# BECS: A Privacy-Preserving Computing Resource Sharing Mechanism for 6G Computing Power Network

Kun Yan , *Member, IEEE*, Wenping Ma , *Member, IEEE*, Shaohui Sun , *Member, IEEE*

***Abstract*—The 6G Computing Power Network (CPN) is envisioned to orchestrate vast, distributed computing resources for future intelligent applications. However, achieving efficient, trusted, and privacy-preserving computing resource sharing in this decentralized environment poses significant challenges. To address these issues, this article proposes a blockchain and evolutionary algorithm-based computing resource sharing (BECS) mechanism. BECS is designed to dynamically and adaptively balance task offloading among computing resources within the 6G CPN, thereby enhancing resource utilization. We model computing resource sharing as a multi-objective optimization problem, aiming to improve resource utilization while addressing other trade-offs. To tackle this NP-hard problem, we devise a kernel-distance-based dominance relation and incorporate it into the Non-dominated Sorting Genetic Algorithm III (NSGA-III), thereby significantly enhancing population diversity. In addition, we propose a pseudonym scheme based on zero-knowledge proofs to protect user privacy during computing resource sharing. Finally, security analysis and simulation results demonstrate that BECS can effectively leverage all computing resources in the 6G CPN, thereby significantly improving resource utilization while preserving user privacy.**

***Index Terms*—6G computing power network, computing resource sharing, multi-objective evolutionary optimization, blockchain, pseudonym scheme.**

## I. Introduction

In the upcoming 6G era, it is anticipated that everything will be intelligently connected, enabling a wide range of data-intensive applications. By deeply integrating communication networks with various vertical industries, 6G will enable unprecedented applications such as Holographic Integrated Sensing and Communication (HISC), Artificial General Intelligence (AGI), and Digital Twins (DT) [1]. This indicates that AI will be one of the most crucial technologies for constructing a comprehensively intelligent 6G networks, enabling network services to evolve dynamically and autonomously in response to demand. The coordination of end-edge-cloud computing devices to form a Computing Power Network (CPN) is expected to become a leading paradigm for supporting ubiquitous intelligent services in 6G networks [2]. In essence, the CPN envisions intelligent orchestration, where computing tasks are dynamically offloaded to the most suitable execution environment based on real-time service demands. This fosters a highly dynamic and hybrid computing environment that promotes the sharing of computing resources across 6G networks. However, realizing this vision in the 6G CPN presents fundamental challenges. In particular, it is necessary to design a resource sharing mechanism that can simultaneously ensure trustworthiness and preserve user privacy, while navigating the complex performance trade-offs inherent in such a massive and decentralized system.

While Multi-access Edge Computing (MEC) in 5G networks paved the way for offloading computing tasks to the network edge and unleashing potential for the Internet of Things (IoT) [3], its limitations become evident when faced with more complex demands in the future. MEC is typically characterized by localized deployments, limited resources, and siloed management, all of which hinder the integration and orchestration of ubiquitously distributed, heterogeneous computing resources across the network. In the 6G era, networks are expected to scale up to support trillions of connections. Meanwhile, intelligent and diversified applications will not only require computing resources that far exceed previous levels but will also urgently demand a ubiquitous computing paradigm. This paradigm must enable real-time coordination of network-wide computing resources and dynamic on-demand orchestration. Crucially, it involves navigating complex trade-offs among multiple, often conflicting, performance objectives such as minimizing latency, reducing energy consumption, and maximizing resource utilization across seamless cross-domain allocations [4]. As a result, 5G MEC is inadequate for meeting future computing demands in terms of both scale and operational paradigms. Therefore, 6G networks require the CPN to deeply integrate and intelligently orchestrate dispersed, heterogeneous, and geographically distributed computing resources across the entire network. Such integration promotes full resource utilization and enables efficient, large-scale computing resource sharing [5].

Driven by the unprecedented features and services of future 6G networks, communication networks are increasingly exploring distributed or multi-center management models [6]. Furthermore, the 6G CPN aims to comprehensively mobilize vast heterogeneous devices, edge nodes, and cloud computing resources across organizational boundaries. This cross-domain and heterogeneous environment, which lacks a central authority of trust, poses significant challenges for the access,

Manuscript received April 22, 2025. This work was supported in part by the Key Industry Innovation Chain Project of Shaanxi Provincial Science and the Technology Department, China under Grant 2022ZDLGY03-08. *(Corresponding author: Wenping Ma.)*

Kun Yan and Wenping Ma are with the School of Telecommunications Engineering, Xidian University, Xi'an 710071, China (e-mail: kyan@stu.xidian.edu.cn; wp_ma@mail.xidian.edu.cn).

Shaohui Sun is with the State Key Laboratory of Wireless Mobile Communication, Datang Mobile Communications Equipment Co., Ltd, Beijing 100083, China (e-mail: sunshaohui@catt.cn).

scheduling, trading, and management of computing resources [7]. Traditional centralized architectures not only struggle with inefficiency and single points of failure, but they also lack the transparency and immutability needed to build trust among diverse stakeholders. Therefore, it is necessary to establish a novel distributed management mechanism that enables autonomous, reliable, and efficient collaboration among devices. Such a mechanism must provide verifiable, tamper-proof records for resource trading and enforce operational rules transparently, without relying on a central intermediary. Blockchain, with its inherent decentralization and its ability to enhance trust and transparency among multiple stakeholders, emerges as a key enabling technology for future 6G network management [8]. It offers secure, transparent, and traceable solutions for computing resource sharing, trusted trading, access control, and other critical functionalities within the CPN.

Furthermore, the ubiquitous sharing and collaboration of computing resources exacerbate the risk of user privacy leakage. The 6G CPN involves a wide variety of computing devices. In particular, when user devices participate in computing resource sharing, not only their identity but also their behavioral patterns and interaction records can be collected and linked, allowing attackers to infer sensitive information [9]. These potential threats create a significant barrier to participation. To achieve efficient computing resource sharing, it is essential to provide a mechanism that allows users to prove their legitimacy and conduct transactions without revealing their persistent identities, thus ensuring both accountability and anonymity.

To address these intertwined challenges of efficiency, trust, and privacy, this article introduces a novel mechanism for 6G CPN, named BECS, which utilizes **B**lockchain and **E**volutionary algorithms for efficient **C**omputing **S**haring. It is designed to create a robust and efficient ecosystem where heterogeneous computing resources can be shared on a large scale. By effectively tackling the critical issues of resource allocation efficiency, decentralized trust, and user privacy preservation, BECS aims to unlock the full potential of distributed computing in 6G networks. Performance evaluations confirm that the proposed BECS mechanism improves resource utilization by up to 54%. Specifically, the main contributions of this article are summarized as follows:

- We propose a dynamic and efficient blockchain-based mechanism for computing resource sharing, aimed at ensuring secure allocation and trading of resources between any devices in 6G CPN, thus enhancing utilization.
- We formulate computing allocation as a multi-objective optimization problem (MOOP) with six objectives, employing an evolutionary algorithm to balance the interplay among these objectives, thus achieving optimal allocation schemes for 6G CPN.
- We propose a novel evolutionary algorithm, NSGA-III-KDR, which improves the dominance relation of Non-dominated Sorting Genetic Algorithm III (NSGA-III) by using the kernel distance to enhance diversity in addressing the computing allocation MOOP.
- We design a novel pseudonym scheme based on the Schnorr protocol, which protects user privacy during computing resource sharing in 6G CPN.

The remainder of this article is organized as follows. Section II reviews related work. Section III introduces the system model and formulates the computing resource sharing problem. Section IV describes NSGA-III-KDR and its application in solving the computing allocation MOOP. Section V presents the proposed pseudonym scheme and computing trading. Section VI analyzes the security and computational complexity of the proposed scheme. Section VII presents simulation results. Section VIII concludes the article.

## II. Related Works

In this section, we first introduce the paradigm of the 6G CPN and contrast it with 5G MEC. Then, we discuss existing works on computing allocation and sharing. Finally, we elaborate blockchain for decentralized management and privacy preservation.

### A. The Paradigm of 6G Computing Power Network

The trajectory from 5G to 6G signals a profound transformation, moving beyond the traditional pursuit of enhanced communication metrics. Driven by the rapid advancement of AI, it represents a fundamental paradigm shift toward the deep integration of computing and networking [4]. Although 5G MEC optimizes computing services by pushing computing power to the network edge, it is typically an isolated and localized computing architecture managed by a single operator within a limited area [10]. Consequently, despite MEC's computing resources being proximity to the user, they are unable to achieve efficient orchestration and collaboration with other computing resources on a network-wide scale.

6G CPN will deeply integrate all computing devices within the network to form a distributed and intelligent computing environment [11]. By seamlessly abstracting, virtualizing, and integrating decentralized computing resources managed by different operators and originating from diverse device types, 6G CPN aims to unify the heterogeneous resource landscape. It constructs a unified computing architecture that spans from user devices to multi-domain edge nodes and cloud centers. [12]. This transition from isolated resources to a unified architecture also assigns different identity roles to computing devices within the network. In 5G MEC, user devices are typically resource requesters, while edge servers act as resource providers. In contrast, 6G CPN breaks this limitation, every computing device can function both as a requester and a provider, thereby enabling full utilization of all computing resources in the network [13].

### B. Computing Allocation and Sharing in CPN/End-Edge-Cloud

The dynamic allocation and efficient utilization of computing resources are central to the concept of CPN. Addressing the inherent optimization challenges in managing these aspects has been a key research focus. Lu *et al.* [2] utilized deep reinforcement learning to find the optimal task transfer and

TABLE I
DIFFERENCES BETWEEN BECS AND OTHER MAIN RELATED WORKS

| Ref. | Core Scenario | Optimization Method | MOOP | Cloud-Edge-End Collaboration | Blockchain Features | Advanced Privacy Scheme | 6G Relevance |
|---|---|---|---|---|---|---|---|
| [2] | Energy-efficient task transfer in Wireless CPN | Multi-Agent Deep Reinforcement Learning | × | × | × | × | ✓ |
| [5] | Efficient task offloading in Edge CPN | Two-Stage Evolutionary Search | × | × | × | × | ✓ |
| [15] | Computation offloading in Industrial IoT | NSGA-III | ✓ | ✓ | × | × | × |
| [16] | Dependent task offloading in MEC | MOEA/D | ✓ | ✓ | × | × | × |
| [18] | Secure computation offloading in IoT | Deep Reinforcement Learning | × | ✓ | ✓ | × | × |
| [19] | Cooperative task offloading in MEC | Multi-Agent DRL, Game Theory | × | × | ✓ | × | × |
| [20] | Secure computation offloading in cyber-physical systems | Deep Reinforcement Learning | × | ✓ | ✓ | × | × |
| [21] | Secure task offloading in MEC | Distributed Deep Q-Learning | × | ✓ | ✓ | × | × |
| **BECS** | Privacy-preserving computing resource sharing in 6G CPN | NSGA-III-KDR | ✓ | ✓ | ✓ | ✓ | ✓ |

computing allocation strategies in wireless CPN. Chen *et al.* [5] proposed an on-demand two-stage computing resource scheduling model to achieve efficient task offloading in edge CPN. However, as 6G services become increasingly diverse and demanding, optimizing for single or dual objectives often proves insufficient to capture the inherent trade-offs between factors like latency, energy consumption, cost, and resource utilization [14].

