Energy-Efficient and Physical-Layer Secure Computation Offloading in Blockchain-Empowered Internet of Things

Yiliang Liu, Member, IEEE, Zhou Su, Senior Member, IEEE, and Yuntao Wang, Student Member, IEEE

Abstract—This article investigates computation offloading in blockchain-empowered Internet of Things (IoT), where the task data uploading link from sensors to a base station (BS) is protected by intelligent reflecting surface (IRS)-assisted physical-layer security (PLS). After receiving task data, the BS allocates computational resources provided by mobile-edge computing (MEC) servers to help sensors perform tasks. Existing blockchain-based computation offloading schemes usually focus on network performance improvements, such as energy consumption minimization (ECM) or latency minimization, and neglect the Gas fee for computation offloading, resulting in the dissatisfaction of high Gas providers. Also, the secrecy rate during the data uploading process cannot be measured by a steady value because of the time-varying characteristics of IRS-based wireless channels, thereby computational resources allocation with a secrecy rate measured before data uploading is inappropriate.

In this article, we design a Gas-oriented computation offloading scheme that guarantees a low degree of dissatisfaction of sensors, while reducing energy consumption. Also, we deduce the ergodic secrecy rate of IRS-assisted PLS transmission that can represent the global secrecy performance to allocate computational resources. The simulations show that the proposed scheme has lower energy consumption compared to existing schemes and ensures that the node paying higher Gas gets stronger computational resources.

Index Terms—Blockchain, gas oriented, intelligent reflecting surface (IRS), Internet of Things (IoT), physical-layer security (PLS).

I. INTRODUCTION

WITH the rapid development of wireless communications and sensor manufacturing technologies, the narrowband-enabled Internet of Things (IoT) technologies, released by 3GPP, have gained increasing attention in industrial, medical, business fields, etc. [1]. Following that, the security issues of heterogeneous devices and infrastructures have raised many concerns. Building on their characteristics of decentralization, lightweight, and multimoding, the blockchain and physical-layer security (PLS) are regarded as enabling endogenous security technologies to simultaneously address trust and confidentiality issues of IoT networks [2]–[4].

The typical application of the blockchain-empowered IoT is the computation offloading of the tasks of sensor nodes. Smart contracts perform the automated and trust-free offloading process between sensor nodes and mobile-edge computing (MEC) servers on top of the blockchain to ensure that transactions cannot be denied and malicious behaviors will also be traced [5]–[7]. However, existing IoT computation offloading schemes have the following two challenges. First, traditional schemes usually focus on the optimization of computation or communication performance, such as computing latency minimization, energy consumption minimization (ECM), or network throughput maximization [3], [8]–[10]. The Gas factor only affects the block generation speed and is not considered in computational resource allocation, leading to the dissatisfaction of sensors as better computational resources are not allocated to those sensors under the performance priority strategy even if they pay high Gas. Second, the coverage of the central trusted authority (TA) is limited in dynamic and pervasive IoT networks [11], [12]. Hence, the security of links from the sensors to the MEC servers cannot be protected effectively. In addition, due to the cost constraint of sensor devices, traditional PLS schemes, such as multiple-antenna beamforming or artificial noise technologies [13], are not suitable for IoT networks.

To address these problems mentioned above, we present an intelligent reflecting surface (IRS)-assisted secure computation offloading scheme to guarantee that the node paying higher Gas has more opportunities to get a stronger computational resource, while reducing the energy consumption of the computation offloading process. The main contributions are summarized as follows.

1) We formulate a PLS-assisted computation offloading model in blockchain-empowered IoT, where phase-shift matrix adjustment of IRS devices, multiple-antenna technologies at the base station (BS), and the allocation of computational resources of MEC servers with dissatisfaction thresholds of sensors are jointly considered.

$1$Gas refers to the cost necessary to perform transactions on Ethereum, where Gas price has the unit ETH or Gwei. The fee of a transaction can be calculated by the amount of Gas $\times$ Gas price. Gas price has a transformation to legal tenders. For instance, a transaction needs 100 Gwei with $6.5 \times 10^{-7}$ dollar/Gwei.
to reduce the energy consumption of the computation offloading process.

2) We propose a phase-shift matrix adjustment scheme via manifold optimization to improve the secrecy rate. Then, we deduce the expression of the ergodic secrecy rate of this adjustment scheme, which provides a global metric of secrecy performance for computational resource allocation.

3) We transform the original computational resource allocation problem into a 2-D matching problem of bipartite graphs, which is solved by the Kuhn–Munkres (KMs) algorithm. Especially, the edges of the bipartite graph are restricted with dissatisfaction degrees of sensors, so the optimal matching achieves the goal that the sensors with higher Gas payment are prioritized with better computational resources. After computational resource allocation, we propose an adaptive PLS coding method to improve the effective secrecy rate during the data uploading process.

The remainder of this article is organized as follows. Section II surveys the related works. Section III describes the system model and problem formulation. The IRS-assisted PLS and computation offloading schemes are proposed in Sections IV and V, respectively. We show simulation results in Section VI and conclude this article in Section VII.

Notations: Bold uppercase letters, such as $A$, denote matrices, and bold lowercase letters, such as $a$, denote column vectors. $A^H$, $A^T$, and $A^\dagger$ represent the conjugate transpose, transpose, and conjugate transformation of $A$, respectively. $I_a$ is an identity matrix with its rank $a$. $CN(\mu, \sigma^2)$ is a complex normal (Gaussian) distribution with mean $\mu$ and variance $\sigma^2$. $|x|$ is the Euclidean norm of $x$. diag$(X)$ is the diagonal matrix of $X$. $\mathbb{E}(\cdot)$ is the expectation operation. arg$(x)$ is the angle of complex variable $x$. $\Re(x)$ means the real part of complex variable $x$.

II. RELATED WORK

A. Blockchain-Empowered IoT

The blockchain technology is now integrated into the IoT networks to ensure trust and traceability of node behaviors. Despite the benefit that blockchain brings to traditional IoT applications, there are still many challenges in its actual implementation, such as blockchain data storage, power consumption, and task offloading. To cope with these issues, many efforts to adopt blockchain in IoT applications have been conducted in recent years.

1) Data Storage and Prototype: Yu et al. [14] presented a multiple-layer storage architecture for large-scale IoT networks to meet the storage requirement of massive transaction data of blockchain. Furthermore, Pyoung and Baek [15] proposed finite lifetime blocks to reduce storage costs in IoT, where outdated transactions and blocks can be safely removed from the blockchain. Pan et al. [16] designed the prototype of an edge IoT resource management system based on blockchain and smart contracts, where all the IoT activities and transactions are recorded on the blockchain for security purposes.

2) Computation Offloading: In this case, Xu et al. safeguarded data integrity during real-time data processing in blockchain-empowered IoT [17], while considering the imbalances of computational resource distribution and workload. Due to the computation and storage constraints of mobile IoT devices in blockchain-enabled applications, Liu et al. [18] offloaded compute-intensive Proof-of-Work (PoW) mining missions to nearby edge servers. Chen et al. [19] further investigated joint computation offloading of mining tasks and data processing tasks for Industrial IoT (IIoT) devices and used Nash equilibrium of the game theory to minimize their economic costs. Feng et al. [20] and Nguyen et al. [21] designed a novel deep reinforcement learning algorithm to improve the communication and computation performance during the entire computation offloading process. Considering the cost of resources and income of the service provisioning, Fan et al. [22] used the Stackelberg game where computation resource providers are set as the leader, and IoT users are followers. The objective of Fan’s scheme is to optimize the revenues of computation resources. Seng et al. [23] proposed a matching algorithm between IoT users and edge servers by considering task execution time and energy consumption, and the matching algorithm is implemented by smart contracts without relying on central nodes. Our work focuses on the confidentiality of the task data uploading process via PLS schemes and the Gas effect on the computational resource allocation, which are not discussed in previous researches.

