Edge-Assisted Multi-Layer Offloading Optimization of LEO Satellite-Terrestrial Integrated Networks

Xuelin Cao, Bo Yang, Yulong Shen, Chau Yuen, *Fellow, IEEE*, Yan Zhang, *Fellow, IEEE*, Zhu Han, *Fellow, IEEE*, H. Vincent Poor, *Life Fellow, IEEE* and Lajos Hanzo, *Life Fellow, IEEE*

***Abstract*—Sixth-Generation (6G) technologies will revolutionize the wireless ecosystem by enabling the delivery of futuristic services through satellite-terrestrial integrated networks (STINs). As the number of subscribers connected to STINs increases, it becomes necessary to investigate whether the edge computing paradigm may be applied to low Earth orbit satellite (LEOS) networks for supporting computation-intensive and delay-sensitive services for anyone, anywhere, and at any time. Inspired by this research dilemma, we investigate a LEOS edge-assisted multi-layer multi-access edge computing (MEC) system. In this system, the MEC philosophy will be extended to LEOS, for defining the LEOS edge, in order to enhance the coverage of the multi-layer MEC system and address the users' computing problems both in congested and isolated areas. We then design its operating offloading framework and explore its feasible implementation methodologies. In this context, we formulate a joint optimization problem for the associated communication and computation resource allocation for minimizing the overall energy dissipation of our LEOS edge-assisted multi-layer MEC system while maintaining a low computing latency. To solve the optimization problem effectively, we adopt the classic alternating optimization (AO) method for decomposing the original problem and then solve each sub-problem using low-complexity iterative algorithms. Finally, our numerical results show that the offloading scheme conceived achieves low computing latency and energy dissipation compared to the state-of-the-art solutions, a single layer MEC supported by LEOS or base stations (BS).**

***Index Terms*—Satellite-terrestrial integrated network, LEO satellite, multi-access edge computing, 6G**

This work is partially supported by the Huashan Talent Fund. The work of B. Yang is partially supported by the National Natural Science Fund for Excellent Young Scientists Fund Program (Overseas). The work of Y. Shen is partially supported by the National Key Research and Development Program of China 2018YFE0207600, the National Natural Science Foundation of China 61972308, Natural Science Basic Research Program of Shaanxi 2019JC-17. The work of C. Yuen is partially supported by the Ministry of Education, Singapore, under its MOE Tier 2 (Award number MOE-T2EP50220-0019). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Ministry of Education, Singapore. The work of Z. Han is partially supported by U.S. National Science Foundation under Grant CNS-2107216, CNS-2128368 and CMMI-2222810. The work of H. V. Poor is partially supported by U.S. National Science Foundation under Grant CNS-2128448. The work of L. Hanzo is partially supported by the Engineering and Physical Sciences Research Council projects EP/W016605/1 and EP/P003990/1 (COALESCE) as well as of the European Research Council's Advanced Fellow Grant QuantCom (Grant No. 789028). (Corresponding author: Xuelin Cao.)

X. Cao is with the School of Cyber Engineering, Xidian University, Xi'an 710071, China, and also with the Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore 487372. (e-mail: caoxuelin@xidian.edu.cn); B. Yang is with the School of Computer Science, Northwestern Polytechnical University, Xi'an, China. (email: yang_bo@nwpu.edu.cn); Y. Shen is with the School of Computer Science and Technology, Xidian University, Xi'an, China. (e-mail: ylshen@mail.xidian.edu.cn); C. Yuen is with the Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore 487372. (e-mail: yuenchau@sutd.edu.sg); Y. Zhang is with the Department of Informatics, University of Oslo, Norway. (e-mail: yanzhang@ieee.org); Z. Han is with the Department of Electrical and Computer Engineering in the University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul, South Korea, 446-701. (e-mail: zhan2@uh.edu); H. V. Poor is with the Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ, 08544, USA. (e-mail: poor@princeton.edu); L. Hanzo is with the School of Electronics and Computer Science, University of Southamptonc, Southamptonc, SO17 1BJ, UK. (e-mail: lh@ecs.soton.ac.uk).

I. INTRODUCTION

THe new space race is heating up as private Internet and aerospace companies such as SpaceX, Amazon and Telesat are building large satellite constellations in low Earth orbit (LEO) to provide global broadband Internet access. For instance, SpaceX's Starlink constellation has already entered a public beta phase for subscribers in North America and the United Kingdom with more than 1,500 satellites in use, and more than 12,000 LEO satellites (LEOS) will be deployed by 2027 [1]–[3]. They are expected to provide broadband services for billions of users worldwide, even in hither to unreachable areas at low latency (i.e., below 30 ms), high data rate (i.e., above 100 Mb/s) and wide coverage. Hence, satellite-terrestrial integrated networks (STINs) are expected to become predominant over the next couple of years [4]–[9]. Beyond enabling Internet access for billions of users, STINs are also capable of connecting a large number of edge devices. Thus, a novel aspect of STINs is the provision of computational resources over satellites which constitutes a new paradigm, i.e., treating LEOSs as the "new edge", namely the LEOS edge [1], [2], [10]–[12].

Recently, interest in LEOS edge research has increased tremendously, and commercial solutions have also appeared. For instance, Loft Orbital is developing a processor for providing onboard edge computing capabilities for military satellites in orbit, while the company called Ibeos provides bus and payload computing solutions that are suitable for all types of space environments [13]. Compared to the traditional LEOS-assisted multi-access edge computing (MEC) system, the LEOS edge will bring about compelling benefits, since not all computational tasks will have to be forwarded to the data center. However, offloading data from all users to the LEOS edge for processing will put substantial strain on satellite networks, since next-generation applications will have to process a large amount of data or require low latency and excellent energy efficiency [14]–[16], as exemplified by flawless holographic services projecting holographic subjects to a remote place. A similar potential bottle-neck also exists in terrestrial networks, where the geographically distributed ground stations harnessed as computing modes may become incapable of handling the requests because the computing resources at the edge of terrestrial networks are also limited.

A. Motivation

The seamless integration of ground stations and the LEOS edge under the next-generation wireless ecosystem is expected to mitigate the aforementioned impediments and improve service availability, continuity, ubiquity and scalability. These novel concepts have opened up many new frontiers for network operators and service providers in terms of providing versatile uninterrupted services in diverse application scenarios. In this paper, the multi-layer MEC system concept plays a significant role in STINs by providing low computing latency, high energy efficiency and extensive coverage. However, integrating BS servers and the LEOS edge in STINs for providing high-speed computing services while maintaining high energy efficiency poses the following challenges.

- *Framework:* How to design a multi-layer MEC framework for STINs with the assistance of the LEOS edge?
- *Implementation:* How to conceive efficient network access, resource management and computation offloading for the framework designed?
- *Optimization:* How to minimize the overall energy dissipation of the LEOS edge-assisted multi-layer MEC system while maintaining a low computing latency?

Inspired by these potential benefits and the resultant challenges, in this paper, we propose the LEOS edge-assisted multi-layer MEC system for STINs, a concept highlighted in Fig. 1, where the LEOS edge is expected to complement the terrestrial networks in providing high-efficiency computing services for ground users at a low energy.

B. State-of-the-Art

In the STIN-aided six-generation (6G) wireless ecosystem of Fig. 1, computing and storage resources are embedded either in the terrestrial or satellite networks, as close to mobile devices as possible to offer low computing latency, high-throughput, high-privacy and low-cost services. Employing both local computing and edge computing in support of complex tasks, MEC is becoming a promising paradigm and has attracted a lot of attention.

1) MEC-Enabled Terrestrial Networks: The demand for processing data close to its source led to the increased popularity of the edge computing paradigm. Embedding computing resources into the edge of terrestrial networks reduces both the computing latency and energy dissipation while guaranteeing privacy and security. In this context, most of the works focused their attention on the offloading design and optimization of terrestrial networks. Explicitly, Yang *et al.* [18] studied the tradeoff between the performance gains and energy dissipation in collaborative offloading, while Wang *et al.* [19] conceived a Non-Orthogonal Multiple Access (NOMA)-based fog computing framework for reducing the computing latency and energy dissipation. To achieve high energy efficiency, Dai *et al.* [20] designed a two-tier MEC framework by jointly considering the computation offloading and user association. As a further development, Wang *et al.* [21] studied the energy-efficient task offloading design of a massive multiple-input and multiple-output (MIMO)-aided multi-pair fog computing system. Relying on a breakthrough in the fabrication of programmable meta-materials, reconfigurable intelligent surfaces (RIS) [22]–[27] improve the offloading links of MEC systems. For example, Bai *et al.* [28] proposed a RIS-assisted MEC solution for reducing the offloading latency, while the security of RIS-assisted MEC systems was studied in [29]. In a 6G context, artificial intelligence (AI) aided MEC systems were explored in [30]–[36]. Specifically, Yang *et al.* [30]–[32] adopted deep learning techniques instead of conventional optimization methods for constructing beneficial offloading policies. Furthermore, edge intelligence in MEC systems was explored for finding more optimal solutions [33]–[36].

2) MEC-Enabled Satellite Networks: Traditionally, LEOSs are used in a “bent pipe” architecture, where every satellite simply acts as a communication link and pushes off the processing to the terrestrial datacenters [37]. In other words, most of the satellites in space simply serve a single mission. These satellites typically carry out weather forecasting, disaster management, or region monitoring. For example, when collecting remote sensing data over Greenland (Ulloriaq) [38], the satellite is idle for most of the duration of its orbit. With the development of commercialized LEOSs and the availability of a larger number of distributed ground stations, the authors of [2], [11], [39]–[43] advocated space-assisted offloading. For example, Di *et al.* [39], [40] proposed a terrestrial-satellite network architecture for efficient data offloading, where the LEOS has to forward the ground user’s data either to a terrestrial gateway station or to a macro base station (BS) for processing. To support the computational requirements of remote areas, Cheng *et al.* [41] proposed a space-air-ground integrated networking aided edge/cloud computing architecture, where UAVs provide near-user edge computing services and satellites provide access to powerful cloud computing. Similarly, Li *et al.* [42] studied a cache-enabled LEOS, where LEOSs are connected to macro BSs through wireless links for updating their cached contents. The proliferation of satellite constellations calls for a new LEOS edge capable of supporting edge computing from space. Hence, the research community has turned to the exploration of MEC over LEOSs. For example, Bhosale *et al.* [11] proposed a service orchestration technique for the LEOS edge, while Pfandzelter *et al.* [2] discussed the unique characteristics of the LEOS edge. As a further advance, Tang *et al.* [43] proposed multi-tier computing intrinsically combined with local computing, offloading tasks both to LEOS and to the clouds, and Cao *et al.* [44] investigated the interplay of RIS and multi-layer MEC in the context of space information networks.