Therefore, formulating computing allocation as a MOOP has become a more effective approach for handling these conflicting objectives. Meanwhile, evolutionary algorithms are widely adopted due to their ability to identify a set of non-dominated solutions representing different trade-offs. Peng *et al.* [15] formulated complex task offloading in the IIoT as a four-objective MOOP and developed a method to dynamically allocate computing resources based on the NSGA-III. Gong *et al.* [16] employed the multi-objective evolutionary algorithm based on decomposition (MOEA/D) to optimize a three-objective edge task offloading problem, aiming to minimize delay and maximize rewards. While these approaches demonstrate the applicability of MOEAs, standard algorithms like NSGA-III and MOEA/D can face challenges when dealing with many-objective optimization, often struggling to maintain sufficient population diversity alongside convergence pressure [17]. This limitation can hinder the exploration of the full solution space, particularly for complex 6G CPN scenarios involving numerous performance dimensions and heterogeneous resources.

### *C. Blockchain for Decentralized Management and Privacy Preservation*

Building on the need for decentralized trust and management in 6G networks, blockchain technology has been actively explored as a key enabler [22]. Xie *et al.* [23] exploited the immutability of blockchain to propose a resource trading mechanism based on sharding and directed acyclic graphs for large-scale 6G networks, thereby enhancing resource utilization efficiency. Nguyen *et al.* [18] proposed a blockchain-based mobile edge-cloud computation offloading scheme, leveraging the distributed characteristics of blockchain to to provide secure and trusted computing services. Wang *et al.* [24] proposed a provable secure blockchain-based federated learning framework for wireless CPN, aimed at accelerating the convergence of federated learning and enhancing the efficiency of wireless CPN. These works highlight blockchain's potential to automate, secure, and streamline interactions like resource discovery, access control, scheduling coordination, and payment settlement via smart contracts [25], thereby fostering a reliable environment for large-scale computing resource sharing.

6G networks will integrate AI to fully merge the physical and digital worlds, necessitating enhanced security and privacy [26]. Nguyen *et al.* [19] utilized blockchain to provide adequate security for task offloading in mobile edge computing. Wang *et al.* [20] proposed a blockchain-enabled cyber-physical system integrating cloud and edge computing to achieve secure computing offloading. Samy *et al.* [21] introduced a blockchain-based framework for task offloading, ensuring security, integrity, and privacy in mobile edge computing. Although leveraging the characteristics of blockchain can provide preliminary security and privacy protection for edge and cloud computing allocation and trading, device-to-device computing resource sharing in 6G networks will require more comprehensive solutions to ensure the security and privacy between devices [27].

An overview of related works is given in Table I. Distinct from the aforementioned works, BECS is designed as a holistic mechanism for the 6G CPN. It uniquely combines a blockchain architecture, advanced multi-objective optimization

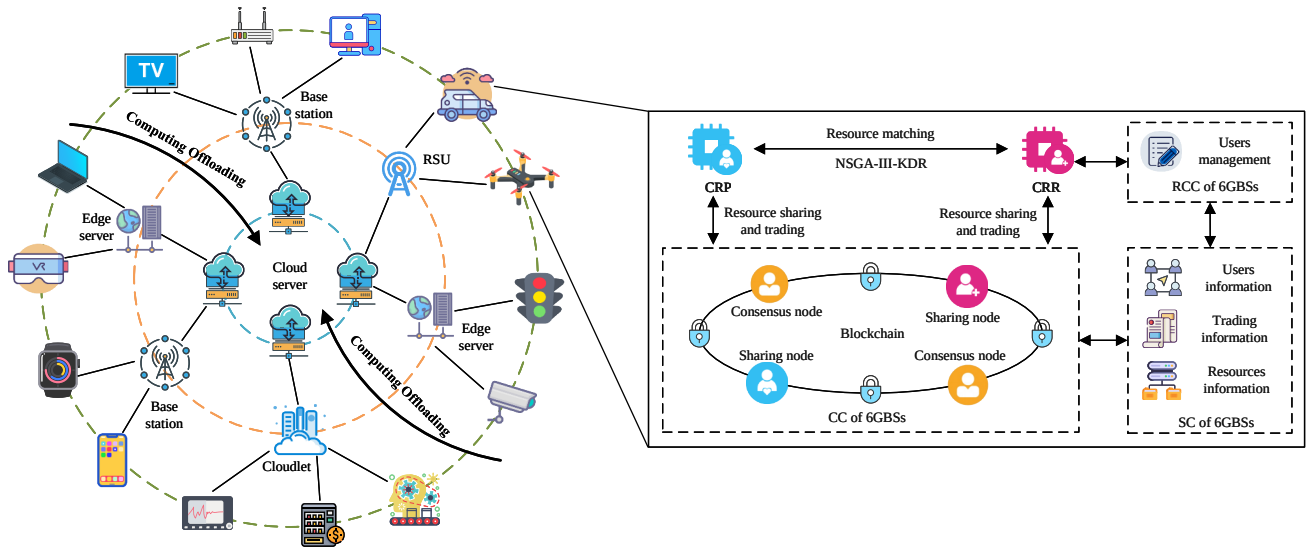


Fig. 1. Overview of the BECS architecture in 6G CPN.

techniques, and cryptographic privacy protection to tackle the intertwined challenges of efficiency and security in computing resource sharing of 6G CPN.

## III. System Model and Problem Formulation

In this section, we introduce the computing resource sharing system model considered by BECS, along with other models used in constructing the MOOP, including the communication, computing, and service models.

### A. System Model

To meet the demands of future intelligent applications, BECS introduces a generalized architecture, as illustrated in Fig. 1, designed to support diverse forms of computing resource sharing in 6G CPN. In this architecture, all computing devices, denoted as $\mathbb{D} = \{D_1, \ldots, D_d, \ldots, D_D\}$, are categorized into three layers: the user computing layer, edge computing layer, and cloud computing layer. This classification fundamentally redefines their interplay. Moreover, BECS overcomes the rigid and siloed structure of traditional 5G MEC. Instead of treating computing resources as isolated within discrete physical layers, BECS abstracts and virtualizes all participating devices across the network into an integrated, unified, and orchestrated resource pool. Consequently, the layer classification reflects the functional capabilities of devices rather than enforcing rigid operational boundaries. The resources in $\mathbb{D}$ are inherently heterogeneous, encompassing devices from different administrative domains, multiple operators, and diverse ownerships, thereby forming a cohesive, network-wide computing continuum that can be dynamically orchestrated.

The proposed BECS is primarily designed for a metropolitan area network scenario. In this context, numerous computing devices are geographically distributed within a city and are interconnected through a high-bandwidth, low-latency optical fiber backbone. This dense and high-speed infrastructure represents a key characteristic of future 6G-enabled smart city deployments [28]. Therefore, BECS focuses on resource sharing within this metropolitan scope, where the primary communication bottlenecks and latency variations originate from wireless access links and task processing, rather than the stable, high-speed wired backbone. Specifically, the set of user computing devices, denoted as $\mathbb{U} = \{U_1, \ldots, U_u, \ldots, U_U\}$, includes smartphones, computers, wearable devices, IoT devices, vehicles, and other devices that directly interact with users. These devices typically possess limited computing power. The set of edge computing devices, denoted as $\mathbb{E} = \{E_1, \ldots, E_e, \ldots, E_E\}$, includes edge servers, roadside units, cloudlets, and other devices capable of providing time-sensitive computing services to users. The set of cloud computing devices, denoted as $\mathbb{C} = \{C_1, \ldots, C_c, \ldots, C_C\}$, consists of remote computing centers capable of providing large-scale computing power. The evolutionary algorithm matches computing resource requesters with providers, facilitating a multi-dimensional measurement of the deep reuse of computing resources. Additionally, with the support of permissioned blockchain, BECS enables dynamic management and trading of fine-grained computing resources. The proposed architecture consists of three main components:

1) Computing Resource Providers (CRPs): As large language models become increasingly widespread, devices with abundant computing power will increasingly provide computing support to devices with limited resources. In BECS, all devices within 6G CPN with free computing resources can serve as CRPs.

2) Computing Resource Requesters (CRRs): In general, when a device lacks sufficient computing capability to handle a task's demands, it needs to request additional resources. In BECS, any device can be a CRR, provided that the requested resources exceed its own computing capacity.

3) 6G Base Stations (6GBSs): As critical components in BECS, 6GBSs provide reliable communication services to devices and serve as blockchain maintenance nodes responsible for transaction bookkeeping and block packaging. Each 6GBS

consists of three components:

- Registration and Certification Component (RCC): The RCC is responsible for managing the identities of devices and issuing and verifying certificates.
- Computation Component (CC): The CC is responsible for maintaining the blockchain.
- Storage Component (SC): The SC is responsible for storing data from devices and the blockchain.

In contrast to the static client-server architecture of 5G MEC, CRP and CRR in BECS do not represent fixed device roles but are instead transient functional states of computing devices, determined by their real-time computational surplus or deficit. This fluidity and flexibility are essential to unlocking the potential of computing resource sharing in 6G CPN. Meanwhile, with the support of blockchain and evolutionary algorithms, computing devices can autonomously match and schedule supply and demand, thereby meeting the ubiquitous and heterogeneous computing demands anticipated in future networks.

This study primarily focuses on efficient computing resource sharing. Accordingly, the computing device $D_d$ is denoted as $D_d = \{\varsigma_d, \psi_d\}$, where $\varsigma_d$ is usually measured by device's CPU [29]. The inherent heterogeneity of the 6G CPN is captured by the diverse characteristics of these three layers, where devices possess varying computing capacities, service prices, and are subject to different network conditions. Meanwhile, $\mathbb{T} = \{T_1, \ldots, T_t, \ldots, T_T\}$ denotes the set of all computing tasks. The tuple $T_t = \{\varphi_t, \xi_t, \tau_t, d_t\}$ describes the computing task $T_t$. $\varphi_t$ and $\xi_t$ can be obtained using methods described in [30], such as graph analysis. Notations that will be used are presented in Table II.

TABLE II
SUMMARY OF NOTATIONS

| Notation | Definition |
|---|---|
| $\mathbb{D}$ | the set of all computing devices |
| $\mathbb{U}$ | the set of user computing devices |
| $\mathbb{E}$ | the set of edge computing devices |
| $\mathbb{C}$ | the set of cloud computing devices |
| $\mathbb{T}$ | the set of all computing tasks |
| $\varsigma_d$ | the computing capacity (in CPU cycles/s) of $D_d$ |
| $\psi_d$ | the service price of $D_d$ per unit of time |
| $\varphi_t$ | the data size of $T_t$ |
| $\xi_t$ | the required computing amount (in CPU cycles) of $T_t$ |
| $\xi_d$ | the average computing amount (in CPU cycles) of tasks assigned to $D_d$ |
| $\tau_t$ | the maximum time consumption allowed of $T_t$ |
| $d_t \in \{u_t, e_t, c_t\}$ | the final execution device for $T_t$ |
| $p_u$, $p_n$ | the transmission power of $U_u$ and $U_n$ |
| $\sigma_0^2$ | the noise power |
| $B$ | the bandwidth of each subchannel |
| $N_d = \{U, E, C\}$ | the number of $D_d$ in each layer |
| $\rho_d$ | the computing resources occupancy of $D_d$ in each layer |
| $\alpha_d$ | the tasks processed number per unit of $D_d$'s computing capacity |
| $p_{d^* \to d,t}$ | the probability of offloading $T_t$ from $D_{d^*}$ to $D_d$, and satisfies $\sum_{d \in \{u,e,c\}} p_{d^* \to d,t} = 1$ |
| $\kappa_d$ | the effective capacitance coefficient of $D_d$ |
| $P_{ide}$ | the minimum value of the objective function |

### B. Communication Model

In BECS, a Non-Orthogonal Multiple Access (NOMA) based communication model is considered for the 6G uplink, which allows multiple user devices to transmit data to the 6GBS over the same frequency band, thereby significantly enhancing spectral efficiency [31]. Successful decoding in NOMA hinges on the Successive Interference Cancellation (SIC) technique at the receiver [32]. This requires the signals to be decoded sequentially, which is predicated on a known channel gain order.