B. IRS-Assisted PLS Technologies

IRS devices are usually applied to satellite communications and radars but are not widely used in the mobile communication field because of cost and fabrication constraints. Thanks to the development of new material technologies, IRS has gradually been used in mobile communications and networks to improve signal quality and security performance. In the area of IRS-assisted PLS, many investigations are conducted to improve the secrecy rate, such as the semidefinite relaxation (SDR)-based and majorization minimization (MM)-based phase-shift matrix optimization methods [24], [25]. However, there are very few researches on expressions of security performance metrics. Yang et al. [26] and Trigui et al. [27] considered the IRS-assisted single-input–single-output single-antenna eavesdropper (SISOSE) case and gave its expressions of ergodic secrecy rate and secrecy outage probability. For multiple-input–single-output multiple-antenna eavesdropper (MISOME) scenarios, the Monte Carlo method is used to obtain the ergodic secrecy rate with a lot of computational overhead [28]. This article considers the uplink communications of IoT with a passive multiple-antenna eavesdropper, which obeys IRS-assisted single-input–multiple-output multiple-antenna eavesdropper (SIMOME), and its ergodic secrecy rate expression is not investigated in existing works.

C. Discussion

The Gas factor is not considered in existing investigations of computation offloading in blockchain-empowered IoT networks. Besides, the absence of expressions of ergodic
contracts, including three functional layers [29], [30]. 

open source, and public blockchain platform supporting smart and follow the rules of Ethereum. Ethereum is a decentralized, 
sensors to the BS. 

is deployed to enhance the security of the uplink channel from 
scale fading effect of sensor communications, an IRS device 
ensuring the fairness among sensors. 

cation scheme that can reduce the energy consumption, while 
data uploading lack optimization objectives and parameters. In 
SIMOME results in that computational resource allocation and 
secrecy rate and effective secrecy rate of IRS-assisted SIMOME results in that computational resource allocation and data uploading lack optimization objectives and parameters. In this work, we first design a manifold-based phase-shift matrix adjustment scheme for IRS-assisted PLS. Then, we provide its expressions of ergodic secrecy rate and effective secrecy rate, followed by a Gas-oriented computational resource allocation scheme that can reduce the energy consumption, while ensuring the fairness among sensors.

III. SYSTEM MODEL AND PROBLEM FORMULATION

This article considers a blockchain-empowered IoT network, as shown in Fig. 1, which includes \( N_I \) sensor nodes, denoted by \( \{ U_1, U_2, \ldots, U_{N_I} \} \). Each sensor is equipped with one antenna. A set of \( N_K \) MEC servers, denoted by \( \{ M_1, M_2, \ldots, M_{N_K} \} \), are associated with an \( N_K \)-antenna BS via wired links, and \( N_K \geq N_I \). Due to the transmission power constraints and the large-scale fading effect of sensor communications, an IRS device is deployed to enhance the security of the uplink channel from sensors to the BS.

All sensors and MEC servers are registered in Ethereum and follow the rules of Ethereum. Ethereum is a decentralized, open source, and public blockchain platform supporting smart contracts, including three functional layers [29], [30].

1) Underlaying Layer: It mainly involves the transmission and storage mechanisms of transaction and block data, as well as the blockchain-based cryptographic algorithms, such as hash and certificates.

2) Core Layer: It includes incentive, consensus, and blockchain topology mechanisms. The incentive mechanism is based on Ether, which introduces economic incentives to make the nodes give their efforts to verify data in Ethereum. The consensus mechanism achieves information synchronization among untrusted parties in decentralized systems, such as PoW, Proof of Stake (PoS). The blockchain topology mechanism involves transactions, mining, block verification, Gas, etc., where Gas is the cost necessary to perform transactions on Ethereum.

3) Application Layer: It involves script codes, smart contracts, and other programmable codes that can be used to enable diverse Ethereum transactions.

A. Ethereum-Assisted Computation Offloading Architecture

As the computational resources of sensors are scarce, these \( N_I \) sensors offload their computational tasks, denoted by \( \{ Z_1, Z_2, \ldots, Z_{N_I} \} \), to MEC servers via the wireless channels between sensors and the BS. After receiving tasks, the BS allocates virtual machines provided by MEC servers to execute these tasks. To record these tasks, sensors store the indices of publishing tasks on Ethereum via a contract function, i.e., `task_publish_contract()` , and the BS stores the indices of results on Ethereum via the contract function `result_upload_contract()` . Once the blocks of these contracts are deployed on Ethereum, no one can repudiate the debts of transactions, and the results given by MEC servers are recorded to defend against malicious MEC servers. Our work adopts the Ethereum-assisted computation offloading architecture presented in [31], which is described as follows.

1) Transmitting Tasks to the BS: The sensor \( U_i \) sends the task data to the BS via IRS-assisted PLS schemes to resist eavesdropping attacks. To avoid interuser interference, sensors use the time-division multiple-access (TDMA) technology during transmission processes.

2) Creating Contracts of Publishing Tasks: At the same time of task transmission, by following the contract format of Ethereum, the sensor \( U_i \) launches a contract, i.e., \#1 = `task_publish_contract(Z_i)` , which includes the hash message corresponding to the offloaded task \( Z_i \) and the signature of \( U_i \). Also, the contract has the address information of the transaction parties,\(^2\) Gas for this transaction defined by \( V_i \), Gas price, etc. Then, the contract \#1 is broadcast in Ethereum.

3) Computing Tasks in MEC Server: After receiving tasks, the BS allocates computational resources of MEC servers to perform these tasks. To avoid the complexity caused by tasks and computing power partition, it is assumed that a task uses one MEC server to compute results and a MEC server can be allocated to only one task. Finally, the results are uploaded to the core server for subsequent services.

\(^2\)The sending address is related to \( U_i \), and the received address is the constructor of the contract as the contract should be broadcast in Ethereum.

![Fig. 1. Secure computation offloading in IoT networks, where the sensor \( U_i \) sends computation tasks to the BS via IRS and instantaneously broadcasts the task publishing contracts (denoted by #1). After receiving tasks, the BS allocates computational resources of MEC server \( M_k \) to complete the tasks, then \( M_k \) broadcasts the result uploading contracts (denoted by #2). Both task publishing and result uploading should be recorded in blocks via the #1 block generation and #2 block generation, respectively. The uplink links from sensors to the BS are protected by an IRS-assisted PLS scheme.]
4) Generating Blocks of Contracts of Task Publishing: All nodes in Ethereum synchronize the transaction of task_publish_contract($Z_i$) and check its format and signature. If passing, Ethereum nodes compete with each other to win the right of charging the account of the transaction, then the block is generated in Ethereum by the winners. All members registered in Ethereum designing to get payments can take part in the competition of account-charging rights.

