# Blockchain-Empowered Lifecycle Management for AI-Generated Content Products in Edge Networks

Yinqiu Liu, Hongyang Du, Dusit Niyato, Jiawen Kang, Zehui Xiong, Chunyan Miao, Xuemin (Sherman) Shen, and Abbas Jamalipour

## Abstract

The rapid development of Artificial Intelligence-Generated Content (AIGC) has brought daunting challenges in the areas of service latency, security, and trustworthiness. Recently researchers have presented the edge AIGC paradigm, effectively optimizing the service latency by distributing AIGC services to edge devices. However, AIGC products are still unprotected and vulnerable to tampering and plagiarization. Moreover, as a kind of online non-fungible digital property, the free circulation of AIGC products is hindered by the lack of trustworthiness in open networks. For the first time, in this article, we present a blockchain-empowered framework to manage the lifecycle of edge AIGC products. Specifically, leveraging fraud proofability, we first propose a protocol to protect the ownership and copyright of AIGC, called Proof-of-AIGC. Then, we design an incentive mechanism to guarantee the legitimate and timely execution of the funds-AIGC ownership exchanges among anonymous users. Furthermore, we implement a multi-weight subjective logic-based reputation scheme with which AIGC producers can determine which edge service provider is trustworthy and can reliably handle their services. Through the use of test data, we demonstrate the superiority of the proposed approach. Last but not least, we discuss important open directions for further research.

## Introduction

As an emerging technique, Artificial Intelligence-Generated Content (AIGC) has attracted significant attention from both academia and industry [1]. Instead of manually generating the content, AIGC enables the automatic creation (e.g., writing an essay, composing a song, and drawing a picture) using machine learning techniques such as Generative Adversarial Networks (GAN) and diffusion models. Consequently, we can acquire massive high-quality multimodal content while significantly saving on the required labor. Since 2014, AIGC has experienced rapid development and has been widely adopted in 3D gaming, voice assistants, video processing, and so on [2].

However, the current centralized AIGC framework suffers from high service latency. For instance, to generate an image on *Hugging Face* platform (https://huggingface.co/spaces) using *Stable Diffusion* model, users have to wait for 40–60 seconds. The reasons are twofold. Firstly, AIGC inference is complicated and time-consuming. In the above example, the Stable Diffusion model creates images from scratch by conducting denoising operations gradually, which takes around 20–30 seconds. Moreover, the queueing latency is also considerable (20–30 seconds in our example) since massive service requests congest one central server.

Recently, researchers have presented the idea of edge AIGC, which deploys AIGC generation services on edge devices [1]. By distributing services to numerous edge devices which are close to users, service latency can be effectively reduced. Meanwhile, the robustness gets increased due to the elimination of single-point-failure. Moreover, users can customize AIGC services, for example, sharing their background, locations, or characters with edge devices to generate personalized content accordingly. Finally, since the users directly communicate with edge devices, personal information can be protected from leakage. Although enjoying these advantages, the following challenges exist in deploying edge AIGC.

As digital property on the Internet, AIGC products are vulnerable to tampering and plagiarization (the tampering and plagiarization are shown later).

The economic system of AIGC is complicated. Without a mechanism guaranteeing that all the participants can benefit from AIGC circulation and obtain their deserved revenue legitimately, the generation, distribution, and trading of AIGC products will be discouraged.

Recall that the generation services become distributed in edge AIGC. Therefore, as Edge Service Providers (ESPs) show significant heterogeneity in terms of model configuration and service quality, the users can hardly select reliable ESPs for their tasks.

Fortunately, blockchain provides available solutions for these issues. As a distributed ledger, blockchain can construct trustworthiness

*Yinqiu Liu, Hongyang Du, Dusit Niyato, and Chunyan Miao are with Nanyang Technological University, Singapore; Jiawen Kang (corresponding author) is with Guangdong University of Technology, China; Zehui Xiong is with Singapore University of Technology and Design, Singapore; Xuemin Shen is with the University of Waterloo, Canada; Abbas Jamalipour is with the University of Sydney, Australia.*

Digital Object Identifier: 10.1109/MWC.003.2300053



1536-1284/24/$25.00 © 2024 IEEE

among anonymous participants by maintaining an immutable and traceable history [3]. Moreover, smart contracts make the blockchain programmable, enabling the on-chain deployment of arbitrarily complex mechanisms (e.g., two-phase locks and incentive mechanisms). Consequently, the status and trading of AIGC products can be monitored on-chain, eliminating the security and trustworthiness problems. In 2022, Oben AI published the proposal of *AIGC chain* (https://www.aigcchain.io/about), which allows users to contribute resources for training distributed AIGC models and acquiring rewards. As the first blockchain for AIGC, however, this project is still under development and far from completing the whole ecosystem. Moreover, it only uses blockchain as a crowdsourcing platform for generating AIGC, while the distribution and trading of AIGC are unprotected.

In this article, we propose the blockchain-empowered AIGC product lifecycle management in edge networks. Specifically, we first define "*AIGC product lifecycle*" and discuss four major concerns regarding lifecycle management. To help AIGC products defend malicious attacks, a Proof-of-AIGC mechanism is proposed, using fraud proofs to deal with plagiarization. Given the complex economic system of AIGC, we further equip our framework with an on-chain incentive mechanism based on Hash Time Look (HTL) [4]. With guaranteed and timely revenue issuance, the circulation of AIGC can be motivated and incentivized. Finally, noticing the heterogeneity of ESPs, we enable AIGC producers to select the ESPs based on their accumulated reputation, which is modeled by Multi-weight Subjective Logic (MWSL) method [5]. *To the best of our knowledge, this is the first work discussing the issues and solutions of AIGC product lifecycle management.* Our contributions are summarized as follows:

- We present Proof-of-AIGC mechanism. Different from Proof-of-X (e.g., Proof-of-Semantics [6]), a challenge scheme is implemented, thus deregistering plagiarized AIGC products and protecting users' copyright.
- We propose an incentive mechanism with one-way incentives and two-way guarantees. The former encourages users to participate in managing the AIGC product lifecycle, and the latter ensures the atomic executions of AIGC trading, that is, funds-ownership exchanges.
- We design a reputation-based ESP selection strategy. By calculating and sharing reputation, users can easily quantify the trustworthiness of numerous heterogeneous ESPs and assign their tasks to the most reliable one.

## AIGC: Current Progress, Lifecycle Management, and Concerns

In this section, we first review the development of AIGC. Then, we show the AIGC product lifecycle in edge networks. Finally, important security and circulation concerns existing in the AIGC product lifecycle are discussed.