Without loss of generality, the users are indexed such that their channel gains are in descending order, i.e., $g_1 > \cdots > g_u > \cdots > g_U$. Consequently, when decoding the signal for user $U_u$, signals from users with stronger channel conditions $(U_1, \ldots, U_{u-1})$ have already been cancelled. The signals from users with weaker channel conditions $(U_{u+1}, \ldots, U_U)$ constitute the co-channel interference. The resulting data transmission rate for $U_u$ is therefore calculated via Shannon's theorem as follows:

$$R_u = B \log_2 \left( 1 + \frac{p_u g_u}{\sigma_0^2 + \sum_{n=u+1}^{U} p_n g_n} \right). \tag{1}$$

If $U_u$ offloads $T_t$ to $D_d$, the data transmission latency can be expressed as follows:

$$L_{u,t}^{tra} = \frac{\varphi_t}{R_u}. \tag{2}$$

Therefore, the communication energy consumption of $U_u$ can be calculated as follows:

$$E_{u,t}^{tra} = p_u L_{u,t}^{tra}. \tag{3}$$

### C. Computing Model

In the considered three layers computing resource sharing structure, the computing tasks of a CRR can be executed in any device of CRP. However, computing devices vary in terms of task transfer and processing capabilities. This intense competition for computing resources at each layer, inevitable in 6G PCN with numerous CRRs and CRPs, necessitates balancing the task load across the system's layers. Generally, with limited computing resources at each layer, tasks may have to wait for an available processor. Therefore, the M/M/c model [33], which describes task offloading as a Poisson process with an average arrival rate of $\lambda_t$, can be used to model task processing delays. Based on the superposition property of the Poisson process and the offloading interactions among the three layers of computing resources, the average task arrival rate for each layer can be calculated as

$$\lambda_d = \begin{cases} \sum_{t=1}^{T}(\lambda_t \times p_{d^* \to u,t}) & \text{if } d = u \\ \sum_{t=1}^{T}\sum_{d^* \in \{u,e\}}(\lambda_t \times p_{d^* \to e,t}) & \text{if } d = e \\ \sum_{t=1}^{T}(\sum_{d^* \in \{u,e,c\}} \lambda_t \times p_{d^* \to c,t}) & \text{if } d = c \end{cases}. \quad (4)$$

Therefore, the average time consumption of each task at each layer, encompassing both the queuing and the execution times, can be calculated as

$$L_{d,t}^{com} = \frac{C(N_d, \rho_d)\rho_d}{\lambda_d(1-\rho_d)} + \frac{\xi_t}{\varsigma_d}, \quad (5)$$

where $\rho_d$ can be calculated as $\rho_d = \frac{\lambda_d}{N_d \alpha_d \varsigma_d}$. $C(N_d, \rho_d)$ is known as Erlang's C formula, which can be calculated as

$$C(N_d, \rho_d) = \frac{\frac{(N_d\rho_d)^{N_d}}{N_d!}}{\sum_{k=0}^{N_d-1}\frac{(N_d\rho_d)^k}{k!} + \frac{(N_d\rho_d)^{N_d}}{N_d!(1-\rho_d)}}. \quad (6)$$

At the same time, the energy consumption of $D_d$ in each layer during the execution of computing tasks can be calculated as follows [34]:

$$E_{d,t}^{com} = \kappa_d \xi_t \varsigma_d^2, \quad (7)$$

where $\kappa_d$ is depending on the chip architecture.

### D. Service Model

1) Computing occupancy: As a direct indicator of computing resource utilization, it is quantified by the number of devices participating in computing resource sharing while maintaining the provision of all network services. It aims to enhance the overall breadth of resource engagement across the system, which can be defined as follows:

$$O_{tot} = \sum_{n=1}^{D} x_n, \quad (8)$$

with

$$x_n = \begin{cases} 1 & \text{Device } d_n \text{ is occupied} \\ 0 & \text{Device } d_n \text{ is free} \end{cases}. \quad (9)$$

2) Privacy entropy: More applicable computing offloading can be achieved by constructing three layers computing structures. However, the decentralization of computing devices introduces privacy leakage risks during task offloading. Privacy entropy, a method of quantifying privacy security, is used to measure the security of information transmission [35]. Higher privacy entropy indicates more disordered data, preventing attackers from making inferences and thereby ensuring secure information transmission.

The types of offloaded data vary according to the computing tasks. Thus, in BECS, we set the relationship between $T_t$ and $p_{d^* \to d,t}$ to be a one-to-one correspondence, and the privacy entropy of $T_t$ in $D_d$ can be calculated as

$$H_d = -\sum_{t=1}^{T} p_{d^* \to d,t} \log_2 p_{d^* \to d,t}. \quad (10)$$

3) Load balancing: As a key metric for evaluating computing device workloads, load balancing aims to equalize each device's load to the average while ensuring an equitable and efficient distribution of workload based on device capabilities. Concentrating multiple tasks on a single device reduces computational efficiency and increases energy consumption. The standard deviation of tasks and the computing capabilities of devices can indicate whether they are load-balanced [36]. Therefore, the load balancing of $D_d$ can be calculated as

$$B_d = \sqrt{\frac{\sum_{d=1}^{D}\left(\lambda_d \xi_t - \varsigma_d\right)^2}{D}}. \quad (11)$$

4) Sharing revenue: A suitable revenue can encourage the participation of computing devices in computing resource sharing, thereby enhancing the overall resource utilization. The logarithmic utility function is used to quantify the sharing revenue of $D_d$, which can be calculated as

$$R_d = \ln\left(1 + \beta_1 \psi_d \frac{\xi_t}{\varsigma_d} + \beta_2 \varsigma_d\right). \quad (12)$$

### E. Premise Assumptions

The assumptions and requirements specified below are applied consistently throughout this article unless otherwise stated.

1) The proposed computing resource sharing operates within a metropolitan area, where edge and cloud nodes constitute a well-connected computing infrastructure, linked by a high-capacity optical backbone.

2) Computing tasks can be offloaded from the outer to the inner layer as depicted in Fig. 1, or within the same layers, provided that the computing capacity of CRP exceeds that of CRR.

3) Each user device is equipped with a single antenna, facilitating real-time communication with the 6GBS.

4) The communication latency on the optical backbone is assumed negligible in comparison to the dominant and more variable delays introduced by wireless access and task computing [37].

4) Based on the permissioned blockchain, user registration information is visible only to 6GBSs within the blockchain, whereas the computing resources status is visible to all users.

5) All users securely deliver their keys through a secure channel.

## IV. Computing Allocation Based on NSGA-III-KDR

In this section, we first introduce the proposed computing allocation MOOP. Subsequently, we explain the principle of kernel distance-based Dominance Relation and the optimization process based on NSGA-III-KDR.

### A. MOOP of Computing Allocation

The core of our computing allocation mechanism is formulated as a MOOP. The goal is to intelligently assign computing tasks to available resources in the 6G CPN, balancing multiple, often conflicting, performance objectives. These objectives span from system efficiency and cost to user-centric metrics like privacy and service quality.

First, the system cost objectives are defined, which include the total time consumption ($L_{tot}$) and total energy consumption ($E_{tot}$), which serve as key metrics for evaluating computing efficiency and energy sustainability in green 6G networks.

**Total Time Consumption ($L_{tot}$):** As formulated in (13), this metric captures the overall latency. It is the sum of the computation time ($L^{com}$) across all three layers and the wireless transmission time ($L^{tra}$) incurred when user devices offload tasks.

$$L_{tot} = \sum_{t=1}^{T} \sum_{d \in \{u,e,c\}} p_{d^* \to d,t} L_{d,t}^{com} + \sum_{t=1}^{T} \sum_{u=1}^{U} p_{u \to d,t} L_{u,t}^{tra}, \tag{13}$$

**Total Energy Consumption ($E_{tot}$):** Similarly, as shown in (14), this objective accounts for the total energy spent. It comprises the energy for task execution ($E^{com}$) on the designated devices and the energy for wireless transmission ($E^{tra}$) from user devices.

$$E_{tot} = \sum_{t=1}^{T} \sum_{d \in \{u,e,c\}} p_{d^* \to d,t} E_{d,t}^{com} + \sum_{t=1}^{T} \sum_{u=1}^{U} p_{u \to d,t} E_{u,t}^{tra}. \tag{14}$$

Next, a set of objectives is defined to address key aspects of system performance, including resource utilization ($O_{ave}$), privacy entropy ($H_{ave}$), load balancing ($B_{tot}$), and CRP revenue ($R_{tot}$).

**Average Resource Utilization ($O_{ave}$):** As formulated in (15), the average resource utilization quantifies the extent of resource engagement by representing the ratio of occupied devices ($O_{tot}$) to the total number of devices ($D$).

$$O_{ave} = \frac{1}{D} O_{tot}, \tag{15}$$

**Average Privacy Entropy ($H_{ave}$):** As shown in (16), the average privacy entropy serves as an indicator of user privacy protection. A higher individual entropy ($H_{d,t}$) reflects a more disordered and less predictable task allocation pattern, thereby increasing the difficulty for adversaries to infer sensitive information.

$$H_{ave} = \frac{1}{D} \sum_{d \in \{u,e,c\}} H_{d,t}, \tag{16}$$

**Total Load Balancing ($B_{tot}$):** As expressed in (17), the total load balancing metric is computed by aggregating the individual load balance indicators $B_d$ across all devices. The underlying indicator $B_d$, defined in (11), corresponds to the standard deviation of workloads, where a smaller value indicates a more balanced load distribution and contributes to the mitigation of computational bottlenecks.

$$B_{tot} = \sum_{d \in \{u,e,c\}} B_d, \tag{17}$$

**Total CRP Revenue ($R_{tot}$):** As given in (18), the total CRP revenue is obtained by aggregating the individual revenues ($R_d$) of all resource-providing devices. The individual revenue function $R_d$, detailed in (12), is formulated as a logarithmic utility to model economic incentives, capturing diminishing returns and promoting broad participation.