5) Creating Contracts of Result Uploading: Without loss of generality, we assume that the task of $U_i$, i.e., $Z_i$, is offloaded to the MEC server $M_k$. After completing the task, $M_k$ launches a contract, i.e., result_upload_contract($Z_i$), which includes the hash message of results, the signature of $M_k$, allocated computational resources, address information, Gas for this transaction, Gas price, etc.

6) Generating Blocks of Contracts of Result Uploading: Similar to the block generation of contracts of task publishing, all members registered in Ethereum can take part in this competition of account-charging right. The winner can obtain Gas as a payment.

In this system, the IoT service acquirer should pay sensors and MEC servers as the original data are generated by sensors and the data process is done by MEC servers. In this case, an IoT-enabled business model is established.

B. Communication Model

The article considers an IRS-assisted uplink communication model with a passive $N_e$-antenna eavesdropper (Eve), an $N_s$-antenna BS, $N_t$ single-antenna sensors, and an IRS device. The IRS device is equipped with $N$ programmable phase-shift elements. All channels are assumed to obey Rayleigh fading, i.e., the channel from $U_i$ to Eve is defined as $g_i \sim CN_{N_e,1}(0, I_{N_e})$, the wiretap channel aimed at $U_i$ from IRS to Eve is defined as $Z_i \sim CN_{N_e,N_s}(0, I_{N_e} \otimes I_{N_s})$, the direct link from $U_i$ to BS is defined as $l_i \sim CN_{N_t,1}(0, I_{N_t})$, the channel from $U_i$ to IRS is defined as $h_i \sim CN_{N_t,1}(0, I_{N_t})$, and the channel from IRS to BS for the transmission of $U_i$ is defined as $A_i \sim CN_{N_s,N_e}(0, I_{N_s} \otimes I_{N_e})$. The instantaneous CSIs of legitimate devices, including $l_i$, $h_i$, and $A_i$, can be perfectly obtained via channel estimation, whereas the instantaneous Eve’s CSIs $g_i$ and $Z_i$ are unknown.

For the security purpose, IRS controls programmable phase-shift elements via a phase-shift matrix, where the phase-shift matrix for the transmission period of $U_i$ is defined as an $N \times N$ matrix $\Phi_i$, i.e.,

$$\Phi_i = \text{diag}[^{\sqrt{-1}} \exp(\theta_{1,i}), \ldots, \exp(\theta_{n,i}), \ldots, \exp(\theta_{N,i})]$$

(1)

where $j = \sqrt{-1}$, and $\theta_{i,n} \in [0, 2\pi)$ is the phase introduced by the $n$th phase-shift element of IRS at the $i$th period. With the phase-shift matrix $\Phi_i$, the received signals at the BS and Eve can be expressed as follows:

$$y_i = a_i(l_i + A_i \Phi_i h_i)x_i + n_i$$

(2)

$$y_{e,i} = a_{e,i}(g_i + Z_i \Phi_i h_i)x_i + n_{e,i}$$

(3)

where $a_i$ is the path loss between the BS and $U_i$, $a_{e,i}$ is the path loss between the BS and Eve, $x_i$ is the confidential information-bearing signal from $U_i$ with $E(|x_i|^2) = P_i$, and $P_i$ is the transmission power of $U_i$. $n_i$ and $n_{e,i}$ are the additive white Gaussian noise (AWGN) obeying $CN_{N_b,1}(0, \sigma^2 I_{N_b})$ and $CN_{N_e,1}(0, \sigma^2 I_{N_e})$, respectively.

C. Energy Consumption Model

The sensor $U_i$ has the computation task $Z_i$ with $D_i$-bits data that should be uploaded to the BS. After receiving the task data, the BS will select a MEC server, such as $M_k$, to perform the task of $U_i$. In this model, when $D_i$-bits data should be calculated in $M_k$, the computing period is $c_kD_i/f_k$, where $c_k$ (CCN/bit) is the CPU cycle number (CCN) per bit processing at $M_k$, representing computing efficiency of CPU chips, and $f_k$ (CCN/s) is the CCN per second of $M_k$. The energy consumption per second of $M_k$ is $\eta_kf_k^3$ (Joule/s) [10], where $\eta_k$ is the computation energy efficiency coefficient of CPU chips in $M_k$ depending on the chip architecture. The transmission delay of uploading can be expressed as $D_i/(R \cdot B)$, where $R_i$ (bit/s/Hz) is the secrecy rate of uplink channels, and $B$ is the bandwidth. The energy consumption of $Z_i$ with the assistance of $M_k$ can be formulated as follows:

$$Q_{i,k} = \eta_kc_kD_i f_k^2 + \frac{D_i P_i}{R_B}.$$  

(4)

Note that the computing modules of Ethereum are worldwide distribution and are independent of the computation offloading system, so the energy consumption of Ethereum is not considered here.

D. Problem Formulation

In traditional Ethereum systems, Gas payment affects the time and energy consumption of block generation, but has no effect on computational resource allocation. In real scenarios, sensors paying higher Gas have more desire to get better computational resources, rather than recording their behaviors on Ethereum. Hence, when $U_i$ is allocated $M_k$, we define the degree of dissatisfaction of $U_i$ as follows:

$$O_i = W_{i,k}r(V_i) - r\left(\frac{f_k}{c_k}\right)$$

(5)

where $W_{i,k}$ is the element of the $i$th row and the $k$th column of $W$. $W_{i,k}$ is a binary variable taking 1 when $M_k$ is assigned to $U_i$, and 0 otherwise. $r(V_i)$ is the index of the descending order of Gas set provided by $N_t$ sensors, and $r(f_k/c_k)$ is the index of the descending order of computing power of $N_k$ MEC servers.

With the consideration of the energy consumption and degrees of dissatisfaction, the problem of the computation offloading focuses on the ECM with constraints as follows:

$$\pi: \min_{\Phi_i \forall_i, W} \sum_{i=1}^{N_t} \sum_{k=1}^{N_k} W_{i,k}Q_{i,k}$$

s.t. $O_i \leq \epsilon \quad \forall i$

$W_{i,k} = \{0 \text{ or } 1\} \quad \forall i, k$

$\sum_{i=1}^{N_t} W_{i,k} \leq 1, \quad \sum_{k=1}^{N_k} W_{i,k} \leq 1$

Eq. (1)
where (7) requires that the degrees of dissatisfaction should be smaller than a threshold $\epsilon$ for all sensors, and $\epsilon \in [0, 1, \ldots, N_T - 1]$ can be adjusted manually. Equation (8) is the binary constraint representing the allocation factor. Equation (9) reveals that each sensor can be allocated with only one MEC server, and each MEC server is only assigned to one sensor. Equation (10) is the passive phase-shift matrix constraint.

It is obvious that $\Phi_i$ and $W$ are coupled variables in the optimization of the $i$th period, and $P1$ is a nonconvex mixed-integer problem. To tackle the problem, we transform $P1$ into two subproblems, i.e., phase-shift optimization and computational resource allocation, then solve them step by step.

IV. PHASE-SHIFT OPTIMIZATION FOR PLS AND ITS ERGODIC SECRECY RATE

From (4), we find that the energy consumption $Q_{lk}$ decreases with the increasing secrecy rate $R_i$ for all sensors. To reduce the energy consumption in the data uploading process, we optimize $\Phi_i$ to maximize $R_i \forall i$.