C. Open Problems and Contributions

At the time of writing, the potential of the LEOS edge-assisted multi-layer MEC paradigm has not been fully exploited, predominantly because the computation offloading link of the LEOS edge is far from perfect. Even though LEOS-enabled MEC constitutes a promising solution for supporting

BSs by providing high-capacity backhaul, wide-ranging coverage, and low latency computing services, there is very limited literature on its energy efficiently, especially for a multi-layer MEC system involving LEOS edge. By contrast, this paper aims to jointly investigate the computing and communication performance of STINs. Against this background, our main contributions are summarized as follows.

- *Framework design:* We conceive a universal framework for multi-layer MEC systems by exploiting the emergence of the LEOS edge in support of low computing latency, high energy efficiency, and seamless coverage in multi-layer MEC systems.
- *Implementation methodologies:* We conceive the associated network access, resource management, and computation offloading for the proposed framework, thereby enhancing both the communication and computing performance of a practical system at a low cost.
- *Optimization algorithms:* We design optimization algorithms for minimizing the energy dissipation of the LEOS edge-assisted multi-layer MEC system advocated.
- *Performance evaluation:* We characterize the proposed LEOS edge-assisted multi-layer MEC STIN in terms of its latency and energy efficiency. Our simulation results show that all of these three aspects can be significantly improved compared to the state-of-the-art benchmarks.

The rest of this paper is organized as follows. In Section II, we describe a LEOS edge-assisted multi-layer MEC system for STINs. We then design an offloading scheme for the proposed multi-cell system and formulate a joint optimization problem that includes the offloading mode, offloading volume, and the computing resource allocation of BS servers, plus the LEOS edge for minimizing the system's overall energy dissipation in Section III. Next, in Section IV, we explore an energy-efficient offloading solution for the formulated mixed-integer nonlinear programming (MINLP) problem. We discuss the numerical results in Section V, and we finally conclude in Section VI.

Notations: As per the traditional notation, a bold letter indicates a vector or matrix. $\max\{\cdot\}$ and $\min\{\cdot\}$ represent the maximum value and the minimum value, respectively. The amplitude of a complex number x is denoted by $|x|$.

II. System Model

In this section, we first introduce a LEOS edge-assisted computation scenario in which the terrestrial users offload their tasks to the BS server or the LEOS edge. Then, the communication, computing, and energy dissipation models are discussed.

A. Scenario Model

We consider an integrated satellite-terrestrial network deploying the LEOS edge as shown in Fig. 1. In the scenario considered, the BS server cannot carry out the numerous computation tasks of all the ground users within the scope of the cell it covers due to its limited computational capability. In this case, the LEOS edge has to coordinate with each BS server to assist the task processing of ground users, such as holographic video surveillance. Therefore, each ground user

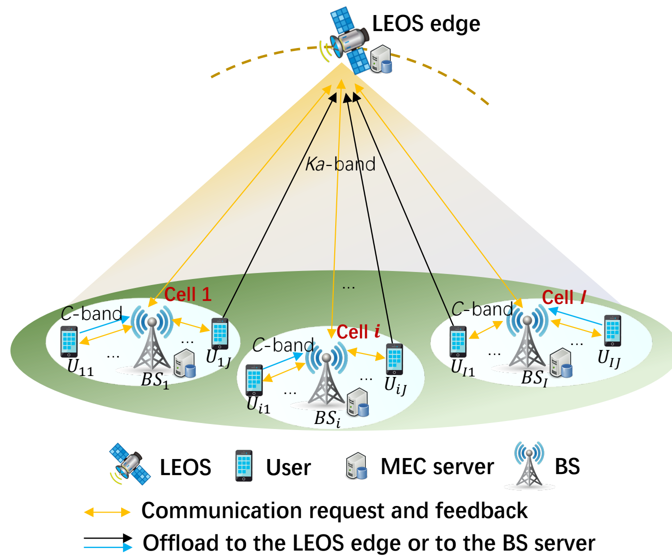


Fig. 1: A LEOS edge-assisted multi-layer MEC system for STINs

can offload its tasks in two modes: 1) offloading its tasks to the BS server via wireless backhaul links over the C-band, or 2) offloading its tasks to the LEOS edge via wireless backhaul links over the Ka-band. After computing at the LEOS edge or the BS server, the computing results are fed back to the users via the Ka-band or the C-band.

In our system, we consider a single LEOS and I BSs, each serving J ground users, and the LEOS edge can serve $I \times J$ ground users. Additionally, the LEOS and each BS are equipped with a computational server for carrying out computational tasks. The set of BS servers and the ground users in each BS are denoted by $\mathcal{I} = \{1, \ldots, i, \ldots, I\}$ and $\mathcal{J} = \{1, \ldots, j, \ldots, J\}$, respectively. We denote the ith BS server and its jth ground user as BS_i and U_{ij}, respectively, where $i \in \mathcal{I}, j \in \mathcal{J}$. Let L_{ij} and l_{ij} represent the total number of bits and the number of bits to be offloaded, respectively. It should be noted that the task of U_{ij} is offloaded to the BS server or the LEOS edge when $l_{ij} \neq 0$, otherwise, it is processed locally. In addition, let u_{ij} indicate the offloading mode, i.e.,

$$u_{ij} = \begin{cases} 1, & \text{U}_{ij} \text{ offloads to } \text{BS}_i, \ \forall i, j \in \mathcal{I}, \mathcal{J}, \\ 0, & \text{U}_{ij} \text{ offloads to the LEOS edge}, \forall i, j \in \mathcal{I}, \mathcal{J}. \end{cases} \quad (1)$$

We assume that each ground user is equipped with a single antenna, each BS and the LEOS are equipped with multiple antennas and the popular orthogonal frequency-division multiple access (OFDMA) scheme for accessing either the LEOS or the BS cells. In particular, total I BS cells share the same frequency resource over the C-band spectrum, which can be evenly split and allocated to the ground users that have to offload their tasks to the BS server. Similarly, the frequency resource over the Ka-band spectrum is also evenly split and allocated to the ground users that have to offload their tasks to the LEOS edge. Since we adopted the OFDMA scheme, there is no intra-cell interference in each BS cell. We also assume that there is no inter-cell interference among BS cells because the coverage of BSs is non-overlapping. Note that the proposed system model can also be extended to a larger system that consists of more LEOSs, where association between BSs and LEOs has to be carefully considered.

B. Communication Model

1) Terrestrial Communications: According to the model considered, the signal received by BS$_i$ over the C-band is given by

$$y_{ij}^{BS_i} = \sqrt{p_{ij}}h_{ij}^{C}s_{ij} + N_i, \quad \exists u_{ij} = 1, \forall i, j \in \mathcal{I}, \mathcal{J}, \qquad (2)$$

where p_{ij} is the transmit power of U$_{ij}$, s_{ij} is the signal of U$_{ij}$ with unit energy, $N_i \sim (0, \sigma^2)$ is an additive white Gaussian noise (AWGN) at BS$_i$, and σ^2 is the noise variance. Furthermore, h_{ij}^{C} is the channel between U$_{ij}$ and BS$_i$, which is dominated by the Line-of-Sight (LoS) path. Then, we have

$$h_{ij}^{C} = \xi d_{ij}^{-\alpha}, \qquad (3)$$

where d_{ij} denotes the distance between U$_{ij}$ and BS$_i$, while ξ corresponds to the unity channel gain at the reference distance of $d_{ij} = 1m$ and α is the path loss exponent.

The achievable capacity of U$_{ij}$ served by BS$_i$ over the C-band is given by

$$R_{ij}^{BS_i} = \frac{B_C}{J}\log_2\left(1 + \frac{p_{ij}|h_{ij}^{C}|^2}{\sigma^2}\right), \quad \exists u_{ij} = 1, \forall i, j \in \mathcal{I}, \mathcal{J}, \qquad (4)$$

where B_C is the total bandwidth in the C-band.

Therefore, the total capacity at BS$_i$ is formulated by

$$R_{total}^{BS_i} = \sum_{j=1}^{J}\frac{u_{ij}B_C}{J}\log_2\left(1 + \frac{p_{ij}|h_{ij}^{C}|^2}{\sigma^2}\right). \qquad (5)$$

2) LEOS Communications: For the LEOS communications, the altitude, speed and position information of the LEOS are known to all BSs in a time slot due to the orbital pre-planning. For the sake of simplicity, a quasi-static fading channel model is considered in a time slot. Thus, the signal received by the LEOS over the Ka-band is given by

$$y_{ij}^{LEO} = \sqrt{p_{ij}}h_{ij}^{Ka}s_{ij} + N_0, \quad \exists u_{ij} = 0, \forall i, j \in \mathcal{I}, \mathcal{J}, \qquad (6)$$

where $N_0 \sim (0, \hat{\sigma}^2)$ is the additive white Gaussian noise (AWGN) at the LEOS, and $\hat{\sigma}^2$ is the noise variance. Furthermore, h_{ij}^{Ka} is the channel between U$_{ij}$ and the LEOS, which is given by

$$h_{ij}^{Ka} = v_{ij}\gamma_{ij}\hat{d}_{ij}^{-\beta}, \qquad (7)$$

where $v_{ij} \sim (0, 1)$ is a complex Gaussian variable representing Rayleigh fading, γ_{ij} follows log-normal distributed shadow fading, $\hat{d}_{ij}$ is the distance between U$_{ij}$ and the LEOS, and β is the path loss exponent.

The achievable capacity of U$_{ij}$ served by the LEOS over the Ka-band is denoted by

$$R_{ij}^{LEO} = \frac{B_{Ka}}{IJ}\log_2\left(1 + \frac{p_{ij}|h_{ij}^{Ka}|^2}{\hat{\sigma}^2}\right), \exists u_{ij} = 0, \forall i, j \in \mathcal{I}, \mathcal{J}, \qquad (8)$$

where B_{Ka} is the total bandwidth in Ka-band.