$$R_{tot} = \sum_{d \in \{u,e,c\}} R_d. \tag{18}$$

Assigning tasks to free computing devices can effectively improve resource utilization. However, focusing solely on utilization improvement is insufficient, as the objectives within the 6G CPN are inherently conflicting. While individual metrics such as time and energy consumption are well-established [2], [5], [16], the novelty of this study lies in holistically modeling, for the first time, the complex interplay among performance, efficiency, security, and economy. For instance, minimizing time and energy consumption often leads to workload centralization, which directly conflicts with the goal of equitable load balancing. Similarly, maximizing privacy entropy may necessitate non-optimal routing that degrades performance, whereas a strategy aimed at maximizing provider revenue may create economic tensions that impact overall resource occupancy. This landscape of competing demands necessitates a solution that seeks the best balanced compromise. Therefore, to scientifically balance these competing objectives, the MOOP is formulated as follows:

$$\max \left\{ O_{ave}, H_{ave}, R_{tot} \right\}, \tag{19}$$

$$\min \left\{ L_{tot}, E_{tot}, B_{tot} \right\}, \tag{20}$$

subject to:

$$\lambda_d \leq \alpha_d \varsigma_d, \tag{21}$$

$$0 < p_u \leq p_u^{\max}, \tag{22}$$

$$L_{d,t}^{com} \leq \tau_t, \tag{23}$$

$$E_{d,t}^{com} \leq E_d^{max}, \tag{24}$$

$$\varsigma_u < \varsigma_e < \varsigma_c, \varsigma_{d^*} < \varsigma_d, \tag{25}$$

$$\lambda_d \xi_d \leq \varsigma_d \tag{26}$$

Constraint (21) states that the actual task processing rate of the computing device must not exceed its service rate. Constraint (22) restricts the maximum transmit power of the user device. Constraint (23) guarantees that the task execution time does not exceed its deadline. Constraint (24) specifies that the computing energy consumption should not exceed the device's maximum available energy. Constraint (25) differentiates the computing capacities across three layers and specifies task offloading from devices with lower capacity to those with higher capacity. Constraint (26) mandates that the average rate of incoming computing workload to a device must not exceed its processing capacity.

### *B. Kernel Distance-based Dominance Relation*

NSGA-III, an excellent evolutionary algorithm, can achieve fast global searches with quality assurance, preventing MOOP from settling into local optimality. Additionally, it tackles high-dimensional problems by preserving population diversity

through uniformly distributed reference points [38]. In practical MOOP, maximizing population diversity while ensuring convergence is one of the best ways to quickly and effectively obtain the optimal solution. Consequently, NSGA-III-KDR is proposed to balance the aforementioned five objectives.

The performance of Pareto dominance relation-based algorithms often exhibits serious dimensionality implications due to dominance resistance in MOOP with more than three objectives [39]. To address this, Tian *et al.* [17] proposed a strengthened dominance relation (SDR) to enhance NSGA-II. Considering that NSGA-III builds upon NSGA-II by introducing reference points, we propose a kernel distance-based dominance relation (KDR) in NSGA-III-KDR. It builds on the theoretical foundation of SDR to replace the original Pareto dominance relation in NSGA-III. Specifically, if solution $X_1$ dominates solution $X_2$ in KDR, then (27) is satisfied.

$$\begin{cases} d_k(X_1) < d_k(X_2), & \theta_{X_1X_2} \le \overline{\theta} \\ d_k(X_1) \cdot \frac{\theta_{X_1X_2}}{\overline{\theta}} < d_k(X_2), & \theta_{X_1X_2} > \overline{\theta} \end{cases}, \tag{27}$$

where $\theta_{X_1X_2}$ represents the acute angle between the two candidate solutions $X_1$ and $X_2$, which can be calculated as $\theta_{X_1X_2} = \arccos\big(F(X_1), F(X_2)\big)$, and $\overline{\theta}$ denotes the size of the niche to which each candidate solution belongs, and can be set to the $\left\lfloor \frac{|P|}{2} \right\rfloor$-th minimum element of

$$\left\{ \min_{q \in P \setminus \{p\}} \theta_{pq} | p \in P \right\}, \tag{28}$$

where $\theta_{pq}$ is the acute angle between $p$ and $q$ of any pair of candidate solutions.

The kernel distance is chosen because, as the number of objectives increases, neither the Euclidean distance nor the Mahalanobis distance can accurately reflect the crowding degree between individuals [40]. Therefore, NSGA-III-KDR utilizes the kernel distance from the point $X$ to the ideal point $P_{ide}$ to measure the similarity between them in handling high-dimensional problems [41]. It can be calculated as follows:

$$d_k(X) = \sqrt{2 - 2\exp\left(-\frac{\sum_{i=1}^{m}(\|f_i(X) - P_{ide}\|^2)}{2\sigma^2}\right)}. \tag{29}$$

### C. Encoding of Computing Resources

Evolutionary algorithms utilize the concept of population evolution to tackle practical MOOPs. Here, individuals in a population are represented by a series of numbers, each mapped to potential solutions of a MOOP through specific encoding. Thus, encoding is crucial for implementing population evolution in practical MOOPs.

In BECS, the occupancy status of computing devices is encoded as genes, while the computing resources involved in sharing are encoded as chromosomes, constituting the entire CRP for evolution. As illustrated in Fig. 2, a gene value of 0 indicates a free computing device, whereas a value of 1 signifies that this device is occupied. This method allows BECS to integrate various types of computing resources, thereby building a generalized computing resource sharing platform. Since the occupancy status of computing devices directly correlates with gene encoding, and chromosomes relate to computing allocation strategies, dynamic and fine-grained updates of computing resources are enabled.

**Algorithm 1** Population Evolution

**Input:** The initialized population $X^1$, the total number of iterations $TI$, and the population size $PS$
**Output:** the final population $X^{TI}$

**for** $j = 1$ to $TI$ **do**
  **for** the chromosome in $X^j$ **do**
    Evaluate objective functions by (13), (14), (15), (16), (17) and (18);
  **end for**
  $X^j$ conducts crossover and mutation operation to produce $S^j$,
  Generate a merged population $Y^j = X^j \bigcup S^j$ with $2PS$ population size;
  **for** merged population $Y^j$ **do**
    Execute Non-KDR-dominated sort;
  **end for**
  Determine the reference point on the hyper-plane;
  Associate chromosomes and reference points;
  Evaluate niche-preservation operation, generate a new population $X^{j+1}$;
**end for**
**return** $X^{TI}$

### D. Population Evolution

To adapt to the dynamic 6G CPN environment, in which task arrivals and resource availability fluctuate over time, BECS operates not as a one-time process but as an adaptive control loop. When a CRR requests computing resources in the network, the computing resource sharing mechanism is triggered. Subsequently, the system executes the NSGA-III-KDR-based computing allocation scheme to match the optimal CRP according to the current network state. NSGA-III-KDR essentially follows the algorithmic framework of NSGA-III, as shown in Algorithm 1. During the evolution process, the chromosomes in the initial population $X^1$ generate entirely new chromosomes (computing allocation strategies) through crossover and mutation operations. The crossover operation enhances chromosome diversity, while the mutation operation, under specific conditions, modifies individual genes to

| computing resources | $U_1$ | ... | $U_u$ | ... | $U_U$ | $E_1$ | ... | $E_e$ | ... | $E_E$ | $C_1$ | ... | $C_c$ | ... | $C_C$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| chromosome | 1 | ... | 0 | ... | 1 | 0 | ... | 1 | ... | 1 | 1 | ... | 0 | ... | 0 |

Fig. 2. Encoding of computing resources.

seek those with higher adaptability. In each iteration, the population generated by crossover and mutation merges with the original, then executing a Non-KDR-dominated sort that replaces the non-dominated sort in NSGA-III, thereby effectively enhancing the diversity of the computing resource population. Subsequently, by associating chromosomes with reference points in the hyper-plane and executing the Niche-Preservation Operation, the next generation population can be generated. After $TI$ iterations of NSGA-III-KDR, the final population $X^{TI}$, containing the ultimate computing allocation strategy, is obtained.

### E. Optimum Selection

The final population $X^{TI}$ comprises a set of the best feasible solutions for computing allocation, termed Pareto solutions. However, in practical computing allocation problems, the CRR needs to match only a specific CRP. Therefore, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), in conjunction with entropy weighting, is utilized to evaluate the most optimal solution among many Pareto solutions. As a method approximating the ideal solution, TOPSIS rapidly identifies the optimal solution by ranking all solutions according to their distance from the positive (negative) ideal solution. Additionally, entropy weighting helps eliminate the arbitrariness of subjectively determined weights. This combination enables an objective and fair determination of the optimal computing allocation from $X^{TI}$. The specific algorithm is detailed in [42].

By executing NSGA-III-KDR in BECS, CRP can set $\psi_d$ more reasonably by analyzing the optimal solution. Simultaneously, CRR can select the most appropriate CRP for task offloading based on the optimal solution. Consequently, both parties engaged in computing resource sharing achieve reliable, dynamic, and secure computing allocation and trading.

## V. Secure Computing Trading with Privacy Protection

In this section, we focus on the principles of the proposed pseudonym scheme and secure computing trading under pseudonyms in BECS.

### A. System Initialization

In this phase, upon inputting the security parameter $\lambda$, the system administrator selects a secure elliptic curve $\mathcal{E} : y^2 = x^3 + ax + b \mod p$. The group $\mathcal{G}$ is an additive elliptic curve group with order $q$ and generator $g$ defined over $\mathcal{E}$, where $p$ and $q$ are two large prime numbers. Subsequently, 6GBS $B_b$ chooses a random number $b \in Z_q^*$ as its private keys and caculates the public keys as $bg$. Additionally, it selects a collision-resistant hash function $\mathcal{H}() : \{0,1\}^* \to \{0,1\}^l$ and makes $\{\mathcal{E}, \mathcal{G}, a, b, p, q, \mathcal{H}()\}$ publicly available in blockchain.

### B. Computing Devices Registration

All devices must be registered upon entering the system. $D_d$ first selects $d \in Z_q^*$ as the private key and then calculates the public key as $dg$. Subsequently, it sends its real identity $ID_d$ and the public key $dg$ to the nearest $B_b$ via a secure channel. After receiving the message from $D_d$, $B_b$ stores $\mathcal{H}(\mathcal{H}(ID_d) \parallel dg)$ in SC, while simultaneously generating and sending $D_d$'s identity certificate: $Cert_d = Sig_b(\mathcal{H}(ID_d) \parallel dg)$.

### C. Pseudonym Generation

When $D_d$ participates in computing resource sharing, it must first send its public key $dg$ and identity certificate $Cert_d$ to the nearest $B_b$. After $B_b$ verifies the legitimacy of $D_d$'s identity, it generates a pseudonym for $D_d$, as detailed in Algorithm 2. Based on the Schnorr protocol [43], $D_d$ proves possession of the private key $d$ to $B_b$ without revealing it, by utilizing the public parameter $g$ and random numbers. $B_b$ verifies the legitimacy of $D_d$'s identity by confirming the correctness of $D_d$'s proof and response. Utilizing non-interactive zero-knowledge protocol, $D_d$ acquires a pseudonym $(x, y)$ for communication from $B_b$ while preserving privacy.