A. Phase-Shift Optimization

According to (2) and (3), the secrecy rate $R_i$ between $U_i$ and BS is given as follows [13]:

$$R_i = \left(C_{m,i} - C_{w,i}\right)^+ \quad (11)$$

where

$$C_{m,i} = \log_2 \left(1 + \frac{\sigma_i^2 P_i}{\sigma_i^2} |l_i + A_i \Phi_i h_i|^2\right) \quad (12)$$

$$C_{w,i} = \log_2 \left(1 + \frac{\sigma_i^2 P_i}{\sigma_i^2} |g_i + Z_i \Phi_i h_i|^2\right). \quad (13)$$

The objective is to find the optimal $\Phi_i$ to achieve $\{R_i^+, \Phi^+\} = \max_{\Phi} R_i$. However, it is hard to maximize $R_i$ as $g_i$ and $Z_i$ are unknown, so the objective is transformed to maximize the channel gain between $U_i$ and the BS as follows:

$$P2: \max_{\Phi} |l_i + A_i \Phi_i h_i|^2 \quad (14)$$

s.t. Eq. (1). \quad (15)

P2 is a classic Boolean quadratic program problem. The traditional methods to solve P2 are the SDR technique [32] and manifold optimization [33]. The objective parameter $\Phi_i$ is the middle of two CSI parameters and is hard to handle. Hence, refer to [33], $A_i \Phi_i h_i$ can be reformulated as follows:

$$A_i \Phi_i h_i = A_i \text{diag}(h_i) q_i = \Theta_i q_i \quad (16)$$

where $q_i = \text{vec}(\Phi_i) = \left[\exp(j\theta_{i1}), \ldots, \exp(j\theta_{iN})\right]^T$. With (16), the objective of P2 can be transformed as follows:

$$|l_i + A_i \Phi_i h_i|^2 = q_i^H \Theta_i^H \Theta_i q_i + |l_i|^2 + l_i^H \Theta_i q_i + q_i^H \Theta_i^H l_i. \quad (17)$$

Then, the minus of the objective in P2 is expressed as a function with respect to $q_i$, i.e.,

$$f(q_i) = -\left(q_i^H \Theta_i^H \Theta_i q_i + |l_i|^2 + l_i^H \Theta_i q_i + q_i^H \Theta_i^H l_i\right). \quad (18)$$

The constraint (15) is defined by a complex circle manifold [33], which is given by the following equation:

$$O = \left\{q_i \in \mathbb{C}^N : |[q_i]^n|^2 = 1, n = 1, \ldots, N \right\} \quad (19)$$

where $[q_i]^n$ is the $n$th element of $q_i$. It is obvious that (19) is equivalent to constraint (15), and the optimal point of original problem P2, i.e., $q_i^*$ can be found on the complex circle manifold $O$. Hence, P2 is equivalently transformed as follows:

$$P3: \min_{q_i \in O} f(q_i). \quad (20)$$

P3 is regarded as a complex circle manifold optimization problem. To find $q_i^*$, we use the Riemannian-gradient descent algorithm presented in [33], where the Riemannian gradient of a manifold point is decided jointly by the Euclidean gradient and the tangent space on manifold $O$ at this point. Without loss of generality, the $s$th point (the $s$th iteration in gradient descent) of manifold $O$ is $q_i(s) \in O$, and the tangent space for $O$ at the $s$th point can be expressed as follows:

$$T_{q_i(s)}O = \left\{v \in \mathbb{C}^N : \Re(v \circ q_i(s)^\dagger) = 0\right\} \quad (21)$$

where $v$ is the tangent vector at $q_i(s)$, $\circ$ is the Hadamard product, $\Re(\cdot)$ means the real part of a complex variable, and $(\cdot)^\dagger$ is the conjugate operation. The objective function $f(q_i)$ with respect to $q_i$ is defined in (18), and the Euclidean gradient at $q_i(s)$ is as follows:

$$\nabla_{q_i(s)} f(q_i) = -2\Theta_i^H \Theta_i q_i - 2\Theta_i^H l_i. \quad (22)$$

Among all tangent vectors on $T_{q_i(s)}O$, the one that yields the fastest increase of the objective function is defined as the Riemannian gradient [33], i.e.,

$$\text{grad}_{q_i(s)} f(q_i) = \nabla_{q_i(s)} f(q_i) - 2\Re\left(\nabla_{q_i(s)} f(q_i) \circ q_i(s)^\dagger\right) \circ q_i(s). \quad (23)$$

In the Riemannian-gradient descent process, the next point of $q_i(s)$ is with the direction of $s_\tau p_s$, where $s_\tau$ is the step size and $p_s$ is the descent direction vector. However, the movement cannot guarantee that the next point $q_i(s+1)$ is on manifold $O$. Hence, we introduce the retraction function to map a vector on $T_{q_i(s)}O$ onto manifold $O$, which is given as follows:

$$q_i(s+1) = R_{q_i(s)}(s_\tau p_s) \quad (24)$$

where a typical retraction is the normalization function, i.e., $R_X(y) = (y/|y|) \forall i$. Based on the Riemannian gradient and retraction function, we use the conjugate-gradient descent method to find the optimal phase-shift matrix, which is shown in Algorithm 1. In the conjugate-gradient descent algorithm, the update rule for the search direction on manifolds is given by the following equation:

$$p_{s+1} = -\text{grad}_{q_i(s+1)} f(q_i) + \varphi_s T_{q_i(s)\rightarrow q_i(s+1)}(p_s) \quad (25)$$

where $\varphi_s$ is the Polak–Ribiere parameter, and $T_{q_i(s)\rightarrow q_i(s+1)}(p_s)$ is the mapping function of the tangent vector $p_s$ from the tangent space $T_{q_i(s)}O$ to the tangent
Algorithm 1: Phase-Shift Matrix Optimization Based on Conjugate-Gradient Descent

Data: \{\mathbf{h}, A_i, \mathbf{h}_i, \mathbf{w}_i\}, N, N_I
Result: \Phi_f^*, \psi_I

1 for \( i = 1 : 1 : N_I \) do
2 Initialize \( \sigma_0 \) and \( \psi_0 \);
3 Initialize point \( q_i(0), \) get \( \nabla q_i(0) f(q_i) \);
4 while \( |\nabla q_i(s) f(q_i)| \leq \psi \) do
5 Get Riemannian gradient \( \nabla q_i(s) f(q_i) \) by Eq. (23);
6 Get conjugate search direction \( p_s \) by Eq. (25);
7 Get next point \( q_i(s + 1) \) by Eq. (24);
8 Determine step size \( \sigma_s \) and \( \psi_s \) proposed in [35];
9 Get \( q_i^* = q_i(s) \);
10 \( \Phi_f^* = \text{diag}(q_i^*) \);
11 Procedure End

Space \( T_{q_i(s+1)} \). The mapping function is given as follows:

\[
T_{q_i(s)} \rightarrow T_{q_i(s+1)}(p_s) = T_{q_i(s)} \rightarrow \Phi_i \circ p_s \circ \Psi_i(q_i(s+1)) \circ q_i(s+1). \tag{26}
\]

Since the second derivative of \( f(q_i) \), i.e., \( \nabla^2 f(q_i) = -2\Theta_i^H \Theta_i \) and \( \Theta_i^H \Theta_i \) is positive semidefinite, \( f(q_i) \) is concave, such that Algorithm 1 based on conjugate-gradient descent cannot be guaranteed to converge to the optimal point [34]. Hence, the output \( \Phi_f^* \) in Algorithm 1 is the local optimal result.