Therefore, the total capacity at the LEOS is given by

$$\begin{aligned} R_{total}^{LEO} &= \sum_{i=1}^{I}\sum_{j=1}^{J}R_{ij}^{LEO} \qquad (9) \\ &= \sum_{i=1}^{I}\sum_{j=1}^{J}(1 - u_{ij})\frac{B_{Ka}}{IJ}\log_2\left(1 + \frac{p_{ij}|h_{ij}^{Ka}|^2}{\hat{\sigma}^2}\right). \end{aligned}$$

C. Computing Model

The multi-layer MEC system integrates local computing, BS server-aided computing (i.e., offloading tasks from the users to the BS servers), and LEOS edge-aided computing (i.e., offloading tasks from the users to the LEO edges). Without loss of generality, we consider a partial offloading, where a fraction of tasks are processed locally, and the remaining part will be offloaded to the BS server or the LEOS edge.

1) Local Computing: For considering the case that the task of U$_{ij}$ is partially processed locally, c_{ij} represents the number of CPU cycles required for processing a single bit at U$_{ij}$. Furthermore, the computational capability of U$_{ij}$ is f_{ij}^{l}, which is quantified by the number of CPU cycles per second. Note that the task of U$_{ij}$ is fully offloaded to the BS server or to the LEOS edge when $l_{ij} = L_{ij}$. In contrast, the task is fully processed locally when $l_{ij} = 0$. Therefore, the time required for carrying out the local computing, T_{ij}^{l}, is given by

$$T_{ij}^{l} = \begin{cases} \frac{(L_{ij} - l_{ij})c_{ij}}{f_{ij}^{l}}, & l_{ij} \neq L_{ij}, \\ 0, & \text{Otherwise}. \end{cases} \qquad (10)$$

2) BS Sever Computing: When the task of U$_{ij}$ is offloaded to BS$_i$, let $f_{max}^{BS_i}$ and $f_{ij}^{BS_i}$ denote the maximum number of executable CPU cycles at the BS$_i$ server and the computational capability of the BS$_i$ server allocated to U$_{ij}$, respectively, where $\sum_{j=1}^{J} f_{ij}^{BS_i} \leq f_{max}^{BS_i}, i \in \mathcal{I}$. Here, we assume that the computation offloading of U$_{ij}$ starts when l_{ij} bits of U$_{ij}$ are completely uploaded to BS$_i$. In this case, the processing latency at the BS$_i$ sever contains the computational and communication delays, where the feedback latency is negligible since the size of computational results is relatively small. Therefore, the total latency required for carrying out the BS$_i$ server's computing, $T_{ij}^{BS_i}$, is given by

$$T_{ij}^{BS_i} = \begin{cases} \frac{l_{ij}}{R_{ij}^{BS_i}} + \frac{l_{ij}c_{ij}}{f_{ij}^{BS_i}}, & u_{ij} = 1, \\ 0, & \text{Otherwise}. \end{cases} \qquad (11)$$

3) LEOS Edge Computing: When the task of U$_{ij}$ is offloaded to the LEOS edge, let f_{max}^{LEO} and f_{ij}^{LEO} denote the maximum number of executable CPU cycles at the LEOS edge and the computational capability of the LEOS edge allocated to U$_{ij}$, respectively, where $\sum_{i=1}^{I}\sum_{j=1}^{J} f_{ij}^{LEO} \leq f_{max}^{LEO}, i \in \mathcal{I}, j \in \mathcal{J}$. Here, we assume that the computation offloading of U$_{ij}$ starts when l_{ij} bits of U$_{ij}$ are completely uploaded to the LEOS edge. In this case, the processing latency at the LEOS edge contains the computation and communication delays, where the feedback latency is negligible due to the same reason as mentioned above. Thus, the total latency required for carrying out the LEOS edge computing, T_{ij}^{LEO}, is given by

$$T_{ij}^{LEO} = \begin{cases} \frac{l_{ij}}{R_{ij}^{LEO}} + \frac{l_{ij}c_{ij}}{f_{ij}^{LEO}}, & u_{ij}=0, \\ 0, & \text{Otherwise.} \end{cases} \tag{12}$$

The latency of U_{ij} can be calculated by selecting the maximum value between those imposed by the local computing, the BS_i server's computing, and the LEOS edge computing, which is formulated by

$$\begin{aligned} T_{ij} &= \max\left\{T_{ij}^{l}, u_{ij}T_{ij}^{BS_i} + (1-u_{ij})T_{ij}^{LEO}\right\} \\ &= \max\left\{\frac{(L_{ij}-l_{ij})\,c_{ij}}{f_{ij}^{l}}, u_{ij}\left(\frac{l_{ij}}{R_{ij}^{BS_i}} + \frac{l_{ij}c_{ij}}{f_{ij}^{BS_i}}\right) + \right. \\ &\quad \left. (1-u_{ij})\left(\frac{l_{ij}}{R_{ij}^{LEO}} + \frac{l_{ij}c_{ij}}{f_{ij}^{LEO}}\right)\right\}. \end{aligned} \tag{13}$$

D. Energy Model

1) Local Energy: Let the power consumption of local processing be P_l, which is assumed to be identical for all ground users. Then, the energy dissipation of local processing at U_{ij} is given by

$$E_{ij}^{l} = \begin{cases} P_l\frac{(L_{ij}-l_{ij})c_{ij}}{f_{ij}^{l}} + p_{ij}\frac{l_{ij}}{R_{ij}^{BS_i}}, & u_{ij}=1, \\ P_l\frac{(L_{ij}-l_{ij})c_{ij}}{f_{ij}^{l}} + p_{ij}\frac{l_{ij}}{R_{ij}^{LEO}}, & u_{ij}=0. \end{cases} \tag{14}$$

In this case, the total energy dissipation of all ground users is $E_{total}^{U} = \sum_{i=1}^{I}\sum_{j=1}^{J} E_{ij}^{l}$, which is expressed by

$$\begin{aligned} E_{total}^{U} &= \sum_{i=1}^{I}\sum_{j=1}^{J} u_{ij}\left(\frac{P_l\,(L_{ij}-l_{ij})\,c_{ij}}{f_{ij}^{l}} + \frac{p_{ij}l_{ij}}{R_{ij}^{BS_i}}\right) \\ &\quad + \sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\left(\frac{P_l\,(L_{ij}-l_{ij})\,c_{ij}}{f_{ij}^{l}} + \frac{p_{ij}l_{ij}}{R_{ij}^{LEO}}\right). \end{aligned} \tag{15}$$

2) BS Server Energy: Let the power consumption for the transmission of U_{ij} to BS_i be $P_{ij}^{BS_i}$, and let the energy consumption of BS_i server processing within unit time be P_{BS_i}, which is assumed to be identical for all BS servers. Then, the energy dissipation required for processing the computation offloading of U_{ij} at BS_i is given by

$$E_{ij}^{BS_i} = \begin{cases} P_{ij}^{BS_i}\frac{l_{ij}}{R_{ij}^{BS_i}} + P_{BS_i}\frac{l_{ij}c_{ij}}{f_{ij}^{BS_i}}, & u_{ij}=1, \\ 0, & \text{Otherwise.} \end{cases} \tag{16}$$

In this case, the total energy dissipation of BS_i is $E_{total}^{BS_i} = \sum_{j=1}^{J} E_{ij}^{BS_i}$, and then the total energy dissipation of all BS servers is given by

$$E_{total}^{BS} = \begin{cases} \sum_{i=1}^{I}\sum_{j=1}^{J}\left(\frac{P_{ij}^{BS_i}l_{ij}}{R_{ij}^{BS_i}} + \frac{P_{BS_i}l_{ij}c_{ij}}{f_{ij}^{BS_i}}\right), & u_{ij}=1, \\ 0, & \text{Otherwise.} \end{cases} \tag{17}$$

3) LEOS Edge Energy: Let the power consumption for the transmission of U_{ij} to the LEOS edge be P_{ij}^{LEO}, and let the energy dissipation of the LEOS edge processing within unit time be P_{LEO}. Then, the energy dissipation required for processing the computation offloading of U_{ij} at the LEOS edge (i.e., $u_{ij}=0, l_{ij}\neq 0$) be given by

$$E_{ij}^{LEO} = \begin{cases} P_{ij}^{LEO}\frac{l_{ij}}{R_{ij}^{LEO}} + P_{LEO}\frac{l_{ij}c_{ij}}{f_{ij}^{LEO}}, & u_{ij}=0, \\ 0, & \text{Otherwise.} \end{cases} \tag{18}$$

In this case, the total energy dissipation of the LEO is $E_{total}^{LEO} = \sum_{i=1}^{I}\sum_{j=1}^{J} E_{ij}^{LEO}$, which is expressed as

$$E_{total}^{LEO} = \begin{cases} \sum_{i=1}^{I}\sum_{j=1}^{J}\left(\frac{P_{ij}^{LEO}l_{ij}}{R_{ij}^{LEO}} + \frac{P_{LEO}l_{ij}c_{ij}}{f_{ij}^{LEO}}\right), & u_{ij}=0, \\ 0, & \text{Otherwise.} \end{cases} \tag{19}$$

As a result, the overall energy dissipation of the proposed system is formulated by

$$\begin{aligned} E_s &= E_{total}^{U} + E_{total}^{BS} + E_{total}^{LEO} \\ &= \sum_{i=1}^{I}\sum_{j=1}^{J}\left(E_{ij}^{l} + u_{ij}E_{ij}^{BS_i} + (1-u_{ij})\,E_{ij}^{LEO}\right). \end{aligned} \tag{20}$$

Upon substituting (14), (16), and (18) into (20), the system's energy dissipation is given by (21) in the following page.