**Algorithm 2** Pseudonym Generation

**Input:** $D_d$'s private key $d$, public key $dg$.
**Output:** $D_d$'s pseudonym $(x, y)$.

1: $D_d \longrightarrow B_b$: Sends $(\tilde{x} = g, \tilde{y} = dg)$.
2: $B_b \longrightarrow D_d$: Chooses $\gamma \in Z_q^*$, sends $x = \gamma\tilde{x}$.
3: $D_d \longrightarrow B_b$: Calculates $y = dx$, then chooses $\delta \in Z_q^*$, and calculates commitment $K = \delta x$, challenge $\epsilon = \mathcal{H}(x \parallel y \parallel K)$, response $M = \delta + \epsilon d \mod q$, sends $y, K$, and $M$.
4: $B_b$: Calculates $\epsilon' = \mathcal{H}(x \parallel y \parallel K)$ to validates $Mx \stackrel{?}{=} K + \epsilon' y$.
5: **if** the verification is correct **then**
6: $B_b$: Stores $D_d$'s pseudonym $(x, y)$ in SC.
7: $D_d$: Stores its pseudonym $(x, y)$, and the transcript $T = (K, M)$.
8: **end if**
9: **return** $(x, y)$

The pseudonym $(x, y) = (\gamma g, \gamma dg)$, constructed using $D_d$'s public key and random numbers, cannot establish its legitimacy and thus requires $B_b$ to issue the corresponding certificate, as shown in Algorithm 3. Using a non-interactive zero-knowledge protocol, $B_b$ generates a commitment and a response that include its private key. $D_d$ then verifies these to confirm that $B_b$, who possesses the private key $b$, generated the certificate. Upon successful verification, $D_d$ stores the certificate.

**Algorithm 3** Certificate Issuance

**Input:** $D_d$'s pseudonym $(x, y)$; $B_b$'s private key $b$, public key $h = bg$.
**Output:** $D_d$'s pseudonym certificate $\varpi$.

1: $B_b \longrightarrow D_d$: Chooses $\phi \in Z_q^*$, and calculates commitment $O = \phi g$, challenge $\zeta = \mathcal{H}(x \parallel y \parallel O)$, response $P = \phi + \zeta b \mod q$, sends $O$ and $P$.
2: $D_d$: Calculates $\zeta' = \mathcal{H}(x \parallel y \parallel O)$ to verifies $Pg \stackrel{?}{=} O + \zeta' h$, if verification succeeds, then store the certificate $\varpi = (O, P)$.
3: **return** $\varpi$

### D. Pseudonymous Computing Trading

Upon entering the system, CRP synchronizes its computing device information and status within blockchain. CRR sends a computing request, and NSGA-III-KDR is utilized to match the resource-sharing parties. CRP ($D_2$) accepts a task offloading request from CRR ($D_1$). Initially, both parties must verify the pseudonym and certificate according to Algorithm 4. $D_1$ uses transcript $T_1$ to prove the authenticity of the pseudonym to $D_2$ and uses the certificate $\varpi_1$ to prove the legitimacy of the certificate to $D_2$. After both verifications succeed, $D_2$ confirms $D_1$'s identity, and similarly, $D_1$ confirms $D_2$'s identity. Once confirming their identities, CRR offloads its computing tasks to CRP, which then updates its resource status and delivers the computing results to CRR within the deadline. Following the payment of corresponding fees by CRR, both parties finalize the trading.

**Algorithm 4** Identity Verification

**Input:** $D_1$'s pseudonym $(x_1, y_1)$, transcript $T_1 = (K_1, M_1)$, and certificates $\varpi_1 = (O_1, P_1)$ are issued by $B_1$; $B_1$'s public keys $h_1 = b_1 g$.

**Output:** $D_2$ accepts or rejects $D_1$'s identity.

1: $D_1 \longrightarrow D_2$: Sends transcript $T_1 = (K_1, M_1)$ and certificates $\varpi_1 = (O_1, P_1)$ under the pseudonym $(x_1, y_1)$.

2: $D_2$: Calculates $\epsilon_1' = \mathcal{H}(x_1 \parallel y_1 \parallel K_1)$ to verifies the authenticity of pseudonym: $M_1 x_1 \stackrel{?}{=} K_1 + \epsilon_1' y_1$.
Then Calculates $\zeta_1' = \mathcal{H}(x_1 \parallel y_1 \parallel O_1)$ to verifies the legitimacy of pseudonym and certificate: $P_1 g \stackrel{?}{=} O_1 + \zeta_1' h_1$.

3: If all verifications are correct, $D_2$ accepts $D_1$'s identity. Else, $D_2$ rejects $D_1$'s identity.

4: **return** $D_2$ accepts or rejects $D_1$'s identity.

### E. Block Generation and Pseudonymous Update

Once a computing resource sharing trade is completed, the transaction is finalized by generating a block on the system's underlying permissioned blockchain. Within this framework, the 6GBSs serve as trusted consensus nodes that validate transactions and maintain the ledger. The selection of the 6GBS responsible for block generation is determined by the Proof of Trust and Adjustment (PoTA) consensus mechanism. PoTA is a lightweight and efficient protocol that we introduced in our previous work [27].

Specifically, the PoTA protocol selects a bookkeeping node based on a dynamically calculated score. This score comprehensively evaluates both a node's global trust, derived from historical interaction ratings, and its relevant resource service attributes. This mechanism ensures a balance between security and efficiency, making it a practical and feasible consensus mechanism for the dynamic 6G CPN. The 6GBS with the highest score is authorized to aggregate all relevant information, such as transaction details, digital signatures, and resource status updates, into a new block. This block is then broadcast for validation and subsequently appended to the blockchain.

Subsequently, the CRP must update its computing resource status, after which the new state information is broadcast and synchronized across the entire blockchain. Before re-engaging in computing resource sharing, CRP and CRR need to request new pseudonyms and certificates from the nearby 6GBS to ensure the user's identity remains unlinkable. Similarly, the 6GBS verifies the legitimacy of the CRP/CRR identities through Algorithm 4 and then issues CRP/CRR new pseudonyms and certificates.

## VI. Security Analysis and Discussion

In this section, we analyze the security of BECS and discuss the computational complexity of NSGA-III-KDR.

### A. Threat Model and Adversary Assumptions

To facilitate formal analysis, a comprehensive threat model is adopted, encompassing two primary types of adversaries:

- **External Adversary ($\mathcal{A}$):** An entity that is not a legitimate participant in the 6G CPN. Its capabilities are limited to eavesdropping on communication channels, with the objective of compromising user privacy via traffic analysis or forging messages to obtain unauthorized access.
- **Malicious Internal Node ($\mathcal{B}$):** A registered participant within the BECS system (e.g., a compromised CRP or CRR) may comply with protocol specifications but still attempt to exploit its privileges to deanonymize transaction partners, repudiate actions, or disrupt system operations by injecting false information.

The security analysis relies on standard cryptographic assumptions, including the computational hardness of the discrete logarithm problem and the collision resistance of the hash function $\mathcal{H}$.

### B. System Security

The permissioned blockchain architecture establishes a foundational layer of defense against both external and internal adversaries.

**1) Access Control and Authentication:** This mechanism constitutes the first line of defense against the external adversary $\mathcal{A}$. By mandating that all devices register through the RCC and obtain verifiable credentials, the system effectively prevents unauthorized entities from joining the network, thereby mitigating risks such as impersonation and unauthorized access to resources.

**2) Integrity and Non-repudiation:** Blockchain immutability serves as a direct countermeasure to threats posed by the malicious internal node $\mathcal{B}$. It ensures that malicious participants cannot tamper with transaction records to alter agreed-upon terms or repudiate their actions. All activities are irreversibly recorded, providing a verifiable audit trail that enforces accountability.

**3) Resilience and Availability:** The blockchain's distributed architecture is inherently resilient to targeted attacks by a powerful adversary $\mathcal{B}$ capable of compromising one or several 6GBSs. By eliminating single points of failure, the system

ensures service continuity and preserves data integrity even in the presence of partial compromise.

**4) Resistance to Statistical Inference Attacks:** The privacy entropy maximization objective in the proposed MOOP provides a statistical defense against inference attacks launched by the malicious internal node $\mathcal{B}$. By promoting diverse and unpredictable task offloading behaviors, this mechanism obfuscates user behavioral patterns, thereby complementing the direct identity unlinkability provided by the pseudonym scheme.

### C. Pseudonym and Certificate Security

The pseudonym scheme is equipped with specific cryptographic properties to mitigate the capabilities of both adversaries, $\mathcal{A}$ and $\mathcal{B}$, across a range of attack scenarios.

**1) Uniqueness and Authenticity:** This property mitigates impersonation attacks launched by both adversaries, $\mathcal{A}$ and $\mathcal{B}$. The zero-knowledge protocol-based generation process ensures that only the legitimate owner of a private key $d$ can generate a valid pseudonym and its corresponding proof. As a result, it becomes computationally infeasible for an adversary to forge a valid pseudonym for another user, thereby ensuring that each identity within a transaction is authentic.

**2) Unforgeability:** Certificate unforgeability directly prevents both adversaries from obtaining illegitimate trust within the system. Since generating a valid certificate requires access to the private key $b$ of a 6GBS, no adversary can fabricate a fraudulent certificate, thereby preventing impersonation of certified and trustworthy nodes.

**3) Unlinkability:** This property plays a critical role in defending against the privacy-compromising objectives of both adversary types.

- Against the external adversary $\mathcal{A}$, the use of fresh, random nonces $\gamma$ in each session ensures that pseudonyms appear as independent and uncorrelated values on the public channel. This renders traffic analysis and activity inference through eavesdropping ineffective.
- Against the malicious internal node $\mathcal{B}$, the zero-knowledge-based interactions and session-specific pseudonyms prevent it from linking its current transaction partner to any previous or future transactions. This prevents malicious peers from constructing a user's long-term behavioral profile.

**4) Traceability:** This feature is specifically designed as a deterrent and mitigation strategy against the malicious internal node $\mathcal{B}$. While unlinkability safeguards users from peer inference, authorized traceability ensures that when malicious behavior is verified, the 6GBSs can revoke the anonymity of the offending node and expel it from the system. This accountability mechanism deters internal misbehavior without compromising user anonymity.

### D. Computational Complexity of NSGA-III-KDR

In BECS, NSGA-III-KDR uses Non-KDR-dominated sort instead of Non-Dominated Sort from NSGA-III to enhance solution diversity, while maintaining the same computational complexity. Namely, with $L$ optimization objectives and a population size of $X$, NSGA-III-KDR exhibits a computational complexity of $\mathcal{O}(LX^2)$. Specifically, calculating the kernel distance for each solution and determining the angle between any two solutions each incur a complexity of $\mathcal{O}(L)$. Each solution's distance to the ideal point is calculated individually, contributing to a complexity of $\mathcal{O}(LX)$. For dominance judgment, each pair of solutions undergoes one angle calculation and comparison, leading to an overall complexity of $\mathcal{O}(LX^2)$. Thus, the computational complexity of each iteration can be approximated as $\mathcal{O}(LX^2)$. Compared to NSGA-III-based scheme [15], NSGA-III-KDR offers more diverse solutions, improving the probability of superior computing allocation strategies.

## VII. Simulation Results and Analysis

In this section, we initially compare the performance of the proposed NSGA-III-KDR with NSGA-III and MOEA/D applied in computing allocation, as well as the NSGA-II-SDR that inspired us. Subsequently, we test the performance of four algorithms in optimizing the proposed computing allocation MOOP. Finally, we simulate the performance of the proposed pseudonym scheme. The configurations of critical parameters are detailed in Table III.