The computational complexity of Algorithm 1 is analyzed as follows. To meet the convergence condition, the conjugate-gradient descent algorithm requires \( N^2 \) iterations [36]. Each iteration needs the calculations of (23)–(25), and the parameter update, i.e., step 8, which requires \( 2N^2 + 4N, N, N, \) and \( 4N^2 \) inner iterations, respectively. Step 10 needs \( N \) iterations to generate \( \Phi_f^* \). For \( N_I \) sensors, the number of total iterations of Algorithm 1 is \( N_I(4N^4 + 6N^3 + N) \).

B. Ergodic Secrecy Rate

The secrecy rate can be viewed as the inherent property for a given instantaneous CSI realization. The ergodic secrecy rate is the average value of secrecy rates among all realizations of CSIs, which is a usual metric for PLS schemes [3], [13]. As the CSIs of wiretap channels are unknown, to calculate the ergodic secrecy rate, the gain of wiretap channels in (3), i.e., \( \|g_i + \mathbf{Z}_i \Phi_f \mathbf{h}_i\|^2 \), can be regarded as a random variable \( X \). The Gamma distribution can deduce the probability density function (PDF) of \( X \), as shown in Lemma 1. With the PDF of \( X \), the expression of ergodic secrecy rate can be derived as security performance metrics for computational resource allocation. The Gamma distribution can represent the sum of multiple-independent exponentially distributed random variables, so it is suitable for representing the gain of IRS-assisted channels, such as [37].

Lemma 1: For an \( N \times N \) fixed \( \Phi_f \) generated by (1), three independent random variables \( g \sim \mathcal{CN}_{1,N}(0, \mathbf{I}_N) \), \( h \sim \mathcal{CN}_{N,1}(0, \mathbf{I}_N) \), and \( Z \sim \mathcal{CN}_{N_e, N}(0, \mathbf{I}_N \otimes \mathbf{I}_N) \), if we have a random variable \( x \) as follows:

\[
x = |g + \mathbf{Z}_i \Phi_f \mathbf{h}_i|^2. \tag{27}
\]

The PDF and cumulative distribution function (CDF) of \( X \sim X(N_e, N, \Phi_f) \) can be expressed as follows:

\[
f_x(x) = \frac{1}{\Gamma(\mu) \nu^{\mu}} x^{\mu-1} \exp(-x/\nu) \tag{28}
\]

\[
F_x(x) = 1 - \frac{1}{\Gamma(\mu) \Gamma(\mu/\nu)} \tag{29}
\]

\[
\mu = \frac{N_e(1 + N)^2}{(1 + N)^2 + (1 + N)N} \tag{30}
\]

\[
\nu = 1 + N + \frac{(1 + N)N}{1 + N} \tag{31}
\]

where \( \Gamma(\cdot) \) is the Gamma function, and \( \gamma(\epsilon, \eta) \) is the incomplete gamma function defined as follows:

\[
\gamma(\epsilon, \eta) = \int_0^\epsilon \exp(-z) z^{\eta-1} \, dz. \tag{32}
\]

Proof: See Appendix A.

In order to examine the fitting performance of Gamma distributions, we use the KS method to find the maximum divergences between the actual distribution and the Gamma distribution [38], where statistic \( D_p \) is defined as the maximum divergences of PDF tests as follows:

\[
D_p = \max \left| f_X(x) - \hat{f}_X(x) \right| \tag{33}
\]

where \( \hat{f}_X(x) \) is the actual PDF of \( X \sim X(N_e, N, \Phi_f) \) obtained by Monte Carlo simulations. As shown in Fig. 2, \( D_p \) is with an order of magnitude around \( 10^{-4} \), so the Gamma distributions are in a good agreement with the actual distributions.

With Lemma 1, we deduce the expression of ergodic secrecy rate with optimal phase-shift matrix \( \Phi_f^* \). The ergodic secrecy rate is defined as follows [13]:

\[
R_I = \left[ \mathbb{E}(C_m, |\Phi_f^*|) - \mathbb{E}(C_w, |\Phi_f^*|) \right] + \leq \mathbb{E} \left[ (C_m - C_w)^|\Phi_f^*| \right] \tag{34}
\]

\[
\leq \mathbb{E} \left[ (C_m - C_w)^\|\Phi_f^*\| \right] \tag{35}
\]

![KS test for Gamma distributions and real distributions of X ~ X(N_e, N, \Phi_f) via 10^5 Monte Carlo simulations, where \( \Phi_f \) is an arbitrary diagonal matrix as defined in Lemma 1.](image)
respectively. Note that $U_i$'s degrees of dissatisfaction of each sensor below a threshold, $\{\Phi_i^v\}$ can be expressed as follows:

$$R_i = \left[\tilde{C}_{m,i} - \mathbb{E}(C_{w,i}|\Phi_i^v)\right]$$

(36)

where

$$\mathbb{E}(C_{w,i}|\Phi_i^v) = \frac{1}{\ln(2)\Gamma(\mu)}G_{2,3}^{1,1}\left(\frac{\sigma_{e,i}^2}{vp\alpha_{e,i}^2}, 0, 0, \mu\right)$$

(37)

and $\mathbb{E}(C_{m,i}|\Phi_i^v) = \tilde{C}_{m,i}$. $\mu$ and $v$ are defined in (30) and (31), respectively. Note that $\tilde{C}_{m,i}$ is the ergodic channel capacity between $U_i$ and BS, which can be measured by the statistical method as the CSIs of $U_i$, i.e., $I_i$, $A_i$, and $h_i$ in each transmission burst can be obtained perfectly.

Proof: See Appendix B.

V. GAS-ORIENTED COMPUTATION OFFLOADING SCHEME

The proposed computation offloading scheme includes two processes, i.e., computational resource allocation and data uploading. The secrecy rates $R_i$ \ $\forall i$ in (4) are time variant as the coherence time is much shorter than the computation offloading duration. In this case, we use ergodic secrecy rate $\tilde{R}_i$ in Theorem 1 as the metric to allocate computational resource, i.e., use $\tilde{R}_i$ \ $\forall i$ to calculate $Q_{i,k}$ in (4) instead of $R_i$ \ $\forall i$. In the data uploading process, the CSIs of legitimate users are time variant and can be obtained via channel estimation in each transmission burst, so adaptive PLS coding should be designed to improve the efficiency of secure transmissions.

A. Computational Resource Allocation

1) Bipartite Graph Generation: In order to keep the degrees of dissatisfaction of each sensor below a threshold, i.e., constraint (7), $N_I$ sensors are sort as $\{U_1', U_2', \ldots, U_{N_I}'\}$ where their Gas obey $V_1' \geq V_2' \geq \cdots \geq V_{N_I}'$. Similarly, the MEC servers are sort as $\{M_1', M_2', \ldots, M_{N_I}'\}$ where their computational power obey $f_{1}/C_{1} \geq f_{2}/C_{2} \geq \cdots \geq f_{N_I}/C_{N_I}$. By dropping $N_K - N_I$ weaker MEC servers. We define $S = \{V_1', \ldots, V_{N_I}'\}$ and $M = \{M_1', \ldots, N_{N_I}'\}$ as the indices of the reordered sets of sensors and MEC servers, respectively.