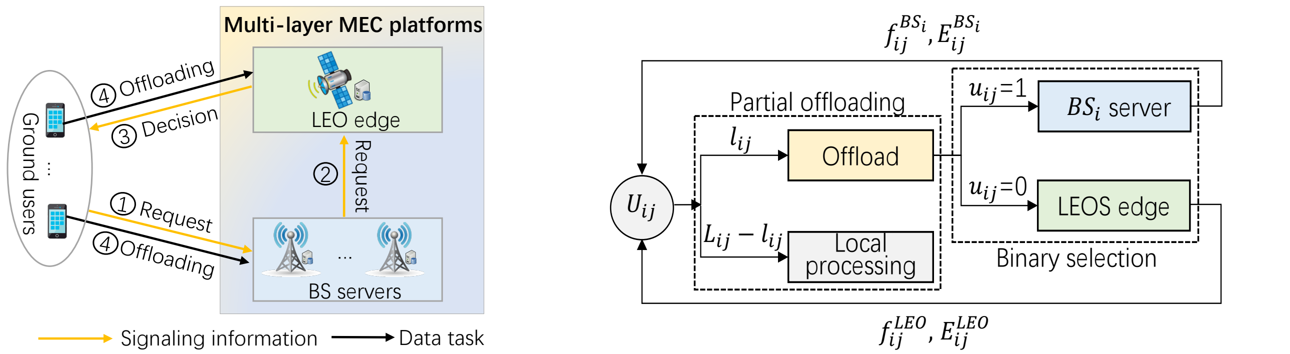
III. Offloading Design and Problem Formulation

In this section, the offloading design is presented first. Then, based on this, we aim for minimizing the energy dissipation of the LEOS edge-assisted system under the latency constraint.

A. Offloading Design

A LEOS edge-assisted offloading design for STINs is presented in Fig. 2. In this offloading design, since the coexistence of the LEOS edge and BS servers, the ground user can offload its task either to the BS server or to the LEOS edge for processing when the computational capability of the ground user cannot meet its task processing requirement. To meet the computation target of ground users, we propose an offloading access scheme including the request, decision, and offloading phases, as shown in Fig. 2(a), where the implementation of the offloading access scheme has to carry out the following three operations.

- **Offloading Request**. At the beginning of a time slot, each ground user ($\mathrm{U}_{ij}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}$) sends an offloading request to its serving BS ($\mathrm{BS}_i, \forall i \in \mathcal{I}$) over the C-band. Accordingly, each BS receives the offloading requests and forwards them to the LEOS over the Ka-band, where the classic OFDMA scheme is adopted both in the C-band and Ka-band.
- **Resource Allocation**. As the LEOS receives offloading requests from BSs, it formulates an offloading policy for allocating the communication and computation resources requested by each ground user (U_{ij}), and broadcasts the resource allocation results to all ground users and BSs.



(a) Offloading access

(b) Offloading policy

Fig. 2: The offloading access and policy designed for STINs

$$E_s=\sum_{i=1}^{I}\sum_{j=1}^{J}\left(\frac{P_l(L_{ij}-l_{ij})\,c_{ij}}{f_{ij}^{l}}+u_{ij}\frac{p_{ij}l_{ij}}{R_{ij}^{BS_i}}+(1-u_{ij})\frac{p_{ij}l_{ij}}{R_{ij}^{LEO}}+u_{ij}\left(\frac{P_{ij}^{BS_i}l_{ij}}{R_{ij}^{BS_i}}+\frac{P_{BS_i}l_{ij}c_{ij}}{f_{ij}^{BS_i}}\right)+(1-u_{ij})\left(\frac{P_{ij}^{LEO}l_{ij}}{R_{ij}^{LEO}}+\frac{P_{LEO}l_{ij}c_{ij}}{f_{ij}^{LEO}}\right)\right). \tag{21}$$

- **Task Processing**. Based on the received optimal allocation results, user U_{ij} offloads its tasks to the BS or the LEOS-edge for processing.

An intuitive example of the offloading policy is shown in Fig. 2(b), where U_{ij} carries out local processing if $l_{ij}=0$; otherwise, it offloads its partial task to BS_i server for processing when $u_{ij}=1$; or offloads its partial task to the LEOS edge for processing when $u_{ij}=0$. In other words, U_{ij} carries out partial offloading and binary selection for task processing after obtaining the optimization results from the LEOS. In this process, once an offloading is required by U_{ij}, the connected BS_i server and its computation resources (i.e., having $f_{ij}^{BS_i}, \forall u_{ij}=1$), or the LEOS edge and its computation resources (i.e., having $f_{ij}^{LEO}, \forall u_{ij}=0$) will be allocated to U_{ij} for its task processing. Note that the optimization of the system's offloading policy will be completed at the LEOS because the individual BSs are unaware of the computational capability of the LEOS edge.

B. Problem Formulation

As expressed in (21), we define the sum energy dissipation of the LEOS edge-assisted offloading system as the energy dissipation of all the ground users associated with different BS servers and the LEOS edge. Based on the offloading access and policy illustrated in Fig. 2, we aim for minimizing the system's energy dissipation while constraining the delay of the user, by jointly optimizing the offloading mode matrix $\mathbf{U}=[\mathbf{u}_1,\ldots,\mathbf{u}_i,\ldots,\mathbf{u}_I]^T\in\mathbb{C}^{I\times J}$, the offloading volume matrix $\mathbf{L}=[\mathbf{l}_1,\ldots,\mathbf{l}_i,\ldots,\mathbf{l}_I]^T\in\mathbb{C}^{I\times J}$, the BS server computing resource allocation matrix $\mathbf{F}^{BS}=[\mathbf{f}_1^{BS},\ldots,\mathbf{f}_i^{BS},\ldots,\mathbf{f}_I^{BS}]^T\in\mathbb{C}^{I\times J}$, and the LEOS edge computing resource allocation matrix $\mathbf{F}^{LEO}=[\mathbf{f}_1^{LEO},\ldots,\mathbf{f}_i^{LEO},\ldots,\mathbf{f}_I^{LEO}]^T\in\mathbb{C}^{I\times J}$, where the related J-dimensional vectors are denoted by $\mathbf{u}_i=[u_{i1},\ldots,u_{ij},\ldots,u_{iJ}]^T$, $\mathbf{l}_i=[l_{i1},\ldots,l_{ij},\ldots,l_{iJ}]^T$, $\mathbf{f}_i^{BS}=[f_{i1}^{BS_i},\ldots,f_{ij}^{BS_i},\ldots,f_{iJ}^{BS_i}]^T$, and $\mathbf{f}_i^{LEO}=[f_{i1}^{LEO},\ldots,f_{ij}^{LEO},\ldots,f_{iJ}^{LEO}]^T$, respectively. Therefore, a system energy minimization problem is formulated as follows:

$$\mathcal{P}0:\ \min_{\{\mathbf{U},\mathbf{L},\mathbf{F}^{BS},\mathbf{F}^{LEO}\}}\ E_s \tag{22}$$
$$\text{s.t.}\ T_{ij}\le T_{ij}^{max},\ \forall i\in\mathcal{I},\ \forall j\in\mathcal{J}, \tag{22a}$$
$$u_{ij}\in\{0,1\},\ \forall i\in\mathcal{I},\ \forall j\in\mathcal{J}, \tag{22b}$$
$$\sum_{j=1}^{J}u_{ij}f_{ij}^{BS_i}\le f_{max}^{BS_i},\ \forall i\in\mathcal{I}, \tag{22c}$$
$$E_{total}^{BS_i}\le E_{max}^{BS_i},\ \forall i\in\mathcal{I}, \tag{22d}$$
$$\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\,f_{ij}^{LEO}\le f_{max}^{LEO}, \tag{22e}$$
$$E_{total}^{LEO}\le E_{max}^{LEO}, \tag{22f}$$
$$\sum_{j=1}^{J}u_{ij}\le J,\ \forall i\in\mathcal{I}, \tag{22g}$$
$$\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\le IJ, \tag{22h}$$
$$l_{ij}\in[0,L_{ij}],\ \forall i\in\mathcal{I},\ \forall j\in\mathcal{J}, \tag{22i}$$
$$f_{ij}^{BS}\ge 0, f_{ij}^{LEO}\ge 0,\ \forall i\in\mathcal{I},\ \forall j\in\mathcal{J}, \tag{22j}$$
$$E_{ij}^{BS}\ge 0, E_{ij}^{LEO}\ge 0,\ \forall i\in\mathcal{I},\ \forall j\in\mathcal{J}. \tag{22k}$$

The constraints in Problem $\mathcal{P}0$ are detailed as follows: (22a) indicates that the latency of U_{ij} is limited by the maximum latency T_{ij}^{max}, (22b) indicates the offloading mode of U_{ij}, where u_{ij} is a binary value, $u_{ij}=1$ represents that U_{ij} offloads its task to BS_i over the C-band, and $u_{ij}=0$ represents that U_{ij} offloads its task to the LEOS edge over the Ka-band, (22c) and (22d) indicate that the total computational capability and energy consumed by offloading at BS_i has to be less than its maximum computational capability ($f_{max}^{BS_i}$) and maximum energy dissipation ($E_{max}^{BS_i}$), respectively, (22e) and (22f) indicate that the total computational capability and

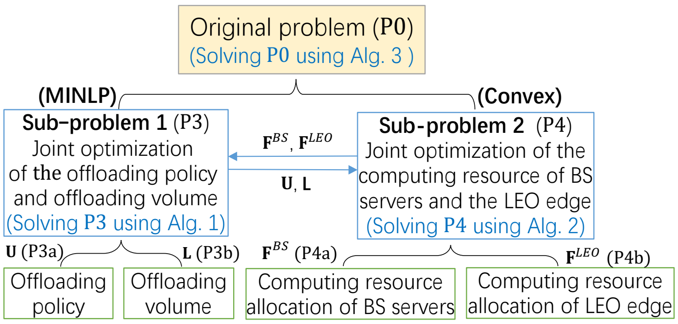


Fig. 3: Energy Efficient offloading problem decomposition and solution roadmap.

energy consumed by offloading at the LEOS edge have to be less than its maximum computationl capability (f_{max}^{LEO}) and maximum energy dissipation (E_{max}^{LEO}), respectively, (22g) indicates that at most J ground users can be served by each BS server. (22h) indicates that at most IJ ground users can be served by the LEOS edge, (22i) specifies the range of the amount of offloading, specifically, $l_{ij} = 0$ represents fully local processing at U_{ij}, while $l_{ij} = L_{ij}$ indicates full offloading at U_{ij}, (22j) and (22k) represent the feasibility value of $f_{ij}^{BS_i}$, f_{ij}^{LEO}, $E_{ij}^{BS_i}$, and E_{ij}^{LEO}.