### Development of AIGC

AIGC is an emerging generation diagram after Professional-Generated Content and User-Generated Content. As the name suggests, the development of AIGC is driven by progress in AI research. Before 2010, machines can hardly generate high-quality content due to the limited capability of deep learning models. Since 2014, various generative neural networks have been presented, such as GAN and variational autoencoders. Consequently, AIGC enters a period of rapid development. In 2020, OpenAI published the *Generative Pre-trained Transformer-3* (GPT-3) model, supporting multiple text generation tasks, for example, mechanism translation and report creation [7]. Two years later, the diffusion-based *DALL-E-2* model is presented. Based on the text description given by users, DALL-E-2 can generate high-quality realistic images automatically. Apart from text-to-text and text-to-image generation, AIGC is widely adopted in video processing, gaming, voice assistants, and so on. Moreover, it is regarded as a building block for many revolutionary techniques, including Web3, metaverse, digital twin, even the future 7G [8].

### AIGC Product Lifecycle Management in Edge Networks

Traditionally, AIGC models are operated by centralized servers, such as *Hugging Face* platform. In this case, massive users send requests to the central server, wait in line, and receive the services. Researchers attempt to deploy AIGC services in edge networks to avoid request congestion and optimize service latency. Compared with central servers, edge devices also have enough computing resources to conduct AIGC inference and are closer to users. Therefore, the users can communicate with devices with lower transmission latency. Moreover, since AIGC services are distributed to multiple edge devices, the waiting latency can be significantly decreased. Nonetheless, the current research only covers the generation of AIGC products. As a kind of non-fungible online property like NFT [9], each AIGC product has its ownership, copyright, and value. Accordingly, the protection and management of AIGC products should cover their whole lifecycle. Next, we define the concept of "AIGC product lifecycle."

The entire AIGC product lifecycle has three phases, namely generation, distribution, and trading (Steps ①–③ in Fig. 1). Taking text-to-image generation as an example, the primary process of each phase is described below.

**Generation:** Producers, with insufficient physical resources, pack prompts, that is, interesting and accurate text descriptions, and requirements in a request and send them to ESPs (Step ①). Edge devices serve as ESPs, providing AIGC generation services for clients using local well-trained AIGC models (Step ②). Since AIGC generation is time-consuming and takes computing resources, ESPs can claim fees from producers.

**Distribution:** After generation, the producers acquire the ownership of the AIGC products. Consequently, they have the right to distribute these products to social media or AIGC platforms through edge networks (Step ③).

**Trading:** Since AIGC products are regarded as a novel kind of non-fungible digital properties, they can be traded. The trading process can be modelled as a fund-ownership exchange between two parties.

During such a lifecycle, several issues are yet to be addressed. As shown in Fig. 1, firstly, the ownership and copyright of AIGC products are vulnerable on the Internet. Meanwhile, the producers

Compared with central servers, edge devices also have enough computing resources to conduct AIGC inference and are closer to users. Therefore, the users can communicate with devices with lower transmission latency.

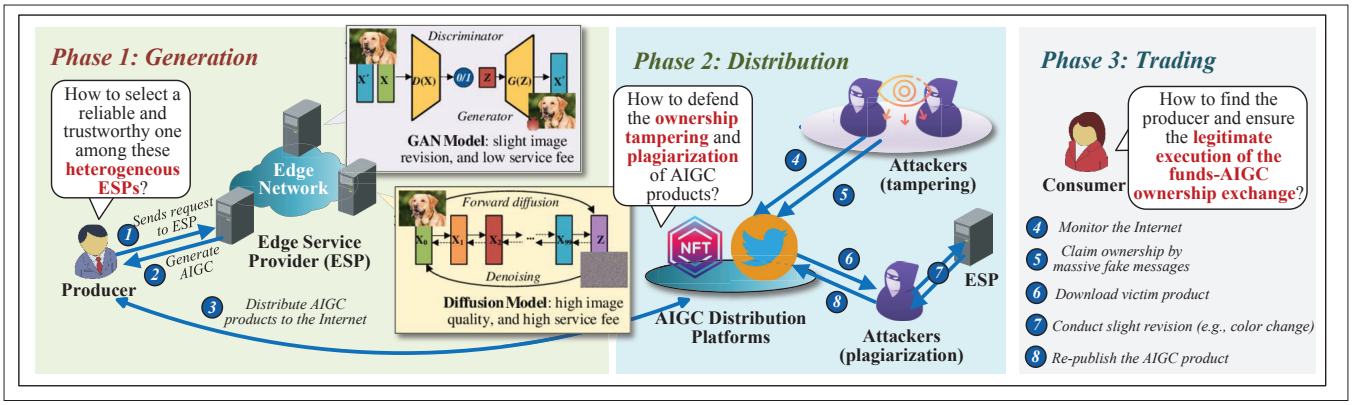


**FIGURE 1**. The AIGC product lifecycle and its important concerns.

also encounter problems in choosing reliable ESPs. Finally, the legitimate trading of AIGC products among anonymous participants is unsolved. In the following part, we discuss these concerns in detail.

### Security Concerns

Since AIGC products are published on the Internet, various kinds of attacks threaten them [10]. Here, we illustrate two crucial attacks targeting the AIGC products, namely the tampering of ownership and the plagiarization of AIGC.

**Tampering of Ownership:** Taking text-to-image AIGC as an example, first, Steps ①–③ in Fig. 1 illustrate its lifecycle. For conducting ownership tampering, the attackers generally deploy many robots to monitor the Internet closely and find high-quality AIGC products timely (Step ④). After selecting the victim image, the attacker, assisted by its robots, distributes massive messages to re-publish the image, pretending that the image is its original (Step ⑤). With robots, the attackers typically broadcast information more rapidly. Hence, the consumers can be misled by attackers. For instance, in Oct. 2022, a prominent artist named AT was painting the Raiden Shogun in front of live audiences. A hacker took the in-process image, reproduced the image with Novel AI (https://novelai.net/), and uploaded it six hours before AT. Although AT is clearly the real producer, he/she can hardly retrieve the ownership due to the lack of authentication platforms which record and protect the AIGC product generation phase [10].