TABLE III
Key Parameters

| Parameters | Values |
|---|---|
| Main physical machine | Intel i7-12700@2.1GHz with 32GB RAM |
| Operating systems | Windows 11 & Ubuntu 24 |
| Number of devices | $\mathbb{U}$: 300; $\mathbb{E}$: 200; $\mathbb{C}$: 100 |
| Transmit power | 20∼30 dbm |
| Noise power | -97 dBm |
| Channel gain | 2.15 dBi |
| Bandwidth | 20 MHz |
| Average arrival rate | 100 tasks/s |
| Computing Capacity [44], [45] | $\mathbb{U}$: 0.6∼10 TFLOPS; $\mathbb{E}$: 10∼1000 TFLOPS; $\mathbb{C}$: >1000 TFLOPS |
| Effective capacitance coefficient [46] | $10^{-29}$ |
| Data size | 500∼3000 KB |
| CPU cycles per byte | 1000 cycles/byte |
| Cryptographic libraries | PBC and Openssl |
| Probability of offloading | $\mathbb{U}$: 0.5, $\mathbb{E}$: 0.3, $\mathbb{C}$: 0.2 |
| Service price | $\mathbb{U}$: 0.1, $\mathbb{E}$: 1, $\mathbb{C}$: 2 |
| $\beta_1$, $\beta_2$; $\sigma$ | 0.6, 0.4; 1 |

### A. Simulation of Proposed NSGA-III-KDR

In this part, we compare the proposed NSGA-III-KDR with state-of-the-art evolutionary algorithms NSGA-III [15], MOEA/D [16], and NSGA-II-SDR [17], utilizing the PlatEMO platform [47]. The widely used SDTLZ [38], MaF [48], and SMOP [49] test suites are employed as the benchmarks. Meanwhile, IGD [50] and PD [51] are selected as performance evaluation metrics. IGD comprehensively measures the convergence and diversity of the algorithms, where a smaller IGD

TABLE IV
IGD AND PD VALUE OF NSGA-III, MOEA/D, NSGA-II-SDR, AND NSGA-III-KDR ON SDTLZ1, SDTLZ2, MAF1, MAF2, SMOP1, AND SMOP2 WITH 5, 10, AND 15 OBJECTIVES. THE BEST RESULT IN EACH ROW IS HIGHLIGHTED.

| Problem | M | IGD | | | | PD | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | NSGA-III | MOEA/D | NSGA-II-SDR | NSGA-III-KDR | NSGA-III | MOEA/D | NSGA-II-SDR | NSGA-III-KDR |
| SDTLZ1 | 5 | 4.0919e-1 = | 1.1456e+0 - | 7.2486e-1 - | **4.072e-1** | 1.0645e+7 - | 5.7924e+6 - | 1.8399e+7 - | **4.7052e+7** |
| | 10 | **1.7833e+1 +** | 3.1719e+1 + | 3.7045e+1 + | 1.3076e+2 | **2.3247e+10 +** | 4.3704e+9 + | 7.6415e+8 + | 1.0865e+10 |
| | 15 | **6.1421e+2 +** | 8.2078e+2 + | 8.4100e+2 + | 6.4468e+3 | **2.8108e+12 +** | 4.7472e+11 + | 9.5939e+7 - | 5.2570e+9 |
| SDTLZ2 | 5 | 1.1871e+0 = | 3.1693e+0 - | 4.3480e+0 - | **1.1838e+0** | 1.9000e+7 - | 1.1551e+7 - | 1.6919e+4 - | **9.6971e+7** |
| | 10 | 6.8214e+1 - | 1.1354e+2 - | 1.4962e+2 - | **6.3516e+1** | 6.3070e+10 - | 8.2536e+9 - | 6.7541e+8 - | **1.0334e+10** |
| | 15 | 2.3123e+3 - | 3.6367e+3 - | 3.6612e+3 - | **2.1131e+3** | 1.3576e+13 - | 4.7141e+11 - | 8.7418e+8 - | **2.9897e+11** |
| MaF1 | 5 | 2.2118e-1 - | 1.6710e-1 - | **1.4031e-1 +** | 1.6200e-1 | 1.7519e+7 - | 3.8550e+6 - | 1.9648e+7 - | **2.4256e+7** |
| | 10 | 3.1732e-1 - | 3.9154e-1 - | **2.9201e-1 +** | 3.1425e-1 | 8.9830e+9 - | 1.5818e+9 - | 6.1301e+9 - | **1.2171e+10** |
| | 15 | **3.8109e-1 +** | 4.7345e-1 - | 4.2790e-1 - | 4.0302e-1 | 1.9765e+11 = | 5.7436e+10 - | 5.9691e+10 - | **2.0030e+11** |
| MaF2 | 5 | 1.4175e-1 - | 1.5433e-1 - | 1.3740e-1 - | **1.2311e-1** | 1.8852e+7 - | 1.3030e+7 - | 1.6372e+7 - | **2.1218e+7** |
| | 10 | 2.7622e-1 - | 3.0954e-1 - | 4.0882e-1 - | **2.5215e-1** | 8.5156e+9 - | 3.8250e+9 - | 8.3977e+9 - | **1.3398e+10** |
| | 15 | **3.2353e-1 +** | 3.8517e-1 + | 5.6897e-1 - | 4.6420e-1 | **2.5275e+11 +** | 1.1452e+11 - | 2.4382e+10 - | 1.3115e+11 |
| SMOP1 | 5 | 1.5928e-1 - | 2.2883e-1 - | 3.4716e-1 - | **1.3947e-1** | 3.6432e+6 - | 3.7787e+6 - | 6.8850e+6 - | **2.5642e+7** |
| | 10 | 4.2414e-1 - | 3.9985e-1 - | 4.9668e-1 - | **2.8703e-1** | 2.0785e+9 - | 9.4496e+8 - | 1.9355e+9 - | **1.1620e+10** |
| | 15 | 4.0587e-1 + | 4.5602e-1 + | **4.0515e-1 +** | 5.2449e-1 | 8.5548e+10 - | 5.5518e+9 - | 7.0498e+10 - | **1.0424e+11** |
| SMOP2 | 5 | 3.8732e-1 - | 5.8495e-1 - | 5.4909e-1 - | **3.6202e-1** | 6.2126e+6 - | 6.4898e+6 - | 1.1547e+7 - | **3.9017e+7** |
| | 10 | 9.1827e-1 - | 5.8728e-1 - | 8.6089e-1 - | **5.5237e-1** | 3.7243e+9 - | 1.8461e+9 - | 3.3883e+9 - | **2.1840e+10** |
| | 15 | **6.2944e-1 +** | 7.2789e-1 + | 6.8003e-1 + | 9.1024e-1 | 1.7008e+11 = | 1.1430e+10 - | 1.3620e+11 - | **2.0211e+11** |
| +/-/= | | 6/10/2 | 5/13/0 | 6/12/0 | | 3/13/2 | 2/16/0 | 1/17/0 | |

value indicates better performance. PD primarily reflects the diversity of the algorithms, with a larger PD value indicating greater population diversity. The crossover probability is set to 1, the mutation probability is set to 1/D, and their distribution indicator is set to 20, where D represents the length of the decision variable. All three algorithms are executed 30 times on different test problems, and the average values are taken. The performance of the three algorithms is compared under different numbers of objectives and benchmarks, as shown in Table IV, where "+", "−", and "=" indicate that the result is significantly better, significantly worse, and statistically similar to that obtained by NSGA-III-KDR, respectively.

It can be concluded from the experimental results that NSGA-III-KDR has the strongest competitiveness, achieving the best IGD and PD numbers of 10 and 15, respectively. This demonstrates that NSGA-III-KDR has better performance compared to the other two algorithms, especially in terms of population diversity, offering a richer set of solutions for computing allocation.

### *B. Simulation of Proposed Computing Allocation Scheme*

In this part, we compare NSGA-III, MOEA/D, NSGA-II-SDR, and NSGA-III-KDR in optimizing the proposed computing allocation MOOP. To simulate a representative snapshot of the dynamic CPN environment, we initialize the population with 50% of the computing devices randomly occupied, then perform the optimization using each of the four algorithms separately. This simulation evaluates the algorithm's ability to obtain a high-quality allocation solution within a single decision epoch, demonstrating its effectiveness in adapting to the heterogeneous and resource-constrained conditions of that specific moment. We calculate the changes in computing resources between the final and initial populations, while ensuring algorithmic convergence. Due to the uncertainty of the evolutionary process, we evaluate the performance of the four algorithms using three statistical indicators. Specifically, the overall distribution is presented via box plots (showing the maximum, upper quartile, median, lower quartile, and minimum), the optimal values based on the TOPSIS method are denoted by triangles, and the average values are represented by squares. This multi-dimensional comparison enables a more comprehensive assessment of algorithm performance.

The improvement rate (IR) of computing resource utilization under four algorithms is shown in Fig. 3. Overall, except for MOEA/D, the other three algorithms demonstrate positive optimization in computing resource utilization. NSGA-III-KDR shows the most significant overall and average improvements, indicating superior performance and the most substantial enhancement in resource utilization. NSGA-III and NSGA-II-SDR follow, showing notable but lesser improvements. Although MOEA/D achieves the best optimal solution, its overall performance is suboptimal, characterized by some negatively optimized computing allocation strategies. This issue likely stems from MOEA/D's predefined fixed neighborhood structures, which may restrict its global search capability and lead to local optimality.

As resource utilization improved, Fig. 4 to Fig. 8 illustrate the changes in the other five objectives considered by the computing allocation MOOP. Except for load balancing, improvements in all other objectives are observed with the optimization based on NSGA-III-KDR, attributed to enhanced resource utilization. The superior diversity of NSGA-III-KDR facilitates a more even distribution of computing resources, as particularly evidenced by the significant increase in privacy entropy. Although NSGA-III exhibits similar trends, it is less effective than NSGA-III-KDR. Notably, with NSGA-III-KDR and NSGA-III, increased resource utilization results in higher delays and sharing revenue. Meanwhile, energy consumption

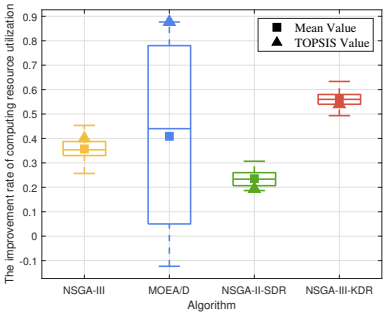


Fig. 3. The IR of computing resource utilization.

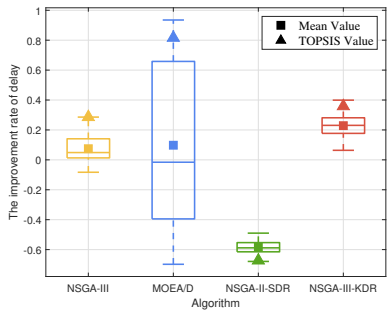


Fig. 4. The IR of time consumption.

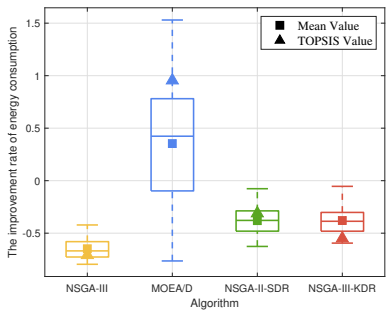


Fig. 5. The IR of energy consumption.