Remark 1. *In Problem $\mathcal{P}0$, there is a total of four optimization variables, namely, the offloading volume matrix, the offloading mode matrix, the BS server computing resource allocation matrix, and the LEOS edge computing resource allocation matrix. The optimization of the former two variables is related to the offloading decision, and the optimization of the last two variables is related to the computing resource allocation of BS servers and the LEOS edge. We observe that the formulated Problem $\mathcal{P}0$ is an MINLP problem, which is NP-hard and whose globally optimal solution is difficult to obtain by using the common standard optimization approaches. Therefore, it is necessary to transform the original Problem $\mathcal{P}0$ into some tractable sub-problems that can be solved separately and alternatively over multiple iterations. Hence, the alternating optimization (AO) method is invoked for solving the original problem in an efficient manner, which is detailed in the following solution roadmap.*

IV. Energy Efficient Offloading Solution

In this section, we present our solution roadmap on $\mathcal{P}0$. As shown in Fig. 3, we first decompose the original optimization problem $\mathcal{P}0$ into an MINLP problem and a convex problem. Then we optimized the offloading volume and mode using the popular Semi-Definite Program (SDP) approach. We then continue by decomposing the joint optimization problem of the computing resource allocation of BS servers and LEOS edge by decoupling and solving the decomposed subproblems using the classic Lagrangian multiplier-based approach.

A. Solution Roadmap

Due to the intractability of original problem, $\mathcal{P}0$ is decomposed into two sub-problems, namely, a joint optimization problem of the offloading volume and offloading mode, which is an MINLP problem, and a joint optimization problem of the computing resource of BS servers and LEOS edge, which is a convex problem. By using the AO approach, these two sub-problems are solved in an alternating way. Specifically, at the first iteration, the computing resource allocation of the BS servers and of the LEOS is given and input to the joint optimization problem of offloading volume and offloading mode. Then, the first decomposed problem is solved, and the resultant $\mathbf{L}$ and $\mathbf{U}$ are entered into the joint computing resource allocation optimization problem of BS servers and LEOS edge. Afterward, the derived computing resource allocation of BS servers and the LEOS are solved by decoupling, yielding $\mathbf{F}^{BS}$ and $\mathbf{F}^{LEO}$. These will then be input into the joint optimization problem of the offloading volume and offloading mode during the second iteration, and then the steps of the first iteration will be repeated. This procedure will be continued until convergence is reached. Within the AO approach invoked, the proposed energy efficient offloading solution is detailed in the following section.

In Problem $\mathcal{P}0$, we substitute (16) and (18) into the energy constraints of BS servers and the LEOS edge in (22d) and (22f). Therefore, Problem $\mathcal{P}0$ is rewritten as

$$\mathcal{P}1: \min_{\{\mathbf{U},\mathbf{L},\mathbf{F}^{BS},\mathbf{F}^{LEO}\}} E_s \tag{23}$$

$$\text{s.t.} \quad (22a)-(22c),(22e),(22g)-(22j). \tag{23a}$$

$$\sum_{j=1}^{J} u_{ij} l_{ij} \left(\frac{P_{ij}^{BS_i}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} c_{ij}}{f_{ij}^{BS_i}} \right) \leq E_{max}^{BS_i}, \quad \forall i \in \mathcal{I}, \tag{23b}$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J} (1-u_{ij}) l_{ij} \left(\frac{P_{ij}^{LEO}}{R_{ij}^{LEO}} + \frac{P_{LEO} c_{ij}}{f_{ij}^{LEO}} \right) \leq E_{max}^{LEO}. \tag{23c}$$

B. Joint Optimization of Offloading Volume and Mode, While Fixing the Computing Resource Allocation of BS Servers and the LEOS Edge

Given the computing resource allocation matrix of BS servers and the LEOS edge, i.e., fixed $\mathbf{F}^{BS}$ and $\mathbf{F}^{LEO}$, Problem $\mathcal{P}1$ in (23) can be reformulated as

$$\mathcal{P}2: \min_{\{\mathbf{U},\mathbf{L}\}} E_s \tag{24}$$

$$\text{s.t.} \quad (22a),(22b),(22g)-(22i),(23b),(23c). \tag{24a}$$

Problem $\mathcal{P}2$ is still challenging to solve because it is an MINLP problem, where the objective function and constraints consist of the binary variable u_{ij} as well as continuous variable l_{ij}, and constraint (22a) also includes the '$\max\{\cdot\}$' function. To solve Problem $\mathcal{P}2$, we transform (24) into an equivalent form as shown in Proposition 1.

Proposition 1. *Problem* (24) *can be transformed into the following equivalent problem:*

$$\mathcal{P}3: \min_{\{\mathbf{U},\mathbf{L}\}} \ E_s \tag{25}$$

$$\text{s.t.} \ (22b), (22g)-(22i), (23b), (23c). \tag{25a}$$

$$\frac{(L_{ij}-l_{ij})\, c_{ij}}{f_{ij}^{l}} \leq T_{ij}^{max}, \ \forall i \in \mathcal{I}, \ \forall j \in \mathcal{J}, \tag{25b}$$

$$u_{ij} l_{ij} \left(\frac{1}{R_{ij}^{BS_i}} + \frac{c_{ij}}{f_{ij}^{BS_i}} \right) \leq T_{ij}^{max}, \ \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \tag{25c}$$

$$(1-u_{ij}) l_{ij} \left(\frac{1}{R_{ij}^{LEO}} + \frac{c_{ij}}{f_{ij}^{LEO}} \right) \leq T_{ij}^{max}, \ \forall i \in \mathcal{I}, \forall j \in \mathcal{J}. \tag{25d}$$

Proof. see Appendix A. □

By observing (25), it can be seen that Problem $\mathcal{P}3$ is also an MINLP problem because of the coupling of the binary variable u_{ij} and continuous variable l_{ij}. To effectively solve Problem $\mathcal{P}3$, we decompose it into two sub-problems: the optimization of the offloading mode and volume.

1) Optimization of Offloading Mode: The offloading mode matrix $\mathbf{U}$ is optimized, while fixing both the offloading volume matrix $\mathbf{L}$, as well as the BS server computing resource allocation matrix $\mathbf{F}^{BS}$, and the LEOS edge computing resource allocation matrix $\mathbf{F}^{LEO}$, yielding

$$\mathcal{P}3a: \min_{\mathbf{U}} \ E_s \tag{26}$$

$$\text{s.t.} \ (22b), (22c), (22e), (22g), (22h), (23b), (23c), (25c), (25d). \tag{26a}$$

Observe from (26), that given the value of $\mathbf{L}, \mathbf{F}^{BS}$, and $\mathbf{F}^{LEO}$, Problem $\mathcal{P}3a$ becomes an '0-1' integer programming problem, because the objective function and constraints consist of the binary variable u_{ij}. To effectively solve Problem $\mathcal{P}3a$, we propose a separable SDP approach for finding the optimal binary policy related to $\mathbf{U}$ (i.e., the solution of (26)). Specifically, we first use semidefinite relaxation to obtain a fractional solution, and then we use the rounding technique of Shmoys and Tardos [45] for recovering the binary value.

It is necessary to relax the binary variables u_{ij} into continuous variables $u_{ij} \in [0, 1]$. The relaxed variable u_{ij} can be viewed as the probability that U_{ij} offloads its task to BS_i server for processing, while $1 - u_{ij}$ can be viewed as the probability that U_{ij} offloads its task to the LEOS edge for processing. Hence, Problem $\mathcal{P}3a$ can be transformed into an equivalent form, as shown in Proposition 2.

Proposition 2. *Problem* (26) *can be transformed into the following problem.*

$$\overline{\mathcal{P}}3a: \min_{\mathbf{W}} \ \sum_{i=1}^{I}\sum_{j=1}^{J} \mathrm{Tr}(\mathbf{A}_0\mathbf{W}) \tag{27}$$

$$\text{s.t.} \ \mathrm{Tr}(\mathbf{A}_1\mathbf{W}) \leq f_{max}^{BS_i}, \tag{27a}$$

$$\mathrm{Tr}(\mathbf{A}_2\mathbf{W}) \leq J, \tag{27b}$$

$$\mathrm{Tr}(\mathbf{A}_3\mathbf{W}) \leq 0, \tag{27c}$$

$$\mathrm{Tr}(\mathbf{A}_4\mathbf{W}) \leq E_{max}^{BS_i}, \tag{27d}$$

$$\mathrm{Tr}(\mathbf{A}_5\mathbf{W}) \leq E_{max}^{LEO} - \sum_{i=1}^{I}\sum_{j=1}^{J} \Lambda_{ij}^{E_L}, \tag{27e}$$

$$\mathrm{Tr}(\mathbf{A}_6\mathbf{W}) \leq T_{ij}^{max}, \tag{27f}$$

$$\mathrm{Tr}(\mathbf{A}_7\mathbf{W}) \leq T_{ij}^{max} - \Lambda_{ij}^{T_L}, \tag{27g}$$

$$\mathrm{Tr}(\mathbf{A}_8\mathbf{W}) = 0, \tag{27h}$$

$$\mathbf{W}(IJ, IJ) = 1, \tag{27i}$$

$$\mathbf{W} \geq 0, \tag{27j}$$

where $\mathbf{W} \triangleq [q^{\mathrm{T}}, 1]^{\mathrm{T}}[q^{\mathrm{T}}, 1]$, $q = [u_{11}, \ldots, u_{ij}, \ldots, u_{IJ}]^{\mathrm{T}}$, $\Lambda_{ij}^{E_L} = l_{ij}\left(\frac{P_{ij}^{LEO}}{R_{ij}^{LEO}} + \frac{P_{LEO}c_{ij}}{f_{ij}^{LEO}}\right)$, *and* $\Lambda_{ij}^{T_L} = l_{ij}\left(\frac{1}{R_{ij}^{LEO}} + \frac{c_{ij}}{f_{ij}^{LEO}}\right)$. *The matrices* $\mathbf{A}_0$ *to* $\mathbf{A}_8$ *are non-negative diagonal matrices, which are given in Appendix B. Observe that* (27) *is a convex problem, and its optimal offloading mode solution* $\mathbf{W}^*$ *can be obtained in polynomial time using a standard SDP solver.*