**Plagiarization of AIGC:** Compared with ownership tampering, the plagiarization of AIGC is harder to be detected. In this case, the attacker will not directly claim ownership of the victim image. Instead, it downloads the high-quality victim image, conducts some slight revision (e.g., adding noise or changing the colors of some objects), and publishes it as a brand-new AIGC product (Steps ⑥–⑧). Since such revision is much easier and cheaper than generating AIGC images from scratch, the attacker can make profits at low costs. In fact, this kind of malicious behaviors has happened in non-fungible token markets. For example, in 2021, the creators of Phunky Ape Yacht Club collections made huge profits by just flipping the high-priced Bored Ape Yacht Club avatars and selling them as the original digital art products [10].

### Circulation Concerns

Apart from security concerns, to realize the free circulation of AIGC, we also encounter two challenges.

**Heterogeneity of ESPs:** The lifecycle of every AIGC product starts from generation, that is, using well-trained AIGC models in ESPs to create content based on producers' prompts. Nonetheless, ESPs in edge networks show great heterogeneity in model and service quality. Taking Fig. 1 as an example, one ESP is equipped with *Stable Diffusion* (https://stability.ai/), the state-of-the-art AIGC model. The training of Stable Diffusion is called forward diffusion, that is, smoothly perturbing the original image data by adding noise. The corresponding training time exceeds 150,000 hours on 256 Nvidia A100 GPUs, at a cost of US$600,000. In contrast, another ESP in Fig. 1 only has a simple GAN model. The quality of the resulting content generated by these two ESPs is significantly different. However, since ESPs may lie to producers, they cannot determine which ESPs are trustworthy.

**Issuance of the Deserved Revenue:** As a kind of non-fungible digital properties, the lifecycle of each AIGC product contains multiple transactions. Firstly, given the high costs of AIGC generation, the computing power and time invested by ESPs should be rewarded with fees. Meanwhile, the producers are only willing to pay if they are guaranteed to receive the AIGC products on time. Likewise, AIGC trading also involves a two-way guarantee of whether the producer and consumer can obtain the funds and AIGC ownership, respectively. However, on the public Internet, the two parties of transactions can hardly build trustworthiness. Such a concern might discourage the producers from distributing and trading products, thus blocking the free circulation of AIGC products.

From the above discussion, we can observe that the difficulty of AIGC lifecycle management originates from two issues, that is, the intrinsic venerability of AIGC as a kind of digital non-fungible property, and the lack of trustworthiness on the Internet. Fortunately, as an immutable ledger and trust maker, blockchain can effectively solve these two issues.

## Blockchain-Empowered AIGC Product Lifecycle Management

### Framework Overview

The proposed blockchain-based framework for AIGC product lifecycle management is shown in Fig. 2. In the following part, we introduce this framework in terms of stakeholders, blockchain platform, and on-chain mechanisms.

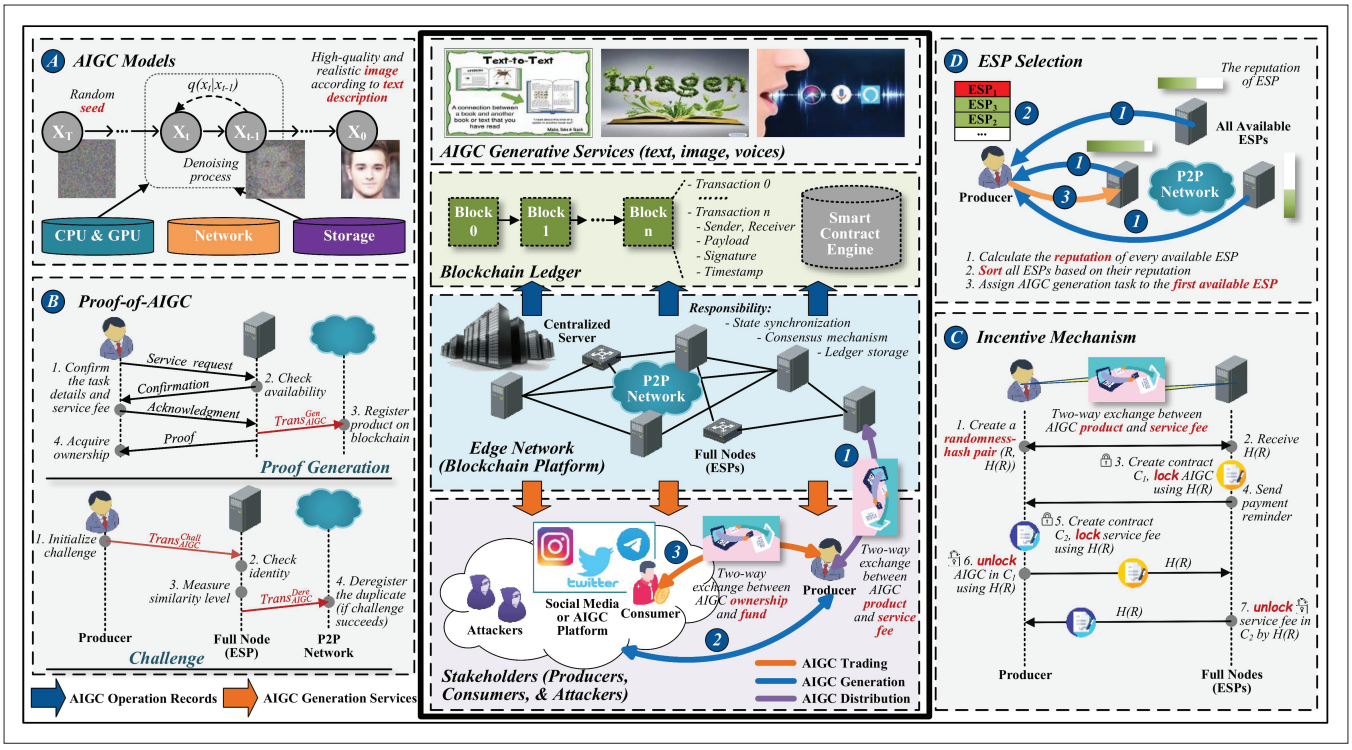


**FIGURE 2**. The blockchain-empowered framework for AIGC product lifecycle management. Part A shows the AIGC models in each ESP. Parts B, C, and D illustrate the Proof-of-AIGC, incentive mechanism, and reputation-based ESP selection, respectively.

**Stakeholders:** The entire AIGC product lifecycle in edge networks involves four types of stakeholders in total, namely producers, ESPs, consumers, and attackers.