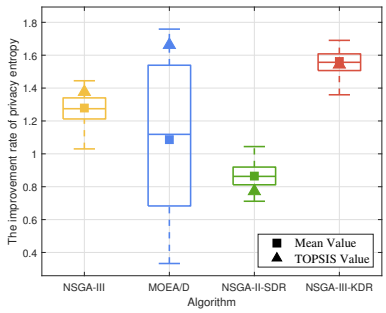


Fig. 6. The IR of privacy entropy.

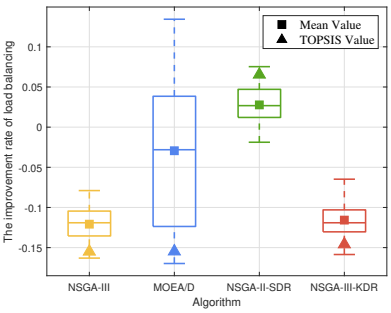


Fig. 7. The IR of load balancing.

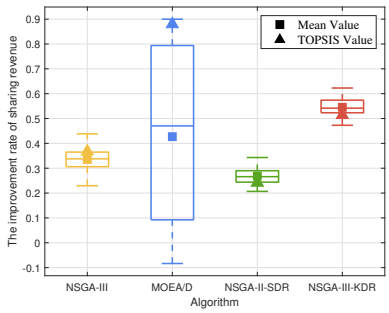


Fig. 8. The IR of sharing revenue.

decreases as more tasks are offloaded to $\mathbb{U}$, leveraging all available resources within 6G CPN comprehensively. The optimization strategy of NSGA-II-SDR favors offloading tasks to $\mathbb{E}$ and $\mathbb{C}$, significantly reduces delay, but uniquely results in positive growth in load balancing among the four algorithms. The optimal solution of MOEA/D surpasses all other algorithms, achieving the highest resource utilization and substantial improvements in all objectives, except for load balancing, and the decline in load balancing suggests a rational allocation of computing resources. However, the overall data of MOEA/D is the worst, with a negative optimization of 12.99%. Furthermore, although its median or average increase in resource utilization is less than that of NSGA-III-KDR, the more substantial improvement in load balancing suggests a less favorable overall evolution. Since the selection of the optimal solution using TOPSIS is random, not every iteration of MOEA/D yields a solution that outperforms those of other algorithms.

### C. Simulation of Proposed Pseudonymous Scheme

In this part, we evaluate the computational overhead of the proposed pseudonym scheme by comparing it with several existing pseudonymous schemes. Each operation and phase are tested 20 times separately, and the average value is recorded as the experimental data to eliminate the influence of hardware fluctuations during the operation. First, the BN curve [52] with a 128-bit security level is selected to implement the bilinear group. The execution time of basic cryptographic operations are shown in Table V. Generally, the execution times required for bitwise XOR and modular multiplication are significantly lower compared to other cryptographic operations, and can thus be disregarded. Moreover, we utilize SHA-256 as the hash function. Then, we assess the time costs associated with system initialization and pseudonym generation (SIPG), certificate issuance or message signing(CIMS), and identity or message verification (IDMV) in the pseudonym authentication process, comparing the proposed scheme with those in [53]–[56], as shown in Table VI.

The proposed scheme, which is based on the Schnorr protocol, primarily involves operations $T_{pm}^{ecc}$, $T_{pa}^{ecc}$, and $T_{ha}$ in the pseudonym authentication process. The total computational overload is $14T_{pm}^{ecc} + 4T_{pa}^{ecc} + 8T_{ha} = 8.664\ ms$. Conversely, Bagga *et al.*'s scheme [53] uses $T_{mtp}$ and $T_{bp}$ in the SIPG and IDMV, leading to a higher computational overload. Similarly, Shen *et al.*'s scheme [54], based on bilinear pairing, also incurs a higher computational overload. Although based on ECC, Yang *et al.*'s scheme [55] includes 17 times $T_{pm}^{ecc}$ within SIPG, leading to a significant computational overload. Wang *et al.*'s scheme [56], similar to the proposed scheme, provides the optimal computational overload in both SIPG and IDMV. However, a total of 16 times $T_{pm}^{ecc}$ yields a marginally greater computational overload than ours. Therefore, our scheme achieves the lowest computational overload compared to other

TABLE V
EXECUTION TIME OF CRYPTOGRAPHIC OPERATIONS

| Notation | Operation | Time (ms) |
|---|---|---|
| $T_{pm}^{ecc}$ | point multiplication in ECC | 0.618 |
| $T_{pa}^{ecc}$ | point addition in ECC | 0.001 |
| $T_{ha}$ | general hash | 0.001 |
| $T_{ep}$ | exponentiation | 0.011 |
| $T_{bp}$ | bilinear pairing | 1.456 |
| $T_{pm}^{bp}$ | point multiplication in bilinear pairing | 1.277 |
| $T_{pa}^{bp}$ | point addition in bilinear pairing | 0.003 |
| $T_{mtp}$ | hash-to-point in bilinear pairing | 5.197 |

TABLE VI
COMPARISON OF COMPUTATIONAL OVERHEAD

| Scheme | SIPG | CIMS | IDMV | Total |
|---|---|---|---|---|
| Bagga *et al.*'s scheme [53] | $4T_{pm}^{ecc} + 2T_{mtp} + 1T_{ha} = 12.867\ ms$ | $3T_{pm}^{ecc} + 3T_{pa}^{ecc} + 3T_{ha} = 1.86\ ms$ | $3T_{bp} + 4T_{pm}^{ecc} + 4T_{pa}^{ecc} + 3T_{ha} = 6.847\ ms$ | $21.574\ ms$ |
| Shen *et al.*'s scheme [54] | $3T_{bp} + 4T_{pm}^{bp} + 1T_{pa}^{bp} + 2T_{ep} + 4T_{ha} = 9.505\ ms$ | $1T_{bp} + 1T_{pm}^{bp} + 1T_{ep} + 1T_{ha} = 2.745\ ms$ | $2T_{bp} + 2T_{pm}^{bp} + 1T_{pa}^{bp} + 1T_{ep} + 1T_{ha} = 5.481\ ms$ | $17.731\ ms$ |
| Yang *et al.*'s scheme [55] | $17T_{pm}^{ecc} + 5T_{pa}^{ecc} + 11T_{ha} = 10.522\ ms$ | $1T_{ha} = 0.001\ ms$ | $3T_{pm}^{ecc} + 4T_{pa}^{ecc} + 2T_{ha} = 1.86\ ms$ | $12.383\ ms$ |
| Wang *et al.*'s scheme [56] | $4T_{pm}^{ecc} + 1T_{pa}^{ecc} + 2T_{ha} = 2.475\ ms$ | $9T_{pm}^{ecc} + 4T_{pa}^{ecc} + 5T_{ha} = 5.571\ ms$ | $3T_{pm}^{ecc} + 2T_{pa}^{ecc} + 4T_{ha} = 1.86\ ms$ | $9.906\ ms$ |
| Our proposed scheme | $7T_{pm}^{ecc} + 1T_{pa}^{ecc} + 4T_{ha} = 4.331\ ms$ | $3T_{pm}^{ecc} + 1T_{pa}^{ecc} + 2T_{ha} = 1.857\ ms$ | $4T_{pm}^{ecc} + 2T_{pa}^{ecc} + 2T_{ha} = 2.476\ ms$ | $8.664\ ms$ |

related schemes.

Next, considering the diversity of computational devices, we test the portability of the proposed pseudonym scheme. We consider five different devices, including a workstation with an Intel i9-12900k@3.9GHz and 64GB RAM (Intel i9), a computer with an i7-12700@2.1GHz and 32GB RAM (Intel i7), a computer with an i5-8500@3GHz and 8GB RAM (Intel i5), a smartphone with Snapdragon 7+ Gen 2 and 16GB RAM (Snapdragon 7+ Gen 2), and a MacBook with an M1 chip and 16GB RAM (Apple M1). We test the time consumption for each of the five phases of the pseudonym authentication process on these devices: System Initialization (SI), Computing Devices Registration (CR), Pseudonym Generation (PG), Certificate Issuance (CI), and Identity Verification (IV), including the Total Time (TT) for the entire process, as shown in Fig. 9. Time consumption varies across devices due to differences in CPU performance. Specifically, the MacBook requires the longest total time to complete a pseudonym authentication, only $20.813\ ms$. The SI is the most time-consuming, as it involves generating system keys and registering user information. However, typically, SI is only performed once, whereas CR, PG, CI, and IV take less than $8\ ms$ across all devices, with PG taking the longest at $7.628\ ms$ on the smartphone. This demonstrates the good portability and lightweight of the proposed pseudonym scheme.

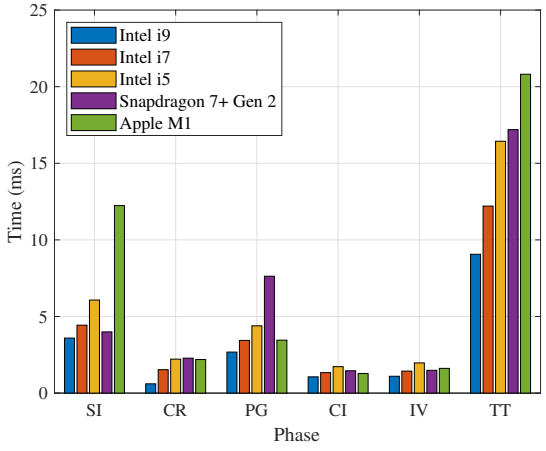


Fig. 9. The time consumption on different devices.

## VIII. CONCLUSION

This article investigates improving computing resource utilization within 6G CPN and proposes BECS, a privacy-preserving computing resource sharing mechanism. We consider various communication, computing, and service factors in 6G CPN, modeling them as a six-objective MOOP. Meanwhile, we utilize the proposed NSGA-III-KDR to find optimal solutions for this MOOP. Additionally, we introduce a novel pseudonym scheme to protect the privacy of users engaged in computing resource sharing. Extensive simulations demonstrate the effectiveness of BECS. Moving forward, we intend to further explore computing sharing in dynamic real-time scenarios and investigate advanced access control solutions to protect the task payload.