Proof. see Appendix B. □

Remark 2. *Since Problem* $\overline{\mathcal{P}}3a$ *is a relaxation of Problem* $\mathcal{P}3a$*, the optimal solution of Problem* $\overline{\mathcal{P}}3a$ *is the lower bound of the optimal solution of* $\mathcal{P}3a$ *provided that* $\mathrm{rank}(\mathbf{W}^*) \neq 1$. *Therefore, it is necessary to recover a rank-1 solution from* $\{\mathbf{W}^*\}$. *The rounding technique of [20], [45] is used for binary value recovery, which contains the following three steps: 1) obtain a fractional solution of* $\mathbf{W}^*$*, 2) construct a weighted bipartite graph to build the relationship between ground users and offloading modes, 3) find an integer matching for obtaining the binary solution.*

2) Optimization of Offloading Volume: The offloading volume optimization matrix $\mathbf{L}$ is optimized, while fixing both the offloading mode matrix $\mathbf{U}$, as well as the BS server computing resource allocation matrix $\mathbf{F}^{BS}$, and the LEO edge computing resource allocation matrix $\mathbf{F}^{LEO}$, yielding

$$\mathcal{P}3b: \min_{\mathbf{L}} \ E_s \tag{28}$$

$$\text{s.t.} \ (22i), (23b), (23c), (25b)-(25d). \tag{28a}$$

It can be seen that the objective function and all constraint functions in Problem $\mathcal{P}3b$ are linear combinations of the continuous variable l_{ij}. Thus, $\mathcal{P}3b$ is a linear programming problem. Algorithm 1 summarizes the solution of Problem $\mathcal{P}3$.

Algorithm 1: Joint optimization of offloading volume and mode for a given computing resource allocation of BS servers and the LEOS edge

---

Initialization: B_C, K_C^i, h_{ij}^C, B_{Ka}, K_{Ka}^i, h_{ij}^{Ka}, σ^2, $\hat{\sigma}^2$, p_{ij}, L_{ij}, c_{ij}, f_{ij}^l, P_l, $P_{ij}^{BS_i}$, P_{BS_i}, P_{ij}^{LEO}, P_{LEO} $f_{max}^{BS_i}$, f_{max}^{LEO}, $E_{max}^{BS_i}$, E_{max}^{LEO}, and T_{ij}^{max}, the maximum iteration number is L_1, and set $l_1 = 0$;

1: Calculate $R_{ij}^{BS_i}$ and R_{ij}^{LEO} according to (4) and (8), where $\forall i \in \mathcal{I},\ \forall j \in \mathcal{J}$;
2: **repeat**
3: Given $\mathbf{F}^{BS}, \mathbf{F}^{LEO}$ and $\mathbf{L}$, solve $\mathbf{U}$ using (27);
4: Given $\mathbf{F}^{BS}, \mathbf{F}^{LEO}$ and $\mathbf{U}$, solve $\mathbf{L}$ using (28);
5: Updates $l_1 \leftarrow l_1 + 1$;
6: **until** $l_1 \geq L_1$;
7: $\mathbf{U}^* = \mathbf{U}$, $\mathbf{L}^* = \mathbf{L}$;

Output: $\mathbf{U}^*$ and $\mathbf{L}^*$.

C. Joint Optimization of the Computing Resource Allocation of BS Servers and of the LEOS Edge, While Fixing the Offloading Volume and Mode

Given an offloading volume matrix $\mathbf{L}$ and offloading mode matrix $\mathbf{U}$, Problem $\mathcal{P}1$ can be reformulated as

$$\mathcal{P}4 \quad : \min_{\{\mathbf{F_{BS}}, \mathbf{F_{LEO}}\}} \quad E_s \tag{29}$$

$$\text{s.t.} \quad (22c), (22e), (22j), (23b), (23c), (25c), (25d). \tag{29a}$$

It may be observed that Problem $\mathcal{P}4$ is a decoupling problem, since the BS offloading and the LEOS edge offloading are decoupled when the offloading volume matrix $\mathbf{L}$ and the offloading mode matrix $\mathbf{U}$ are given. Problem $\mathcal{P}4$ can be decomposed into the following two sub-problems: the computing resource optimization of BS servers and that of the LEOS edge, as shown in Algorithm 2.

1) Optimization of the BS Servers' Computing Resource: The BS server computing resource allocation matrix $\mathbf{F_{BS}}$ is optimized, while fixing both the LEOS edge computing resource allocation matrix $\mathbf{F_{LEO}}$, as well as the offloading volume matrix $\mathbf{L}$, and the offloading mode matrix $\mathbf{U}$, yielding

$$\mathcal{P}4a: \ \min_{\mathbf{F_{BS}}} \ \sum_{i=1}^{I}\sum_{j=1}^{J} u_{ij}\left(\frac{P_{ij}^{BS_i} l_{ij}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} l_{ij} c_{ij}}{f_{ij}^{BS_i}}\right), \tag{30}$$

$$\text{s.t.} \quad (22c), (23b), (25c). \tag{30a}$$

$$f_{ij}^{BS} \geq 0, \ \ \forall i \in \mathcal{I}, \ \forall j \in \mathcal{J}, \tag{30b}$$

Proposition 3. *Problem $\mathcal{P}4a$ in* (30) *is a convex optimization problem. The Karush Kuhn Tucker (KKT) technique can be employed for solving it.*

Proof. see Appendix C. □

Specifically, the Lagrangian function of Problem $\mathcal{P}4a$ in (30) is formulated by

$$\mathcal{L}(f_{ij}^{BS_i}, \lambda_1, \lambda_2, \lambda_3) = \sum_{i=1}^{I}\sum_{j=1}^{J} u_{ij}\left(\frac{P_{ij}^{BS_i} l_{ij}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} l_{ij} c_{ij}}{f_{ij}^{BS_i}}\right) \tag{31}$$

$$- \lambda_1\left(\sum_{j=1}^{J} u_{ij} f_{ij}^{BS_i} - f_{max}^{BS_i}\right)$$

$$- \lambda_2\left(\sum_{j=1}^{J} u_{ij} l_{ij}\left(\frac{P_{ij}^{BS_i}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} c_{ij}}{f_{ij}^{BS_i}}\right) - E_{max}^{BS_i}\right)$$

$$- \lambda_3\left(u_{ij} l_{ij}\left(\frac{1}{R_{ij}^{BS_i}} + \frac{c_{ij}}{f_{ij}^{BS_i}}\right) - T_{ij}^{max}\right),$$

where the variables λ_1, λ_2 and λ_3 are the non-negative Lagrange multipliers. The optimal computing resource allocation $f_{ij}^{BS_i\,*}$ at the BS$_i$ server, the optimal Lagrange multipliers λ_1^*, λ_2^* and λ_3^* should satisfy the following KKT conditions for $i \in \mathcal{I}, j \in \mathcal{J}$:

$$\frac{\partial \mathcal{L}}{\partial f_{ij}^{BS_i}} = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{-u_{ij} P_{BS_i} l_{ij} c_{ij}}{f_{ij}^{BS_i\,*2}} - \lambda_1^* \sum_{j=1}^{J} u_{ij} + \lambda_2^* \sum_{j=1}^{J} \frac{u_{ij} P_{BS_i} l_{ij} c_{ij}}{f_{ij}^{BS_i\,*2}} + \lambda_3^* \frac{u_{ij} l_{ij} c_{ij}}{f_{ij}^{BS_i\,*2}} = 0, \tag{32}$$

$$\lambda_1^*\left(\sum_{j=1}^{J} u_{ij} f_{ij}^{BS_i\,*} - f_{max}^{BS_i}\right) = 0, \tag{33}$$

$$\lambda_2^*\left(\sum_{j=1}^{J} u_{ij} l_{ij}\left(\frac{P_{ij}^{BS_i}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} c_{ij}}{f_{ij}^{BS_i\,*}}\right) - E_{max}^{BS_i}\right) = 0, \tag{34}$$

$$\lambda_3^*\left(u_{ij} l_{ij}\left(\frac{1}{R_{ij}^{BS_i}} + \frac{c_{ij}}{f_{ij}^{BS_i\,*}}\right) - T_{ij}^{max}\right) = 0, \tag{35}$$

$$f_{ij}^{BS_i\,*} \geq 0. \tag{36}$$

The value of $f_{ij}^{BS_i\,*}$ can be directly derived from (32)-(36), which is given by

$$f_{ij}^{BS_i\,*} = \frac{u_{ij} l_{ij} c_{ij}}{T_{ij}^{max} - u_{ij} l_{ij}/R_{ij}^{BS_i}}. \tag{37}$$

Then, we can express λ_1^*, λ_1^* and λ_3^* as follows

$$\begin{cases} \lambda_1^* &= 0, \\ \lambda_2^* &= 0, \\ \lambda_3^* &= \frac{P_{BS_i} \sum_{i=1}^{I}\sum_{j=1}^{J} (T_{ij}^{max} - u_{ij} l_{ij}/R_{ij}^{BS_i})^2}{(T_{ij}^{max} - u_{ij} l_{ij}/R_{ij}^{BS_i})^2}. \end{cases} \tag{38}$$

2) Optimization of the LEOS Edge Computing Resource: The LEOS edge computing resource allocation matrix $\mathbf{F_{LEO}}$ is optimized, while fixing both the BS server computing

Algorithm 2: Joint optimization of the computation resource allocation of BS servers and the LEOS edge for a given offloading volume and mode

Initialization: B_C, K_C^i, h_{ij}^C, B_{Ka}, K_{Ka}^i, h_{ij}^{Ka}, σ^2, $\hat{\sigma}^2$, p_{ij}, L_{ij}, c_{ij}, f_{ij}^l, P_l, $P_{ij}^{BS_i}$, P_{BS_i}, P_{ij}^{LEO}, P_{LEO} $f_{max}^{BS_i}$, f_{max}^{LEO}, $E_{max}^{BS_i}$, E_{max}^{LEO}, and T_{ij}^{max}, the maximum iteration number is L_2, and set $l_2 = 0$;