- *Producer:* Producers initialize the lifecycle of an AIGC product. Due to resource limitations, they only propose prompts (e.g., interesting and accurate text descriptions in text-to-image AIGC) and then request for ESPs to complete the generation tasks. After the generation, they become the first owners of the resulting products and have the right to publish and sell them.
- *ESP:* ESPs, served by edge servers, have enough resources to save well-trained AIGC models and generate content (see Fig. 2, Part A). Therefore, they can provide content generation services for producers. However, given the complexity of AIGC generation, ESPs can charge producers based on the time and computing power that they invest in the tasks.
- *Consumer:* After distribution, the AIGC product will be viewed by numerous people, some of whom may buy it. Such viewers are called consumers. During the lifecycle of an AIGC product, it might experience multiple times of trading with different consumers.
- *Attacker:* Attackers can launch the ownership tampering and plagiarization to destroy the normal AIGC product lifecycle and make profits.

**Blockchain Platform:** In our framework, blockchain has two major functions: providing a traceable and immutable ledger, and supporting on-chain mechanisms. To this end, every phase of the AIGC product lifecycle will be recorded by transactions, whose basic format is *Trans* (*Sender*, *Receiver*, *Payload*, *Timestamp*, *Signature*). Note that the payload is different depending on the specific types of events. Transactions are packed into blocks and submitted to the blockchain network, a distributed Peer-to-Peer (P2P) network. The participants of the P2P network, named full nodes, conduct a consensus mechanism for block verification. Finally, valid blocks can be appended to the ledger and saved by all full nodes in parallel. Since everyone preserves a ledger copy, the attackers have to revise at least 50 percent of copies for tampering history, which is impossible in practice.

Among all participants, ESPs serve as full nodes and are responsible for message synchronization, block verification, and ledger storage. Given the resource limitation, producers and consumers act as clients, relying on ESPs to access the blockchain services. Note that the consensus mechanism in our blockchain is delegated Proof-of-Stake [11], in which ESPs deposit stakes and take turns to create blocks.

**On-Chain Mechanism:** The framework is equipped with a Turing-complete smart contract engine, with which three on-chain mechanisms are implemented. Firstly, we design the Proof-of-AIGC mechanism to defend plagiarization (Fig. 2, Part B). To protect the funds-AIGC ownership exchange, we further implement an incentive mechanism based on HTL (Fig. 2, Part C). Finally, we present the reputation-based ESP selection, which effectively schedules AIGC generation tasks among ESPs (Fig. 2, Part D).

## Proof of AIGC

As shown in Fig. 2, Part B, the Proof-of-AIGC consists of two phases, namely proof generation and challenge.

**Proof Generation:** Proof generation intends to register AIGC products on blockchain. Here, we still take text-to-image AIGC as an example. For generating an image, the producer sends the request to an ESP and then performs a handshake to confirm the service fee. Then, the image creation can be conducted by the ESP, using well-

The economic system of AIGC is complicated because it accommodates different stakeholders, which conduct transactions with each other frequently.

trained AIGC models. After that, ESP initializes a transaction $Trans^{Gen}_{AIGC}$(*Sender*, *Receiver*, *Payload*, *Timestamp*, *Signature*). The Payload format is (*Product index*, *Metadata*), in which *Product index* is calculated by hash function and is regarded as the unique identity for each AIGC product. *Metadata* contains some basic information about the AIGC product. Such a transaction will go through the verification and be recorded by the blockchain. Finally, the ESP will send the image to the producer, with a copy of $Trans^{Gen}_{AIGC}$. $Trans^{Gen}_{AIGC}$ can be regarded as a proof, which not only registers the AIGC product, but also claims its ownership by setting *Receiver* as the producer's address.

**Challenge:** Proof-of-AIGC follows the principle of fraud proof. In other words, we assume that all AIGC products are original work in the proof generation phase. However, the information recorded in $Trans^{Gen}_{AIGC}$ enables producers to challenge any on-chain AIGC product that they believe copies their own work. If the challenge succeeds, the duplicate will be deregistered, thus protecting the copyright of the original producer. Next, we show the challenge workflow.

Suppose that the producer has published an AIGC product (called original product). Then, it surfs the Internet and finds an AIGC product which is significantly similar to its own work (called duplicate). In this case, it can initialize the challenge process by sending a transaction $Trans^{Chall}_{AIGC}$ with the payload ($Product_1$, $Product\ index_1$, $Product_2$, $Product\ index_2$, *Pledge deposit*). Here, $Product_1$ ($Product_2$) and $Product\ index_1$ ($Product\ index_2$) represent the content and indexes of the original product (duplicate), respectively. After receiving such challenge requests, the ESPs will execute the challenge contract, which contains the following four steps:

- *Step 1: Fetch the proofs.* The proofs of both the original product and the duplicate will be fetched from local ledger. Recall that the proofs' format is (Sender, Receiver, Payload, Timestamp, Signature).
- *Step 2: Verify the identity of the challenger.* The ESPs verify challenger's signature in $Trans^{Chall}_{AIGC}$ using Receiver public key in the original product's proof. If signature verification passes, it can prove that the challenger is indeed the owner of the original product.
- *Step 3: Measure the similarity between the original product and the duplicate.* Firstly, ESPs conduct hash operations on $Product_1$ and $Product_2$ and check whether the hashes match $Product\ index_1$ and $Product\ index_2$, respectively. If so, they conduct the similarity measurement using three well-established metrics, namely image histogram, perceptual hash, and difference hash. These metrics are specifically designed for image similarity measurement. For instance, phash measures the image similarity by four steps [10]. Firstly, it preprocesses the input images, such as converting them to grayscale and unifying their resolution. Then, using discrete cosine transform and Wavelet transform, phash extracts the images' visual features such as edges, textures, and patterns into feature matrices. After that, two feature matrices are converted to two segments of binary hash code. Finally, phash calculates the Hamming distance between two hash segments and acquires the image similarity. The similarity measurement result is a tuple named *Similarity*, formed by (*histogram*, *phash*, *dhash*).
- *Step 4: Check the results.* If the similarity level exceeds the pre-defined thresholds in any two metrics, the challenge attempt can be regarded as successful. Otherwise, the challenge fails.

All ESPs execute this contract in parallel and then exchange the challenge results. Since the similarity measurement metrics are quantitative and deterministic, honest ESPs are guaranteed to acquire the consistent results. Then, the result goes through the consensus and is recorded on chain. Note that the metrics and the corresponding thresholds can be adjusted, according to the specific AIGC scenarios (e.g., text and videos). This can follow the Ethereum Improvement Proposal (EIP) procedure [12], that is, launching a proposal and waiting for the approval from the majority of nodes.