## REFERENCES

[1] P. Zhang, H. Yang, Z. Feng *et al.*, "Toward intelligent and efficient 6g networks: Jcsc enabled on-purpose machine communications," *IEEE Wireless Commun.*, vol. 30, no. 1, pp. 150–157, 2023.
[2] Y. Lu, B. Ai, Z. Zhong *et al.*, "Energy-efficient task transfer in wireless computing power networks," *IEEE Internet of Things J.*, vol. 10, no. 11, pp. 9353–9365, 2023.
[3] Q.-V. Pham, F. Fang, V. N. Ha *et al.*, "A survey of multi-access edge computing in 5g and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, 2020.
[4] Q. Duan, J. Huang, S. Hu *et al.*, "Combining federated learning and edge computing toward ubiquitous intelligence in 6g network: Challenges, recent advances, and future directions," *IEEE Commun. Surv. Tutorials*, vol. 25, no. 4, pp. 2892–2950, 2023.
[5] Q. Chen, C. Yang, S. Lan *et al.*, "Two-stage evolutionary search for efficient task offloading in edge computing power networks," *IEEE Internet of Things J.*, vol. 11, no. 19, pp. 30 787–30 799, 2024.
[6] A. Clemm, M. F. Zhani, and R. Boutaba, "Network management 2030: Operations and control of network 2030 services," *J. Network Syst. Manage.*, vol. 28, no. 4, pp. 721–750, 2020.
[7] Y. Liu, X. Xing, Z. Tong *et al.*, "Secure and scalable cross-domain data sharing in zero-trust cloud-edge-end environment based on sharding blockchain," *IEEE Trans. Dependable Secure Comput.*, vol. 21, no. 4, pp. 2603–2618, 2024.
[8] T. Hewa, G. Gür, A. Kalla *et al.*, "The role of blockchain in 6g: Challenges, opportunities and research directions," in *Proc. 2nd 6G Wireless Summit (6G SUMMIT)*, 2020, pp. 1–5.
[9] W. Yang, H. Wang, Z. Li *et al.*, "Privacy-preserving machine learning in cloud–edge–end collaborative environments," *IEEE Internet Things J.*, vol. 12, no. 1, pp. 419–434, 2025.
[10] T. Taleb, K. Samdanis, B. Mada *et al.*, "On multi-access edge computing: A survey of the emerging 5g network edge cloud architecture and orchestration," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.
[11] P. Wang, W. Sun, H. Zhang *et al.*, "Distributed and secure federated learning for wireless computing power networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 7, pp. 9381–9393, 2023.

[12] Q. Chen, C. Yang, S. Lan *et al.*, "Two-stage evolutionary search for efficient task offloading in edge computing power networks," *IEEE Internet Things J.*, vol. 11, no. 19, pp. 30 787–30 799, 2024.

[13] S. Yukun, L. Bo, L. Junlin *et al.*, "Computing power network: A survey," *China Commun.*, vol. 21, no. 9, pp. 109–145, 2024.

[14] S. Hu, Y.-C. Liang, Z. Xiong *et al.*, "Blockchain and artificial intelligence for dynamic resource sharing in 6g and beyond," *IEEE Wireless Commun.*, vol. 28, no. 4, pp. 145–151, 2021.

[15] K. Peng, H. Huang, B. Zhao *et al.*, "Intelligent computation offloading and resource allocation in iiot with end-edge-cloud computing using nsga-iii," *IEEE Trans. Network Sci. Eng.*, vol. 10, no. 5, pp. 3032–3046, 2023.

[16] Y. Gong, K. Bian, F. Hao *et al.*, "Dependent tasks offloading in mobile edge computing: A multi-objective evolutionary optimization strategy," *Future Gener. Comput. Syst.*, vol. 148, pp. 314–325, 2023.

[17] Y. Tian, R. Cheng, X. Zhang *et al.*, "A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 331–345, 2019.

[18] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Secure computation offloading in blockchain based iot networks with deep reinforcement learning," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 4, pp. 3192–3208, 2021.

[19] D. C. Nguyen, M. Ding, P. N. Pathirana *et al.*, "Cooperative task offloading and block mining in blockchain-based edge computing with multi-agent deep reinforcement learning," *IEEE Trans. Mob. Comput.*, vol. 22, no. 4, pp. 2021–2037, 2023.

[20] D. Wang, N. Zhao, B. Song *et al.*, "Resource management for secure computation offloading in softwarized cyber–physical systems," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 9294–9304, 2021.

[21] A. Samy, I. A. Elgendy, H. Yu *et al.*, "Secure task offloading in blockchain-enabled mobile edge computing with deep reinforcement learning," *IEEE Trans. Netw. Serv. Manage.*, vol. 19, no. 4, pp. 4872–4887, 2022.

[22] J. Wang, X. Ling, Y. Le *et al.*, "Blockchain-enabled wireless communications: a new paradigm towards 6g," *Natl. Sci. Rev.*, vol. 8, no. 9, p. nwab069, 04 2021.

[23] J. Xie, K. Zhang, Y. Lu *et al.*, "Resource-efficient dag blockchain with sharding for 6g networks," *IEEE Network*, vol. 36, no. 1, pp. 189–196, 2022.

[24] P. Wang, W. Sun, H. Zhang *et al.*, "Distributed and secure federated learning for wireless computing power networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 7, pp. 9381–9393, 2023.

[25] Z. Yang, K. Liu, Y. Chen *et al.*, "Two-level stackelberg game for iot computational resource trading mechanism: A smart contract approach," *IEEE Trans. Serv. Comput.*, vol. 15, no. 4, pp. 1883–1895, 2022.

[26] V.-L. Nguyen, P.-C. Lin, B.-C. Cheng *et al.*, "Security and privacy for 6g: A survey on prospective technologies and challenges," *IEEE Commun. Surv. Tutorials*, vol. 23, no. 4, pp. 2384–2428, 2021.

[27] K. Yan, W. Ma, and S. Sun, "Communications and networks resources sharing in 6g: Challenges, architecture, and opportunities," *IEEE Wireless Commun.*, vol. 31, no. 6, pp. 102–109, 2024.

[28] A. Hazra, A. Kalita, and M. Gurusamy, "Meeting the requirements of internet of things: The promise of edge computing," *IEEE Internet Things J.*, vol. 11, no. 5, pp. 7474–7498, 2024.

[29] Y. He, Z. Zhang, F. R. Yu *et al.*, "Deep-reinforcement-learning-based optimization for cache-enabled opportunistic interference alignment wireless networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10 433–10 445, 2017.

[30] L. Yang, J. Cao, Y. Yuan *et al.*, "A framework for partitioning and execution of data stream applications in mobile cloud computing," *SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 4, pp. 23–32, 2013.

[31] Y. Liu, S. Zhang, X. Mu *et al.*, "Evolution of noma toward next generation multiple access (ngma) for 6g," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 4, pp. 1037–1071, 2022.

[32] B. Liu, C. Liu, and M. Peng, "Resource allocation for energy-efficient mec in noma-enabled massive iot networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 1015–1027, 2021.

[33] D. Gross, *Fundamentals of Queueing Theory*. Hoboken, NJ, USA: Wiley, 2008.

[34] Y. Wen, W. Zhang, and H. Luo, "Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2012, pp. 2716–2720.

[35] X. Xu, X. Liu, X. Yin *et al.*, "Privacy-aware offloading for training tasks of generative adversarial network in edge computing," *Inf. Sci.*, vol. 532, pp. 1–15, 2020.

[36] Z. Cui, Z. Xue, T. Fan *et al.*, "A many-objective evolutionary algorithm based on constraints for collaborative computation offloading," *Swarm Evol. Comput.*, vol. 77, p. 101244, 2023.

[37] Z. Liao, J. Peng, J. Huang *et al.*, "Distributed probabilistic offloading in edge computing for 6g-enabled massive internet of things," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5298–5308, 2021.

[38] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, 2014.

[39] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *Proc. IEEE Congr. Evol. Comput.*, 2008, pp. 2419–2426.

[40] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *Proc. Int. Conf. Database Theory*, 2001, pp. 420–434.

[41] D. Francois, V. Wertz, and M. Verleysen, "About the locality of kernels in high-dimensional spaces," in *Proc. Int. Symp. Appl. Stochastic Models Data Anal. (ASMDA)*, 2005, pp. 238–245.

[42] K. Yan, P. Zeng, K. Wang *et al.*, "Reputation consensus-based scheme for information sharing in internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 72, no. 10, pp. 13 631–13 636, 2023.

[43] C.-P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, pp. 161–174, 1991.

[44] Y. Li, Q. Zhang, H. Yao *et al.*, "Multimodal reinforcement learning aided dynamic service function chain deployment in satellite-terrestrial network," in *Proc. 21th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2025, pp. 379–385.

[45] M. Jia, J. Wu, Q. Guo *et al.*, "Service-oriented sagin with pervasive intelligence for resource-constrained users," *IEEE Network*, vol. 38, no. 2, pp. 79–86, 2024.

[46] J. Liu, K. Xiong, D. W. K. Ng *et al.*, "Max-min energy balance in wireless-powered hierarchical fog-cloud computing networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7064–7080, 2020.

[47] Y. Tian, R. Cheng, X. Zhang *et al.*, "Platemo: A matlab platform for evolutionary multi-objective optimization [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, 2017.

[48] R. Cheng, M. Li, Y. Tian *et al.*, "A benchmark test suite for evolutionary many-objective optimization," *Complex Intell. Syst.*, vol. 3, pp. 67–81, 2017.

[49] Y. Tian, X. Zhang, C. Wang *et al.*, "An evolutionary algorithm for large-scale sparse multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 380–393, 2020.

[50] A. Zhou, Y. Jin, Q. Zhang *et al.*, "Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion," in *Proc. IEEE Congr. on Evol. Comput.*, 2006, pp. 892–899.

[51] H. Wang, Y. Jin, and X. Yao, "Diversity assessment in many-objective optimization," *IEEE Trans. Cybern.*, vol. 47, no. 6, pp. 1510–1522, 2017.

[52] P. S. Barreto and M. Naehrig, "Pairing-friendly elliptic curves of prime order," in *Proc. 12th Int. Workshop Select. Areas Cryptography (SAC)*. Springer, 2005, pp. 319–331.

[53] P. Bagga, A. K. Sutrala, A. K. Das *et al.*, "Blockchain-based batch authentication protocol for internet of vehicles," *J. Syst. Archit.*, vol. 113, p. 101877, 2021.

[54] M. Shen, H. Liu, L. Zhu *et al.*, "Blockchain-assisted secure device authentication for cross-domain industrial iot," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 942–954, 2020.

[55] Y. Yang, L. Wei, J. Wu *et al.*, "A blockchain-based multidomain authentication scheme for conditional privacy preserving in vehicular ad-hoc network," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8078–8090, 2022.

[56] F. Wang, J. Cui, Q. Zhang *et al.*, "Blockchain-based lightweight message authentication for edge-assisted cross-domain industrial internet of things," *IEEE Trans. Dependable Secure Comput.*, vol. 21, no. 4, pp. 1587–1604, 2024.

**Kun Yan** (Member, IEEE) received a B.S. degree in internet of things engineering from Xi'an University of technology, Xi'an, China, in 2019, He is currently working toward a Ph.D. degree in the School of Telecommunications Engineering, Xidian University. His research interests include 6G networks, internet of things, and blockchain.

**Wenping Ma** (Member, IEEE) received the B.S. and M.S. degrees in fundamental mathematics from Shaanxi Normal University, Xi'an, China, in 1987 and 1990, respectively, and the Ph.D. degree in communication and information system from Xidian University, Xi'an, in 1999, where he is currently a Full Professor with the School of Telecommunications Engineering. His current research interests include information theory, communication theory, blockchain, and 6G networks security.

**Shaohui Sun** (Member, IEEE) received his Ph.D. from Xidian University, Xi'an, China, in 2003. From March 2003 to June 2006, he was a postdoctoral research fellow at the Datang Telecom technology and industry group, Beijing, China. From June 2006 to December 2010, he worked at the Datang Mobile Communications Equipment Co., Ltd., Beijing, where he has been deeply involved in the development and standardization of the Third-Generation Partnership Project Long-Term Evolution (3GPP LTE). Since January 2011, he has been the Chief Technical Officer with Datang Wireless Mobile Innovation Center of the Datang Telecom Technology and Industry group. His current research interest includes advanced technologies related to B5G/6G.