1: Calculate $R_{ij}^{BS_i}$ and R_{ij}^{LEO} according to (4) and (8), where $\forall i \in \mathcal{I}, \ \forall j \in \mathcal{J}$;
2: **repeat**
3: Given $\mathbf{U}, \mathbf{L}$, and $\mathbf{F}^{LEO}$, solve $\mathbf{F}^{\mathbf{BS}}$ using (37);
4: Given $\mathbf{U}, \mathbf{L}$, and $\mathbf{F}^{BS}$, solve $\mathbf{F}^{\mathbf{LEO}}$ using (46);
5: Updates $l_2 \leftarrow l_2 + 1$;
6: **until** $l_2 \geq L_2$;
7: $\mathbf{F}^{BS^*} = \mathbf{F}^{BS}$, $\mathbf{F}^{LEO^*} = \mathbf{F}^{LEO}$;

Output: $\mathbf{F}^{BS^*}$ and $\mathbf{F}^{LEO^*}$.

resource allocation matrix $\mathbf{F_{BS}}$, as well as the offloading volume matrix $\mathbf{L}$, and the offloading mode matrix $\mathbf{U}$, yielding

$$\mathcal{P}4b: \min_{\mathbf{F_{LEO}}} \ \sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\left(\frac{P_{ij}^{LEO}l_{ij}}{R_{ij}^{LEO}}+\frac{P_{LEO}l_{ij}c_{ij}}{f_{ij}^{LEO}}\right), \tag{39}$$

$$\text{s.t.} \quad (22e), (23c), (25d). \tag{39a}$$

$$f_{ij}^{LEO} \geq 0, \ \ \forall i \in \mathcal{I}, \ \forall j \in \mathcal{J}, \tag{39b}$$

Proposition 4. *Problem $\mathcal{P}4b$ in* (39) *is also a convex optimization problem. The Karush Kuhn Tucker (KKT) conditious can be imposed on the problem for finding its optimal solution.*

Proof. see Appendix D. □

Specifically, the Lagrangian function of Problem $\mathcal{P}4b$ in (39) is derived as:

$$\begin{aligned}
&\mathcal{L}(f_{ij}^{LEO}, \mu_1, \mu_2, \mu_3) \\
&= \sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\left(\frac{P_{ij}^{LEO}l_{ij}}{R_{ij}^{LEO}}+\frac{P_{LEO}l_{ij}c_{ij}}{f_{ij}^{LEO}}\right) \\
&- \mu_1\left(\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\,f_{ij}^{LEO} - f_{max}^{LEO}\right) \\
&- \mu_2\left(\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})l_{ij}\left(\frac{P_{ij}^{LEO}}{R_{ij}^{LEO}}+\frac{P_{LEO}c_{ij}}{f_{ij}^{LEO}}\right)- E_{max}^{LEO}\right) \\
&- \mu_3\left((1-u_{ij})l_{ij}\left(\frac{1}{R_{ij}^{LEO}}+\frac{c_{ij}}{f_{ij}^{LEO}}\right)- T_{ij}^{max}\right),
\end{aligned} \tag{40}$$

where the variables μ_1, μ_2 and μ_3 are the non-negative Lagrange multipliers, while the optimal computing resource allocation $f_{ij}^{LEO^*}$ at the LEOS edge, and the optimal Lagrange multipliers μ_1^*, μ_2^* and μ_3^* should satisfy the following KKT conditions for $i \in \mathcal{I}, j \in \mathcal{J}$:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial f_{ij}^{LEO}} &= \sum_{i=1}^{I}\sum_{j=1}^{J}\frac{(1-u_{ij})P_{LEO}l_{ij}c_{ij}}{f_{ij}^{LEO^{*2}}} \\
&- \mu_1^*\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij}) \\
&+ \mu_2^*\sum_{i=1}^{I}\sum_{j=1}^{J}\frac{(1-u_{ij})P_{LEO}l_{ij}c_{ij}}{f_{ij}^{LEO^{*2}}} \\
&+ \mu_3^*\frac{(1-u_{ij})l_{ij}c_{ij}}{f_{ij}^{LEO^{*2}}} = 0,
\end{aligned} \tag{41}$$

$$\mu_1^*\left(\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\,f_{ij}^{LEO^*} - f_{max}^{LEO}\right) = 0, \tag{42}$$

$$\mu_2^*\left(\sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})l_{ij}\left(\frac{P_{ij}^{LEO}}{R_{ij}^{LEO}}+\frac{P_{LEO}c_{ij}}{f_{ij}^{LEO^*}}\right)-E_{max}^{LEO}\right)=0, \tag{43}$$

$$\mu_3^*\left((1-u_{ij})\,l_{ij}\left(\frac{1}{R_{ij}^{LEO}}+\frac{c_{ij}}{f_{ij}^{LEO^*}}\right) - T_{ij}^{max}\right) = 0, \tag{44}$$

$$f_{ij}^{LEO^*} \geq 0. \tag{45}$$

The value of $f_{ij}^{LEO^*}$ can be directly derived from (41)-(45), which is given by

$$f_{ij}^{LEO^*} = \frac{(1-u_{ij})l_{ij}c_{ij}}{T_{ij}^{max}-(1-u_{ij})l_{ij}/R_{ij}^{LEO}}. \tag{46}$$

Then, we can formulate μ_1^*, μ_1^* and μ_3^* as follows

$$\begin{cases}
\mu_1^* &= 0, \\
\mu_2^* &= 0, \\
\mu_3^* &= \frac{P_{LEO}\sum_{i=1}^{I}\sum_{j=1}^{J}(T_{ij}^{max}-(1-u_{ij})l_{ij}/R_{ij}^{LEO})^2}{(T_{ij}^{max}-(1-u_{ij})l_{ij}/R_{ij}^{LEO})^2}.
\end{cases} \tag{47}$$

Based on the above discussions, Algorithm 3 presents the solution of Problem $\mathcal{P}0$.

D. Complexity Analysis

In Algorithm 1, the complexity of solving the convex problems of (27) and (28) is of polynomial order in the number of variables and constraints. Specifically, problem (27) is with $a = IJ$ decision variables, and $b = (3I + 7IJ)$ constraints. Refer to [20], if the interior-point method is considered, the worst-case computational complexity worst-case computational complexity required to solve (27) is $\mathcal{O}\left((a^2b + a^3)b^{\frac{1}{2}}L_1\right)$. Problem (28) is associated with $a' = IJ$ decision variables and $b' = (I + 5IJ)$ constraints such that the complexity is $\mathcal{O}\left((I + 6IJ)L_1\right)$. In Algorithm 2, since the closed-form is obtained in (37) and (46), the computational complexity is $\mathcal{O}\left(L_2\right)$. Thus, the overall complexity for solving $\mathcal{P}0$ in Algorithm 3 is given by $\mathcal{O}\left((a^2b + a^3)b^{\frac{1}{2}}(a' + b')L_1L_2L_3\right)$, which shows that the complexity of problem-solving is low. Moreover, the proposed overall algorithm converges to a locally optimal solution as long as the number of iterations is sufficiently large.

Algorithm 3: Joint optimization of the offloading volume, the offloading mode and the computing resource allocation of BS servers and the LEOS edge

Initialization: B_C, K_C^i, h_{ij}^C, B_{Ka}, K_{Ka}^i, h_{ij}^{Ka}, σ^2, $\hat{\sigma}^2$, p_{ij}, L_{ij}, c_{ij}, f_{ij}^l, P_l, $P_{ij}^{BS_i}$, P_{BS_i}, P_{ij}^{LEO}, P_{LEO} $f_{max}^{BS_i}$, f_{max}^{LEO}, $E_{max}^{BS_i}$, E_{max}^{LEO} and T_{ij}^{max}, the maximum iteration number is L_3, and set $l_3 = 0$;

1: Calculate $R_{ij}^{BS_i}$ and R_{ij}^{LEO} according to (4) and (8), where $\forall i \in \mathcal{I}$, $\forall j \in \mathcal{J}$;
2: **repeat**
3: Joint optimization of $\mathbf{U}$ and $\mathbf{L}$, given $\mathbf{F}^{BS}$ and $\mathbf{F}^{LEO}$, solve $\mathbf{U}$ and $\mathbf{L}$ using Algorithm 1;
4: Joint optimization of $\mathbf{F}^{BS}$ and $\mathbf{F}^{LEO}$, given $\mathbf{U}$ and $\mathbf{L}$, solve $\mathbf{U}$ and $\mathbf{L}$ using Algorithm 2;
5: Updates $l_3 \leftarrow l_3 + 1$;
6: **until** $l_3 \geq L_3$;
7: $\mathbf{U}^* = \mathbf{U}$, $\mathbf{L}^* = \mathbf{L}$, $\mathbf{F}^{BS^*} = \mathbf{F}^{BS}$, $\mathbf{F}^{LEO^*} = \mathbf{F}^{LEO}$;

Output: $\mathbf{U}^*, \mathbf{L}^*, \mathbf{F}^{BS^*}$ and $\mathbf{F}^{LEO^*}$.

TABLE I: SYSTEM PARAMETERS

Parameter	Value	Parameter	Value
I	2	J	10
B_C	500 MHz	B_{Ka}	500 MHz
f_{ij}^l	10^5 CPU cycle/s	L_{ij}	10 Mbit
$f_{max}^{BS_i}$	10^9 CPU cycle/s	α	2
f_{max}^{LEO}	2×10^{10} CPU cycle/s	ξ	1
c_{ij}	300 cycle / bit	σ^2	7.9e-13 mW
P_l	0.001 J/s	T_{ij}^{max}	1 s
P_{LEO}	1 J/s	P_{BS_i}	1 J/s

V. NUMERICAL RESULTS

A. Scenario Settings

1) Topology: We consider a network scenario that consists of a LEOS edge and two BS servers. Each BS server covers 10 ground users. All ground users are uniformly distributed in a circular area with a radius of 1000 m, where BS_1 is located at (0, 0, 0) and BS_2 is located at $(10,000,\ 0,\ 0)$ using three-dimensional Cartesian coordinates with the unit of meter. From a practical implementational perspective, the location of the LEOS edge is assumed to be semi-static in a fixed time frame, which is located at $(5,000,\ 0,\ 20,0000)$. The operating frequency at BSs and the LEOS is 5 GHz and 24 GHz, respectively. Lastly, when the number of ground users in each cell changes, the setting of the latency threshold, T_{ij}^{max}, will change accordingly. Refer to the setting of computing and communication parameters in STINs [5], [20], [41], [43], [46], the main parameters in our system are set as in Table I.