If the challenge succeeds, the ESPs will send a transaction $Trans^{Dere}_{AIGC}$ with the payload ($Product\ index_2$, *Pledge deposit*, *Similarity*). This transaction deregisters the duplicate and returns the pledge deposit provided by the challenger. In this case, the copyright of the challenger can be retrieved. Accordingly, the reputation of the plagiarist decreases, since the deregistration message will be spread on blockchain. If the challenge fails, the challenger will lose its deposit. Such a setting is to prevent attackers from launching spamming attacks, that is, initializing massive challenge requests. The locked deposit will be transferred to a system account and be used for incentivizing other participants, that is, the ESPs executing the challenge contracts.

## Incentive Mechanism

The economic system of AIGC is complicated because it accommodates different stakeholders, which conduct transactions with each other frequently. Thus, we should guarantee that: all the stakeholders can be incentivized to manage the AIGC lifecycle; the funds-AIGC ownership exchanges can be conducted legitimately without repudiation. To this end, an on-chain incentive mechanism is presented.

**One-Way Incentives:** One-way incentives are automatically issued to the ESPs, which maintain the ledger and provide blockchain services. Recall that our blockchain adopts delegated Proof-of-Stake as the consensus mechanism, where ESPs take turns to generate new blocks. During each round of block generation, the generator can pack a coinbase transaction with pre-defined value to reward itself.

**Two-Way Guarantee:** Recall that during the AIGC product lifecycle, there exist two-way exchanges between fund and product ownership. However, users might hesitate to conduct such exchanges, since they cannot guarantee that the other party will strictly follow its promise. To build mutual trust and encourage AIGC circulation, we design a two-way guarantee protocol using Hash Time Lock (HTL) as another part of our incentive mechanism. As a cryptographic protocol ensuring the execution atomicity, HTL can effectively protect anonymous online exchanges. Specifically, HTL is composed of two components, namely hash lock and time lock. Using the hash lock, the

two parities can construct a two-way payment channel, on which they lock the fund and product, respectively. When one participant successfully unlocks the channel, the unlocking of the other side will be triggered, ensuring the atomic exchange. Additionally, the time lock guarantees that all these steps are finished within the pre-defined period.

Take the two-way exchange happening in the AIGC generation phase as an example. In this case, the ESP grants the producer the ownership of its AIGC product, and the producer pays the pre-configured service fee. To do so, we implement a smart contract with two atomic operations named `lock` and `release`. As shown in Fig. 2, Part C, during the handshake process described previously, the producer creates a randomness $R$ and sends its hash $H(R)$ to ESP. When $Trans^{Gen}_{AIGC}$ is recorded on the blockchain, a corresponding contract instance $C_1$ will be created by ESP immediately. $C_1$ calls `lock` function to lock the ownership storing in $Trans^{Gen}_{AIGC}$ using $H(R)$. Only the one with R can release the lock. Meanwhile, the ESP sends a payment reminder to the producer. Receiving the bill, the producer sends a payment transaction with the payload (*Balance*), where *Balance* should be equal to the pre-configured service fee. Then, it also initializes its own contract instance $C_2$, which locks the fund in the payment transaction by $H(R)$. Up till now, both fund and AIGC ownership are on-chain.

Then, the atomic exchange between fund and AIGC ownership can be conducted. Firstly, the producer unlocks the AIGC ownership by calling `release` operation of $C_1$, with an input $R$. $C_1$ will check whether the hash of $R$ matches $H(R)$ and unlock the $Trans^{Gen}_{AIGC}$ if $H(R)$ is correct. Since such a process exposes $R$ to $C_1$, the owner of $C_1$, that is, ESP, can also release the fund locked by $C_2$ using $R$. To prevent participants from intentional delay, a time lock is implemented in the smart contract by defining the expiration timestamp. Consequently, if they fail to unlock the properties on time, the lock will become permanent and the corresponding transactions will be discarded. Clearly, such a protocol guarantees the atomic and timely executions of the exchange process.

### Effectiveness Analysis

Here, we analyze the effectiveness of the proposed mechanisms. Firstly, the proof generation phase of our Proof-of-AIGC registers every AIGC product on the blockchain. In this case, even though attackers can distribute massive fake messages, the consumers will not be misled since they can easily check the immutable registration record. Additionally, the challenge phase enables producers to defend AIGC plagiarization and retrieve their copyright. Note that the introduction of Proof-of-AIGC does not incur heavy overhead for blockchain. From the computation perspective, the metrics used for similarity measurement are widely adopted and not resource-intensive. For instance, the running of phash on a Core i7-7500 CPU and 16.0 GB RAM machine only takes 0.007 sec [13] As for communication costs, all participants execute the challenge contracts in parallel and only synchronize the results for one round, whose complexity is $\mathcal{O}(N)$ ($N$ represents the number of full nodes). The malicious attackers cannot disturb the challenge phase unless they can control over 50 percent of full nodes, which is impossible in practice. Furthermore, the edge deployment realizes the fast executions of the proposed mechanisms. Take Proof-of-AIGC as an example. Firstly, since ESPs are close to producers, the handshake during the proof generation phase can be completed quickly. Meanwhile, the low-latency end-edge communications also ensure the rapid transmission of the generated products. This is critical for video and music AIGC, where the products have large size. Finally, the incentive mechanism motivates ESPs and also encourages the AIGC circulation by guarantying the atomic funds-AIGC ownership exchanges. Next, we address the final concern, that is, the heterogeneity of ESPs.

## Reputation-Based ESP Selection

### Problem Statement

In edge AIGC, each producer can access multiple heterogeneous ESPs simultaneously. In this case, selecting a reliable ESP for the specific task becomes a problem. Traditionally, producers can select the most familiar ESP, that is, the one with which they have traded the most times, to minimize the potential risk. However, such strategies may lead to an imbalanced workload among ESPs, thus increasing the service latency on busy ESPs. Meanwhile, the computing resources of idle ESPs will be wasted.

Traditionally, producers can select the most familiar ESP, that is, the one with which they have traded the most times, to minimize the potential risk. However, such strategies may lead to an imbalanced workload among ESPs, thus increasing the service latency on busy ESPs. Meanwhile, the computing resources of idle ESPs will be wasted.

To this end, we implement a reputation-based ESP selection scheme in our framework. Specifically, it sorts all available ESPs according to their reputation, which is calculated by Multi-weight Subjective Logic (MWSL) [5]. We intend to achieve three goals:

- Helping producers select the most reliable ESP for each AIGC generation task
- Balancing the workload among multiple ESPs, thereby reducing the overall service latency
- Encouraging ESPs to complete the assigned tasks timely and honestly, since a negative reputation will directly affect their profits.