Since secrecy rate is time varying during the computation offloading process, $Q_{i,k}$ defined in (4) cannot be measured. Instead of the secrecy rate, Theorem 1 provides an ergodic secrecy rate that is used to calculate $Q_{i,k}$ for each pair of a sensor and a MEC server as follows:

$$Q_{i,k} = \eta_k c_k D_{i,k} f_k^2 + \frac{D_{i,k} P_i}{\tilde{R}_i B}, i \in S, k \in M$$

(38)

and then a bipartite graph is generated to record all sensor-MEC combinations, as shown in Fig. 3. To meet constraint (7), the case of $r(V_i) - r(f_k/c_k) > \epsilon$ for any sensor-MEC pair should not be recorded in the bipartite graph. Hence, if $|i-k| > \epsilon$, $i \in S, k \in M, Q_{i,k}$ is set to be infinite.

2) Matching Process: We set an $N_I \times N_I$ matching matrix $W$, whose $(i,k)$th element represents an edge, e.g., $W_{i,k} = 1$ means that the vertexes $i$ and $k$ are linked. For another edge $W_{i,k}$, we have $i \neq j$ and $k \neq l$. In this case, we use the lower bound of real ergodic secrecy rate as the performance metric for the optimization process, as shown in (11).

$$P\left(\tilde{R}_i \leq \tilde{R}_i \text{ decoding correctly and } \Phi_i^v\right) = P\left(C_{w,i} \geq C_{m,i} - \tilde{R}_i\right)$$

(42)

where $C_{w,i}$ and $C_{m,i}$ are defined in (12) and (13) with $\Phi_i^v$, respectively. Since the instantaneous CSI of $I_i$, $A_i$, and $h_i$ are assumed to be estimated perfectly in each transmission burst, $U_i$ can adaptively adapt $C_{m,i}$ such that the codebook can be

![Fig. 3. Bipartite graph generation with the dissatisfaction degree constraint, where $\epsilon = 1$.](image)
decoded correctly. The remaining work is to find an optimal $\tilde{R}_i$ with the consideration of secrecy outage probability.

We use the effective secrecy rate as a secrecy metric to optimize $\tilde{R}_i$, as investigated in [3]. The effective secrecy rate is an average rate secretly received at BS over many transmission bursts with the predefined $\tilde{R}_i$, which is expressed as follows:

$$\tilde{R}_i = \left(1 - P(\tilde{R}_i)\right)\tilde{R}_i. \quad (43)$$

In order to maximize $\tilde{R}_i$ by adjusting $\tilde{R}_i$, the next work is to deduce the expression of effective secrecy rate, as shown in Theorem 2.

**Theorem 2 (Effective Secrecy Rate):** The expression of effective secrecy rate $\tilde{R}_i$ with the predefined $\tilde{R}_i$ can be expressed as follows:

$$\tilde{R}_i(\tilde{R}_i) = \left[1 - \frac{1}{\Gamma(N_e)}\Gamma(N_e, \frac{\phi}{1 + |\Phi_i^\dagger h_i|^2})\right]\tilde{R}_i. \quad (44)$$

where $\phi = \sigma^2_c/(2^{-\tilde{R}_i} - 1)/(\gamma^2_e, P_i)$.

**Proof:** As $P(\tilde{R}_i) = 1/\Gamma(N_e)\Gamma(N_e, \phi/(1 + |\Phi_i^\dagger h_i|^2))$ [41, eq. (23)], it is easy to get the expression of effective secrecy rate by substituting $P(\tilde{R}_i)$ into (43).

According to Theorem 2, the adaptive PLS coding scheme is designed to maximize the effective secrecy rate, i.e.,

$$\left(\tilde{R}_i^*\right) = \max_{\tilde{R}_i, P(\tilde{R}_i) \leq \varepsilon} \tilde{R}_i(\tilde{R}_i) \quad (45)$$

where the optimal PLS coding rate $\tilde{R}_i^*$ can be found via the golden-section search over the function $\tilde{R}_i(\tilde{R}_i)$ as the function $\tilde{R}_i(\tilde{R}_i)$ with respect to $\tilde{R}_i$ is unimodal, as examined in Fig. 4(b). Note that the secrecy outage probability $P(\tilde{R}_i)$ should be smaller than a threshold $\varepsilon$ to avoid a large secrecy outage probability in the transmission.

**C. Computational Complexity Analysis**

In the phase-shift matrix optimization process, the computational complexity to get $\Phi_i^\dagger \forall i$ is $N_f(4N^3 + 6N^2 + N)$. The computational resource allocation requires two bubble sort algorithms to get $\{U_1, U_2, \ldots, U_{N_t}\}$ and $\{M_1', M_2', \ldots, M_{N_B}'\}$, each of which needs $N_t^2$ iterations. The KM algorithm needs $N_t^3$ iterations to find the optimal matching $W^* \quad [39]$. The computational complexity of the PLS coding scheme based on the golden-section search algorithm is $O(\log(1/\varepsilon))$, where $\varepsilon$ is the accuracy requirement. In total, the computational complexity of the offloading process is $O[N_f(4N^3 + 6N^2 + N) + N_t^3 + 2N_t^2 + \log(1/\varepsilon)]$.

**VI. SIMULATIONS**

The global simulation parameters are described as follows. The path loss parameter $\alpha_i$ is calculated by $\alpha_i = \tau/(2\pi f_i d_i)$, where $\tau$ is the speed of light, $f_i$ is work spectrum that is set to 2.4 GHz, $d_i \forall i$ is the distance between the BS, and $U_i$ is set to be uniform distribution over [30 m, 50 m]. $f_k \forall k, D_i \forall i$, and $V_i \forall i$ are uniform distributions over [40 GHz, 80 GHz], [610 KB, 1.8 MB], and [1.5 $\times$ 10$^6$, 2 $\times$ 10$^6$]. The AWGN floor parameters $\sigma^2_c$ and $\sigma^2_e$ are $-53$ dBm [42]. The transmission power $P_i \forall i$ is 10 dBm. The computation energy efficiency coefficient $\eta_k \forall k$ is set to be $10^{-27}$. Note that all simulation results are average values from $10^5$ independent runs. The available radio resource block (RB) has 150-KHz bandwidth. If two RBs are used for transmission, we have $B = 300$ KHz.

Here, we examine Theorems 1 and 2 in Fig. 4(a) and (b), respectively. These figures show the good agreements between theoretical results (Theo) and Monte Carlo (MC) simulation results from $10^5$ independent runs. From Fig. 4(a), we can find that the ergodic secrecy rate increases with the increasing number of IRS elements. Also, a comparison is taken between IRS-assisted PLS and the secure beamforming scheme that uses $N_t = N + 1$ transmission antennas in the case of $N_B = 1$ and $N_e = 1$ [13]. From the comparison simulations, we find...
that the IRS-assisted PLS outperforms the secure beamforming scheme because the channel gain by IRS is larger than that of the beamforming scheme [37]. Fig. 4(a) also shows that the number of Eve’s antennas has a strong negative effect on the ergodic secrecy rate. Fig. 4(b) shows the effective secrecy rate in terms of PLS coding rate. It is obvious that there is one peak in each curve, which is consistent with the results shown in [3], meaning that the unimodal function-aimed search algorithms, such as golden-section search, can be used to find the optimal PLS coding rate to maximize the effective secrecy rate.

