of Computer Science at Georgia State University. His research interests include network optimization and machine learning.

Zhipeng Cai received his Ph.D. and M.S. degree from University of Alberta, and B.S. degree from Beijing Institute of Technology. Dr. Cai is currently an Associate Professor in the Department of Computer Science at Georgia State University. Prior to joining GSU, Dr. Cai was a research faculty in the School of Electrical and Computer Engineering at Georgia Institute of Technology. Dr. Cai's research areas focus on Networking and Big data. He is the steering committee chair of the International Conference on Wireless Algorithms, Systems, and Applications (WASA). Dr. Cai served/is serving on the editorial boards of several technical journals (e.g. IEEE Internet of Things Journal, IEEE Transactions on Knowledge and Data Engineering (TKDE), IEEE Transactions on Vehicular Technology (TVT)). He also chaired several international conferences (e.g. IEEE ICDCS 2019, IEEE IPCCC18). Dr. Cai is the recipient of an NSF CAREER Award.

Yingshu Li received her Ph.D. and M.S. degrees from University of Minnesota-Twin Cities, Minneapolis, MN, USA. She received her B.S. degree from Beijing Institute of Technology, Beijing, China. Dr. Li is currently a Professor in the Department of Computer Science at Georgia State University. Her research interests include Privacy-aware Computing, Management of Big Sensory Data, Internet of Things, Social Networks, and Wireless Networking. Dr. Li is the recipient of the NSF CAREER Award. She has served as an Associate Editor or Guest Editor for some prestigious journals such as ACM Transactions on Sensor Networks, IEEE Transactions on Computers, IEEE Transactions on Network Science and Engineering, and IEEE Internet of Things Journal. She has also served as a Steering Committee Chair, General Chair, Program Chair, and Technical Program Committee member for many international conferences.