### Reputation Based on Multi-weight Subjective Logic

According to the Steps ①–③ of Fig. 2, Part D, producers select ESPs by three steps:

- Calculating the reputation of all available ESPs
- Sorting candidate ESPs according to their latest reputation
- Assigning the AIGC generation task to the ESP with the highest reputation.

Note that the item $ESP_1$ is marked red because it denies the service request. In this case, the producer traverses the reputation table and re-sends the request to the next candidate, that is, $ESP_3$. Next, we demonstrate the reputation calculation based on MWSL.

As shown in Fig. 3, MWSL utilizes the term "opinion" to denote the basic items for reputation calculation. The opinion is defined as a three-element vector $[p, n, u]$, where $p$ and $n$ represent the proportion of *positive* and *negative* interactions, respectively. Here, an interaction refers to the process from sending a service request, to confirming generation order, and to acquiring AIGC products. u (from 0 to 1) indicates the uncertainty between producer and ESP and is set manually according to the communication quality [5].

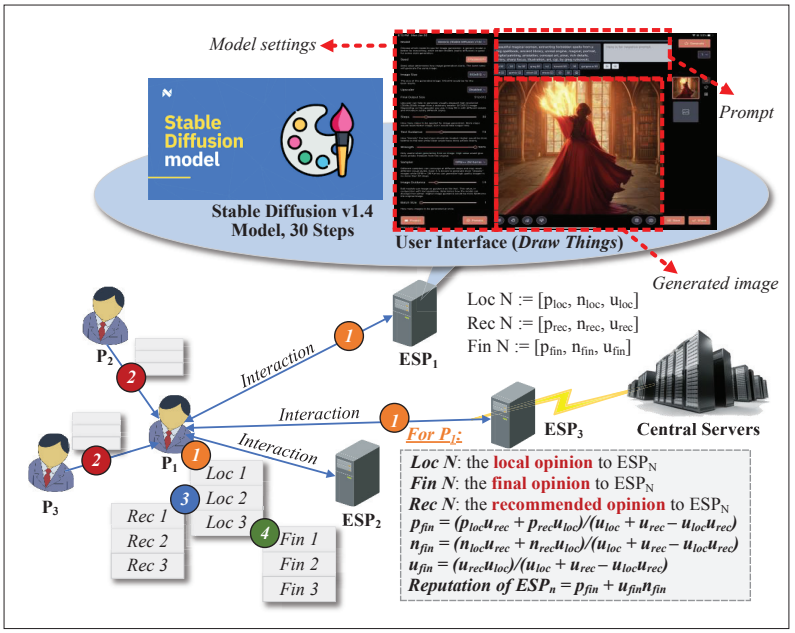


**FIGURE 3**. The reputation calculation process (from the perspective of producer $P_1$) and the illustration of AIGC services.

Suppose that our edge AIGC has three producers ($P_1$–$P_3$) and three ESPs ($ESP_1$–$ESP_3$). There exist three types of opinions in WMSL scheme. Firstly, for a given producer, say $P_1$, if it has direct interactions with these ESPs, $P_1$'s evaluation of them is called local opinions. Meanwhile, considering that $P_2$ and $P_3$ may also have the experience of interacting with these ESPs, their evaluation should also be considered. From the perspective of $P_1$, the evaluation of $ESP_1$-$ESP_3$ from $P_2$ and $P_3$ are called recommended opinions. Then, an overall opinion averaging all the recommended opinions from $P_2$ and $P_3$ can be generated. Moreover, since $P_2$ and $P_3$ have different familiarity degrees with ESPs, the weight of their recommended opinions is different. Finally, $P_1$ can combine the local opinion with the overall opinion, called final opinion. The detailed workflow for reputation calculation contains the following four steps (Steps ①–④ in Fig. 3):

- *Step 1: Generate local opinions.* Every producer updates its local opinion for every ESP.
- *Step 2: Synchronize information.* Producers share the latest local opinions. Assisted by blockchain, they can pack their opinions into transactions for secure sharing.
- *Step 3: Calculate overall opinion.* Each producer collects all received recommended opinions and averages them as the overall opinion. Note that the opinions are weighted before calculating the average. For any recommended opinion from $P_n$ to $ESP_n$, the weight is defined as the sum of $\alpha_1$ times Familiarity and $\alpha_2$ `times` *Value*. *Familiarity* indicates the number of historical interactions between $P_n$ and $ESP_n$. *Value* equals the total service fee for these interactions. Finally, $\alpha_1$ and $\alpha_2$ are two weighting factors whose sum is 1. Notably, the more interactions, the larger the weight.
- *Step 4: Calculate reputation.* Every producer combines its local opinion with overall opinion and calculates the final opinion [$p_{fin}$, $n_{fin}$, $u_{fin}$], as well as the reputation. The corresponding equations are shown in Fig. 3.

## Numerical Results

To prove the validity of the proposed methods, we implement a prototype of our AIGC product lifecycle management framework and deploy the reputation-based ESP selection on it (https://github.com/Lancelot1998/AIGCLifecycleManagement). As shown in Fig. 3, the testbed consists of three ESPs (served by Apple MacBook Pro with 8-Core Intel Core i9 CPU and AMD Radeon Pro 5500M GPU) and three producers (served by iPhones). The text-to-image AIGC services are supported by *Draw Things* application (https://drawthings.ai/). Factors $\alpha_1$ and $\alpha_2$ are set as 0.35 and 0.65, respectively. Additionally, u{loc} is fixed to 0.5 [5]. For each producer, it marks one interaction as "negative" if ESP fails to send the AIGC proof within the pre-confirmed time. The service quality, that is, the probability of receiving positive opinions, of $ESP_1$, $ESP_2$, and $ESP_3$ are 95 percent, 70 percent, and 55 percent, respectively. Finally, after acquiring the ESPs' reputation, producers can sample one ESP using the *Softmax* function.