It is hard to obtain the global optimal result of \( \Phi_i \) in P2 via globally searching algorithms as \( \Phi_i \) is in an \( N \)-continuous dimensional space. Hence, the common way to solve P2 is based on SDR or manifold optimization, such as [32] and [33]. The comparison simulations between the SDR-based scheme and the manifold optimization scheme are shown in Fig. 5. From this figure, we can find the manifold optimization has better performance than that in the SDR-based scheme, because the rank one constraint results in that the SDR-based scheme just provides a near-optimal result for \( \Phi_i \) while the manifold optimization scheme can give the local optimal result [32].

The simulations of computational resource allocation are executed in terms of the number of RBs, the number of IRS elements \( N \), CNN per bit processing \( c_k \), and dissatisfaction degree \( \epsilon \). The comparison simulations have been done among four different allocation schemes, i.e., bidding, grouping, ECM, and the proposed scheme, which are described detailedly as follows.

1) **Bidding Scheme:** It is a traditional strategy that the sensor who pays the highest Gas will get the best computational resource among the MEC servers. This scheme requires two bubble sort algorithms and needs \( O[N_t(4N^4 + 6N^3 + N) + 2N_t^2 + \ln(1/\epsilon)] \) computational complexity.

2) **Group Scheme:** It is proposed in [31] where sensors are divided into different groups based on paid Gas, and the group with higher Gas is prioritized with better computational resources. The computational complexity of this scheme is \( O[N_t(4N^4 + 6N^3 + N) + 2N_t^2 + N_G(N_m + 1)^4 + \ln(1/\epsilon)] \), where \( N_G \) and \( N_m \) are the numbers of groups and sensors in each group, respectively.

3) **ECM Scheme:** In the ECM scheme, the bipartite graph is generated to record the \( Q_{i,k} \) for all sensor-MEC combinations, then, the KM algorithm is used to find the optimal matching to minimize the energy consumption, as used in [8]–[10]. The computational complexity of this scheme is \( O[N_t(4N^4 + 6N^3 + N) + 2N_t^2 + N_t^4 + \log(1/\epsilon)] \).

4) **Proposed Scheme:** This article uses joint phase-shift matrix optimization and Gas-oriented computation offloading to reduce the energy consumption while keeping the dissatisfaction degrees of sensors are below a threshold. The computational complexity is similar to the ECM scheme, as discussed in Section V-D.

From Fig. 6, we can check the number of RBs on the sum energy consumption of 40 sensors, where \( c = 8 \), \( N = 40 \), and \( c = 10 \). It is demonstrated that the sum energy consumption decreases with the increasing number of RBs, because data transmission time is reduced with more bandwidth. Then, we find the increasing trend becomes slow when the bandwidth is enough as the global performance should consider other factors, such as computing power. Also, the proposed scheme outperforms the bidding and group schemes, but has more energy consumption than the ECM scheme.

It is demonstrated that the increasing number of IRS elements will reduce the sum energy consumption, as shown in Fig. 7. However, we find the decline is slow because the computation offloading system follows the “cask principle,” meaning that unilaterally improving the capability of communication modules cannot give a large gain for the performance of computation offloading systems. Fig. 8 shows the sum energy consumption in terms of CCN per bit processing. There exists a rough linear rise for sum energy consumption with the CCN per bit processing. As the CCN per bit processing is the main metric for the algorithm efficiency of CPU chips,
it implies that the best to reduce energy consumption is the improvement of the efficiency of CPU designs.

**Fig. 9** is used to check the dissatisfaction degree of sensors on the sum energy consumption. It is obvious that if the sensor has the highest dissatisfaction, the proposed scheme has the same performance as that in the ECM scheme, meaning the optimization objective of the proposed scheme is to minimize the sum energy consumption and the satisfaction of sensors is no longer considered. Also, if \( \epsilon = 0 \), the sensor gets the computational resource whose ranking should be equivalent to the ranking of the paid Gas among all sensors. Hence, the performance of \( \epsilon = 0 \) is the same as the bidding scheme. Dissatisfaction degrees are not considered in bidding, grouping, and ECM schemes, so the simulation lines keep constant in this figure.

**VII. Conclusion**

In this article, we establish a blockchain-empowered computation offloading system in an IoT network, where the data is protected by IRS-assisted PLS methods. Especially, we deduce the expression of the ergodic secrecy rate of the scheme for the computational resource allocation. Also, we design a Gas-oriented ECM algorithm where higher Gas providers are prioritized with better computational resources, while improving the effective secrecy rate in the data uploading process. Simulations show that the proposed scheme reduces the energy consumption while guaranteeing that the node paying higher Gas has more opportunities to get a stronger computational resource. In future works, we will integrate the promising space-division multiple-access technologies into this system to further reduce the latency of the computation offloading process.

**APPENDIX A**

**Proof of Lemma 1**

**Lemma 2** (Proved in [43]): For \( n \) independent Gaussian random variables \( X_i \sim \mathcal{N}(\mu_i, \sigma^2) \), \( Y = \sum_{i=1}^{n} X_i^2/\sigma^2 \) obeys the noncentral chi-square distribution with degree of freedom \( n \), and we have the mean and variance of \( Y \) as follows:

\[
\mathbb{E}(Y) = n + \lambda,
\quad \mathbb{V}(Y) = 2n + 4\lambda
\]

(47)

where \( \lambda = \sum_{i=1}^{n} \mu_i^2/\sigma^2 \).

First, we calculate the mean of \( X \sim X(N_e, N, \Phi) \) as follows:

\[
\mathbb{E}(x) = \mathbb{E}((|g|^2) + \mathbb{E}((|Z\Phi h|^2).
\]

(48)

It is obvious that \( \mathbb{E}(|g|^2) = N_e \).

Since \( \Phi \) is independent with \( g, Z, \) and \( h \), we introduce two auxiliary random variables \( u_1 = \Phi h \) and \( u_2 = Z\Phi h/|\Phi h| \) with \( u_1 \sim \mathcal{CN}_{N_e,1}(0, I_N) \) and \( u_2 \sim \mathcal{CN}_{N_e,1}(0, I_{N_e}) \), respectively. It is obvious that \( u_1 \) and \( u_2 \) are independent, thus, \( \mathbb{E}(|Z\Phi h|^2) \) can be transformed to

\[
\mathbb{E}(|Z\Phi h|^2) = \mathbb{E}((|Z\Phi h|/|\Phi h|)^2) \mathbb{E}(|\Phi h|^2) \overset{d}{=} N_eN
\]

(49)

where \( \overset{d}{=} \) is achieved because \( \mathbb{E}((|Z\Phi h|/|\Phi h|)^2) = N_e \) and \( \mathbb{E}(|\Phi h|^2) = N_e \). Hence, the mean of \( x \) can be rewritten as
\[
\int_0^\infty z^{\theta-1} (\gamma + \nu)^{-\eta} G_{pq}^m (\mu z^{-1} b_1, \ldots, b_q) \, dz = \frac{\nu^{\theta-\eta}}{\Gamma(\eta)} G_{pq+1, q+1}^{m+1, n+1} \left( \mu \frac{1 - \rho, a_1, \ldots, a_p}{\eta - \rho, b_1, \ldots, b_q} \right)
\] (46)

follows:
\[
E(x) = N_e(1 + N).
\] (50)