Firstly, Fig. 4 illustrates the reputation trends of three ESPs. To validate the superiority of the MWSL-based reputation, we introduce a baseline named Traditional Subjective Logic (TSL) [5]. TSL is similar to MWSL, while it simply averages the recommended opinion rather than weighting them first. For our MWSL, we can observe that during the 1*st*–14*th* rounds, all ESPs accumulate reputation. Given the high service quality, the reputation of $ESP_1$ directly reaches to the top and stays stable, while $ESP_2$ and $ESP_3$ gradually increase their reputation by providing more positive interactions. From the 15*th* round, we let $ESP_1$ intentionally delay the AIGC services. Correspondingly, its reputation drops dramatically, since more negative interactions are reported. In contrast, since $ESP_2$ and $ESP_3$ acquire the chance to handle more tasks, their reputation keeps increasing. We conclude that the proposed reputation scheme can effectively quantify the trustworthiness of ESPs. In this way, the producers can easily judge which ESP is the most reliable. On the other hand, ESPs are also motivated to keep performing honestly. As for TSL, since the weight of the recommended opinions does not increase with the increasing *Familarity* and *Value*, the reputation drop of $ESP_1$ is much slower. In this case, although TSL-based reputation can also reflect the ESP's reliability, its timeliness is worse than our WMSL.

Then, Fig. 5 shows ESPs' workload under different ESP selection methods. Here, we suppose that all three producers request for AIGC services with the same frequency, and we let them randomly select ESPs during the first 5 rounds. From the 6th round, two ESP selection methods are tested, namely the traditional method and the proposed reputation-based method. Recall that traditionally, the producers tend to assign tasks to their most familiar ESPs. As a result, the workload among ESPs is imbalanced, causing long service latency. As shown in Fig. 5, most AIGC generation tasks congest in $ESP_3$, while the computing power of $ESP_2$ becomes wasted. Assisted by reputation, the producers can qualitatively evaluate the trustworthiness of ESPs and no longer need to rely on their empirical judgement. Consequently, the workload among ESPs is effectively balanced. Note that since there is no similar work regarding blockchain-empowered AIGC in the literature, we do not set a baseline to compare with.

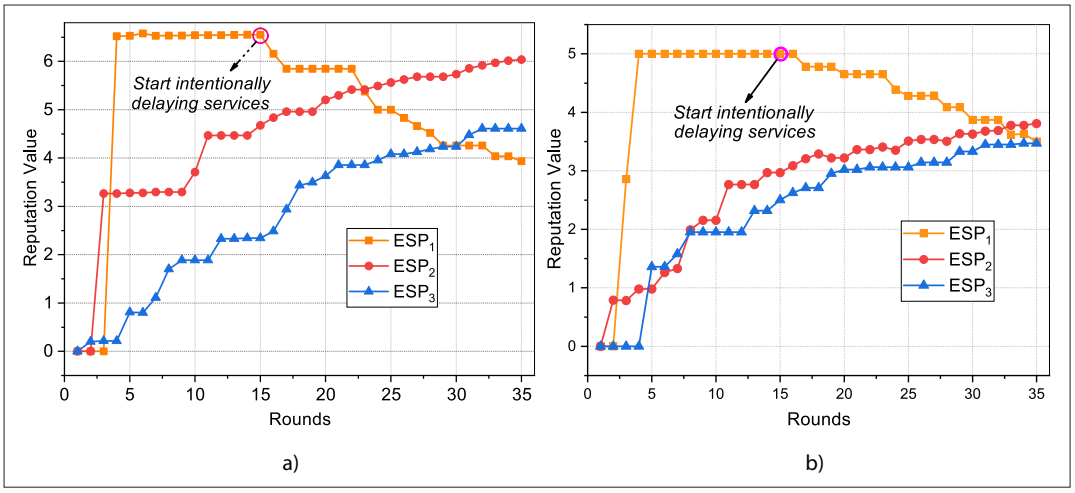


**FIGURE 4**. The reputation trends of three ESPs (from the perspective of a random producer), using WMSL and TSL: a) The MWSL scheme (ours); b) The TSL scheme.

Deepfake refers to synthetic media in which a person in an existing image or video is replaced with someone else's likeness. Given the security property of blockchain, it can help AIGC against deepfake.

# Future Direction

## Blockchain-Based AIGC Governance

The rapid development of AIGC greatly enriches the Internet content, but it also brings *deepfake* [14]. Deepfake refers to synthetic media in which a person in an existing image or video is replaced with someone else's likeness. Given the security property of blockchain, it can help AIGC against deepfake. For example, some distributed governance organization can be deployed on-chain, thus conducting the AIGC supervision and deepfake identification. However, since identifying deepfake requires off-chain knowledge, how to effectively bridge blockchain and physical AIGC is worth exploring.

## Distributed AIGC Model Training

This article focuses on the AIGC product lifecycle. The AIGC model construction lifecycle, including model training, fine-tuning, and inference, is also a meaningful research topic. For instance, since the training of diffusion models is time-consuming and resource-intensive, new algorithms and frameworks for building the distributed AIGC model training are worth studying. In this way, the computing power in the entire edge network can be exploited and thus significantly improve the training speed. Meanwhile, blockchain can be applied to protect the security of the training process and reward the users who contribute their resources fairly.

## Metaverse

AIGC is a building block for metaverse, since it can create numerous multimodal content for rendering immersive and realistic virtual worlds [15]. For example, the text-to-3D AIGC allows machines to collect the background, locations, and characters of users, thereby generating personalized avatars in the metaverse environment. Although such a process brings high QoE and immersiveness, some sensitive personal information might be leaked. Since blockchain has shown great strength in protecting data storage and sharing, the metaverse-oriented AIGC storage, access control, and sharing based on blockchain technique are also worth investigating.

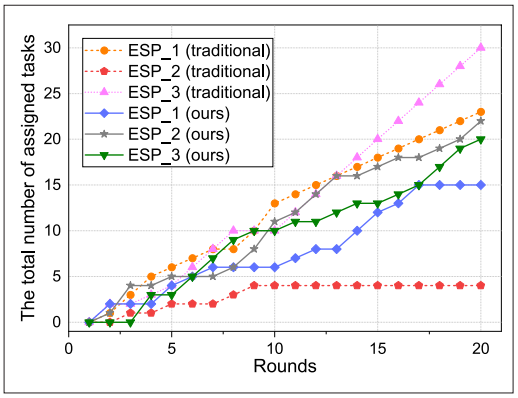


**FIGURE 5**. The total number of assigned tasks of three ESPs.

# Conclusion

In this article, we first introduce the lifecycle of AIGC products in edge networks and summarize four major concerns. To this end, we present a blockchain-empowered framework, realizing the secure and efficient AIGC product lifecycle management. Specifically, Proof-of-AIGC solves the ownership tampering and plagiarization of AIGC products. Additionally, we propose an incentive mechanism to encourage the AIGC circulation. Moreover, we design a MWSL-based reputation scheme to help producers select reliable ESPs. Numerical results demonstrate the reputation can not only reflect the reliability of ESPs, but also facilitate the workload rebalancing. Last but not least, we discuss future directions regarding the combination of blockchain and AIGC.


## Acknowledgment

This research is supported in part by NSFC under grant No. 62102099 and U22A2054, and is also supported in part by MOE Tier 1 (RG87/22), the National Research Foundation (NRF), Singapore and Infocomm Media Development Authority under the Future Communications Research Development Programme (FCP), and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-RP-2020-019), under Energy Research Test-Bed and Industry Partnership Funding Initiative, part of the Energy Grid (EG) 2.0 programme, and under DesCartes

and the Campus for Research Excellence and Technological Enterprise (CREATE) programme.

## References

[1] H. Du *et al.*, "Enabling AI-Generated Content (AIGC) Services in Wireless Edge Networks," 2023; available: https://arxiv.org/abs/2301.03220.
[2] Y. Sun *et al.*, "Travel with Wander in the Metaverse: An AI Chatbot to Visit the Future Earth," *Proc. 2022 IEEE 24th Int'l. Wksp. Multimedia Signal Processing*, 2022, pp. 1–6.
[3] Y. Liu *et al.*, "LightChain: A Lightweight Blockchain System for Industrial Internet of Things," *IEEE Trans. Industrial Informatics*, vol. 15, no. 6, 2019, pp. 3571–81.
[4] W.-J. Lai, C.-W. Hsueh, and J.-L. Wu, "A Fully Decentralized Timelock Encryption System on Blockchain," *Proc. 2019 IEEE Int'l. Conf. Blockchain*, 2019, pp. 302–07.
[5] J. Kang *et al.*, "Toward Secure Blockchain-Enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory," *IEEE Trans. Vehic. Tech.*, vol. 68, no. 3, 2019, pp. 2906–20.
[6] Y. Lin *et al.*, "A Unified Blockchain-Semantic Framework for Wireless Edge Intelligence Enabled Web 3.0," 2022; available: https://arxiv.org/abs/2210.15130.
[7] L. Floridi and M. Chiriatti, "GPRT3: Its Nature, Scope, Limits, and Consequences," *Minds and Machines*, vol. 30, no. 4, Apr. 2020, pp. 681–94.
[8] J. Sun *et al.*, "Metaverse: Survey, Applications, Security, and Opportunities," 2022; available: https://arxiv.org/abs/2210.07990.
[9] N. Karandikar, A. Chakravorty, and C. Rong, "Blockchain Based Transaction System With Fungible and Non-Fungible Tokens for a Community-Based Energy Infrastructure," *Sensors*, vol. 21, no. 11, 2021; available: https://www.mdpi.com/1424-8220/21/11/3822.
[10] Real-world examples of attacks, 2023; available: https://github.com/Lancelot1998/AIGCLifecycleManagement
[11] G. Xu, Y. Liu, and P. W. Khan, "Improvement of the DPOS Consensus Mechanism in Blockchain Based on Vague Sets," *IEEE Trans. Industrial Informatics*, vol. 16, no. 6, 2020, pp. 4252–59.
[12] Ethereum improvement proposal, 2023; available: https://ethereum.org/en/eips/.
[13] "Analysis of Perceptual Hashing Algorithms in Image Manipulation Detection," *Procedia Computer Science*, vol. 185, 2021, pp. 203–12.
[14] Y. Nirkin *et al.*, "Deepfake Detection Based on Discrepancies Between Faces and Their Context," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, 2022, pp. 6111–21.
[15] M. Xu *et al.*, "A Full Dive Into Realizing the Edgeenabled Metaverse: Visions, Enabling Technologies, and Challenges," *IEEE Commun. Surveys Tutorials*, 2022, pp. 1–1.

## Biographies

Yinqiu Liu (yinqiu001@e.ntu.edu.sg) received B.S. degree from Nanjing University of Posts and Telecommunications, China in 2020 and M.Sc. degree from the University of California, Los Angeles, USA in 2022. He is currently working toward the Ph.D. degree in the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests mainly focus on blockchain systems, Internet of Things, and generative models.

Hongyang Du (hongyang001@e.ntu.edu.sg) received the B.S. degree from Beijing Jiaotong University, Beijing, China, in 2021. He is currently working toward the Ph.D. degree in the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests mainly focus on semantic communications, generative AI, integrated sensing and communications, and Metaverse.

Dusit Niyato (dniyato@ntu.edu.sg) is currently a professor in the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He received the B.Eng. degree from King Mongkuts Institute of Technology Ladkrabang (KMITL), Thailand in 1999 and Ph.D. degree in electrical and computer engineering from the University of Manitoba, Canada in 2008. His research interests are in the areas of Internet of Things (IoT), machine learning, and incentive mechanism design.

Jiawen Kang (kavinkang@gdut.edu.cn) received the Ph.D. degree from the Guangdong University of Technology, China in 2018. He was a postdoc at Nanyang Technological University, Singapore from 2018 to 2021. He is currently a professor at Guangdong University of Technology, China. His research interests mainly focus on blockchain, security, and privacy protection in wireless communications and networking.

Zehui Xiong (zehuixiong@sutd.edu.sg) is an Assistant Professor at Singapore University of Technology and Design. Prior to that, he was a researcher with Alibaba-NTU Joint Research Institute, Singapore. He received the Ph.D. degree in Computer Science and Engineering at Nanyang Technological University, Singapore. He was a visiting scholar with Princeton University and University of Waterloo. His research interests include wireless communications, network games and economics, blockchain, and edge intelligence.

Chunyan Miao (ascymiao@ntu.edu.sg) received the B.S. degree from Shandong University, Jinan, China, in 1988, and the M.S. and Ph.D. degrees from Nanyang Technological University, Singapore, in 1998 and 2003, respectively. She is currently a professor in the School of Computer Science and Engineering, Nanyang Technological University, and the director of the Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly. Her research focuses on human agent interaction, human computation, and cognitive agents.

Xuemin Shen (sshen@uwaterloo.ca) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is a University Professor in Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on wireless communication networks, including capacity analysis, mobility and radio resource management, and so on.

Abbas Jamalipour (a.jamalipour@ieee.org) is the Professor of Ubiquitous Mobile Networking at the University of Sydney, Australia, and holds a Ph.D. in Electrical Engineering from Nagoya University, Japan. His research interests mainly focus on mobile networks, cellular networks, satellite communications, and wireless networking.