Then, we will deduce the variance of \(x\), i.e., \(\mathcal{V}(x)\), which is given as follows:
\[
\mathcal{V}(x) = E\left( |x|^2 \right) - |E(x)|^2
\] (51)

where \(E(|x|^2)\) can be expressed as follows:
\[
E\left( |x|^2 \right) = E\left( |g + Z \Phi h|^2 \right) = E\left( |g|^2 \right) + E\left( |Z \Phi h|^2 \right) + E\left( |h^H \Phi^H Z^H g + g^H Z \Phi h|^2 \right)
\]

\[
\mathcal{V}(x) = N_e^2 + N_e.
\] (53)

According to the property of noncentral chi-square distribution defined in Lemma 4, we have
\[
E\left( |g|^2 \right) = \mathcal{V}(g) = |E(g)|^2 = N_e^2 + N_e.
\] (53)

Also, we can get \(E(|g|^2 | Z \Phi h|^2) = N_e^2 N, E(|h^H \Phi^H Z^H g|^2) = E(|g^H Z \Phi h|^2) = N_e N,\) and
\[
E\left( |Z \Phi h|^4 \right) = \mathcal{V}(Z \Phi h) = \left( N_e + N_e^2 \right) (1 + N) N.
\] (54)

Then, according to (51), we have the variance of \(x\) as follows:
\[
\mathcal{V}(x) = N_e (1 + N)^2 + \left( N_e + N_e^2 \right) N.
\] (55)

According to the definition of the Gamma distribution [38], the shape and scale of the Gamma distribution can be expressed as follows:
\[
\mu = \frac{E(x)}{\mathcal{V}(x)}, \quad \nu = \frac{\mathcal{V}(x)}{E(x)}
\] (56)

Then, we get the PDF and CDF of \(X \sim X(N_e, N, \Phi)\) as defined in (28) and (29).

The proof is completed.

**APPENDIX B**

**PROOF OF THEOREM 2**

Here, we deduce the ergodic capacity between \(U_i\) and Eve \(E(C_{w,i} | \Phi^*_i)\) as follows. Since \(\Phi^*_i\) is related with CSIs of legitimate users and is independent with \(g, Z_i, h_i\), \(E(C_{w,i} | \Phi^*_i)\) can be expressed as follows:
\[
E(C_{w,i} | \Phi^*_i) = \int_0^\infty \log_2 \left( 1 + \frac{\sigma^2_e, i P_i}{\sigma^2_e, i} x \right) f_X(x) \, dx
\]

\[
= \frac{1}{\ln(2)} \int_0^\infty \ln \left( 1 + \frac{\sigma^2_e, i P_i}{\sigma^2_e, i} x \right) f_X(x) \, dx
\]

\[
= \frac{1}{\ln(2)} \int_0^\infty \frac{1}{1 + z} \left[ 1 - F_X \left( \frac{\sigma^2_e, i z}{P_i \sigma^2_e, i} \right) \right] \, dz
\]

\[
= \frac{1}{\ln(2)} \Gamma(\mu) \int_0^\infty \frac{\Gamma(\mu, \nu) \sigma^2_e, i}{\sigma^2_e, i} \, dx
\] (57)

where \(\sigma^2_e, i\) is due to a change of variable \(z = [(\sigma^2_e, i P_i)/(\sigma^2_e, i)] x\). \(F_X(\cdot)\) is CDF of the variable \(X \sim X(N_e, N, \Phi)\), as defined in (29). \(\mu\) and \(v\) are defined in (30) and (31). According to [44, eq. (5)], we have
\[
\Gamma(a, x) = G_{1, 2}^0 \left( x \frac{1}{0, a} \right).
\] (58)

Jointly considering (46), shown at the top of the page [45, 7.812 EI II 418(4)], we get
\[
\int_0^\infty \frac{\Gamma(\mu, \nu \sigma^2_e, i)}{\sigma^2_e, i} \, dx = \int_0^\infty (1 + z)^{-1} G_{1, 2}^0 \left( \frac{\sigma^2_e, i z}{v P_i \sigma^2_e, i} \right) \, dz
\]

\[
= G_{2, 3}^3 \left( \frac{\sigma^2_e, i}{v P_i \sigma^2_e, i} \right) \left[ 0, 1, \mu \right].
\] (59)

Substituting (59) into (57), we get the expression of the ergodic capacity of between \(U_i\) and BS as follows:
\[
E(C_{w,i} | \Phi^*_i) = \frac{1}{\ln(2) \Gamma(\mu)} G_{2, 3}^3 \left( \frac{\sigma^2_e, i}{v P_i \sigma^2_e, i} \right) \left[ 0, 1, \mu \right].
\] (60)

Substituting (60) into (36), we obtain the expression of the ergodic secrecy rate.

The proof is completed.

**REFERENCES**

[1] J. Gozalvez, “New 3GPP standard for IoT [mobile radio],” IEEE Veh. Technol. Mag., vol. 11, no. 1, pp. 14–20, Mar. 2016.

[2] Y. Wang et al., “SPDS: A secure and auditable private data sharing scheme for smart grid based on blockchain,” IEEE Trans. Ind. Informat., vol. 17, no. 11, pp. 7686–7699, Nov. 2021.

[3] Y. Liu et al., “Physical layer security assisted computation offloading in intelligently connected vehicle networks,” IEEE Trans. Wireless Commun., vol. 20, no. 6, pp. 3555–3570, Jun. 2021.

[4] Z. Su, Y. Wang, Q. Xu, and N. Zhang, “LVBS: Lightweight vehicular blockchain for secure data sharing in disaster rescue,” IEEE Trans. Dependable Secure Comput., vol. 19, no. 1, pp. 19–32, Jan./Feb. 2022.

[5] Y. Wu, H.-N. Dai, and H. Wang, “Convergence of blockchain and edge computing for secure and scalable IoT critical infrastructures in industry 4.0,” IEEE Internet Things J., vol. 8, no. 4, pp. 2300–2317, Feb. 2021.

[6] L. Xie, H. T. Luan, Z. Su, Q. Xu, and N. Chen, “A game theoretical approach for secure crowdsourcing-based indoor navigation system with reputation mechanism,” IEEE Internet Things J., early access, Sep. 13, 2021, doi: 10.1109/JIOT.2021.3111999.

[7] M. Dai, Z. Su, Q. Xu, Y. Wang, and N. Lu, “A trust-driven contract incentive scheme for mobile crowd-sensing networks,” IEEE Trans. Veh. Technol., vol. 71, no. 2, pp. 1794–1806, Feb. 2022.
Zhou Su (Senior Member, IEEE) has published technical papers, including top journals and top conferences, such as IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE/ACM TRANSACTIONS ON NETWORKING, and INFOCOM. His research interests include multimedia communication, wireless communication, and network traffic.

Dr. Su received the Best Paper Award of International Conference IEEE ICC2020, IEEE BigdataSE2019, and IEEE CyberSciTech 2017. He is an Associate Editor of IEEE INTERNET OF THINGS JOURNAL, IEEE OPEN JOURNAL OF COMPUTER SOCIETY, and IET Communications.

Yuntao Wang (Student Member, IEEE) is currently pursuing the Ph.D. degree with the School of Cyber Science and Engineering, Xi’an Jiaotong University, Xi’an, China.

His research interests include security and privacy protection in general wireless networks, vehicular networks, and UAV networks.