2) Benchmarks: We consider the following benchmarks for our proposed scheme.

- Pure local computing (PLC): All the ground users process their tasks locally without offloading, i.e., $l_{ij} = 0, \forall i \in \mathcal{I},\ \forall j \in \mathcal{J}$.
- Full offloading to the LEOS edge (FOL): All the ground users offload their tasks to the LEOS edge for remote computing, i.e., $l_{ij} = L_{ij}, u_{ij} = 1, \forall i \in \mathcal{I},\ \forall j \in \mathcal{J}$.
- Full offloading to the BS server (FOB): All the ground users offload their tasks to the connected BS server for remote computing, i.e., $l_{ij} = L_{ij}, u_{ij} = 0, \forall i \in \mathcal{I},\ \forall j \in \mathcal{J}$.

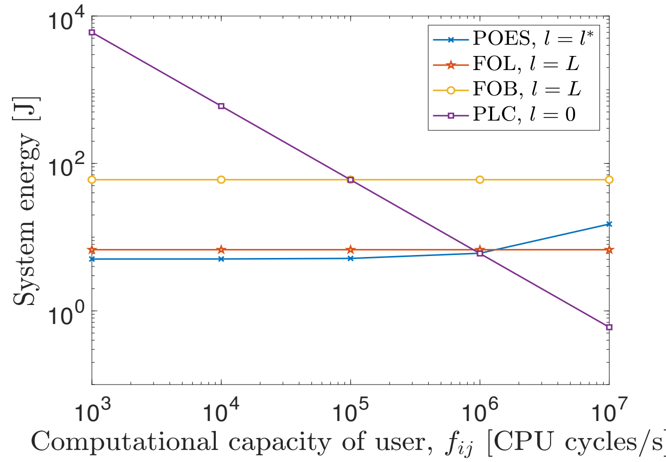


Fig. 4: System energy vs. the computational capability of users.

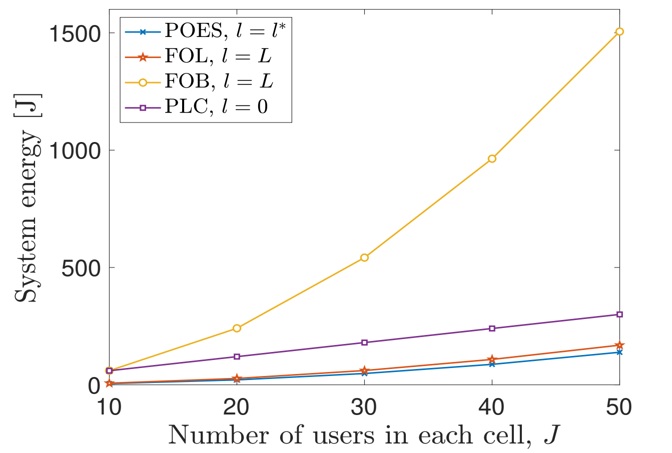


Fig. 5: System energy vs. the number of users in each cell.

In contrast to the above three benchmarks, our proposed scheme, partial offloading with edge selection (POES), is evaluated, where all the ground users offload their partial tasks either to the connected BS server or to the LEOS edge when local computing cannot fully meet the requirements, i.e., $l_{ij} = l_{ij}^*, l_{ij}^* \in [0, L_{ij}], u_{ij} \in \{0, 1\}, \forall i \in \mathcal{I},\ \forall j \in \mathcal{J}$, where l^* is the optimal number of bits to be offloaded.

B. Performance Evaluation

We evaluate the proposed scheme in terms of its energy dissipation, system delay, and impact of T_{max}. Note that these results are evaluated based on the proposed AO approach, which is usually suboptimal. Compared to the deep learning-enabled approach [1] that can usually obtain the optimal solutions [31], the accuracy gap of binary selection is about 5%, and the mean square error (MSE) gap of partial offloading is about 0.8%.

[1]The deep learning-enabled approach has to retrain once the network changes. By contrast, the proposed AO approach can be performed in near-real-time that can fit the change of STINs.

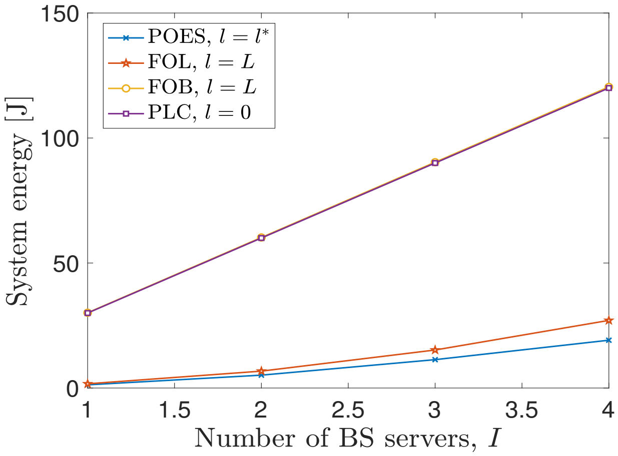


Fig. 6: System energy vs. the number of BS servers.

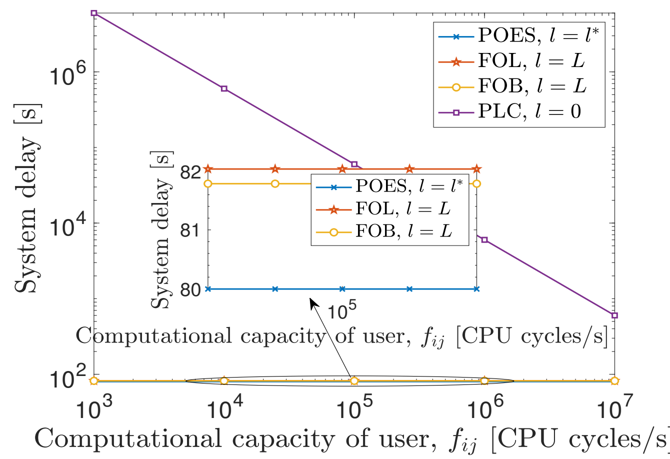


Fig. 7: System delay vs. the computational capability of users.

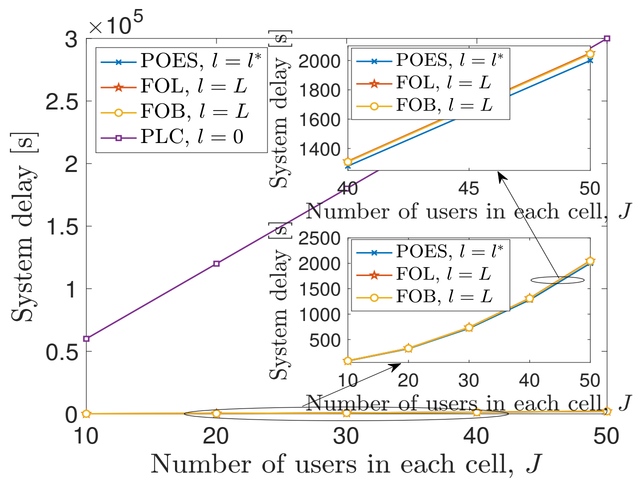


Fig. 8: System delay vs. the number of users in each cell.

1) System Energy: Figure 4 evaluates the energy consumption as the computational capability of users increases. Firstly, it is observed that the energy consumption of the PLC scheme decreases significantly as the computational capability of users increases. By contrast, the energy consumption of all other schemes is almost unchanged. This is because a high-efficiency local computational capability will substantially reduce the processing time of tasks. However, given the full offloading of the FOB and FOL schemes, the system's energy consumption of both will remain unaffected by the computational capability of users. The energy consumption of the POES scheme is better than that of the PLC, the FOB and FOL schemes since the local computational capability of users is low, e.g., $f_{ij}^{l} \leq 10^6$ CPU cycles/s. In general, the more tasks are offloaded, the less energy is consumed because the substantial computational capabilities of the LEOS edge and that of the BS servers are beneficially exploited.

Figure 5 evaluates the energy consumption as the number of users in each cell increases. Firstly, it is observed that the energy consumption of all the schemes increases with the number of users in each cell, especially that of the FOB scheme. This is because as the number of users increases, more tasks need to be processed and thus more energy consumption is involved. Furthermore, the energy consumption of the POES scheme is the lowest, the FOL scheme is better than the PLC scheme, and the FOB scheme is the worst. This is because with the number of users in each cell increasing, the BS server has to spend more time on task processing, which will result in significantly increased energy consumption for the FOB scheme. Compared to the FOB scheme, the energy consumption of the FOL and POES schemes increases slowly with the total number of users, because the LEOS edge has a higher computational capability to process more tasks without incurring more time.

Figure 6 evaluates the energy consumption as the number of BS servers increases. Firstly, it is observed that the energy consumption of all the schemes increases with the number of BS servers, especially that of the FOB and PLC schemes. As the number of BS servers increases, the total number of users increases accordingly, while the number of users served by each BS server remains unchanged. In this situation, higher energy consumption will be incurred by each scheme, since more users are served. Furthermore, the energy consumption of the proposed POES scheme is the lowest, that of the FOL scheme is the second-best, while the FOB and PLC schemes are the worst. This is because with the total number of users increasing, the energy consumption of the POES and FOL schemes increases slowly since the LEOS edge has a higher computational capability for processing more tasks without incurring much time cost. In contrast to POES and FOL schemes, the FOB and PLC schemes show a linear increase because the task processing time of both users and BS servers remains unchanged, when the number of BS servers is increased.

2) System Delay: Figure 7 evaluates the system delay as the computational capability of users increases. It is observed that only the system delay of the PLC scheme decreases rapidly as the computational capability of users increases, while the system delay of the other three schemes remains unchanged. The reason for this trend is similar to that mentioned in Fig. 4. Moreover, we observe that the system delay of the POES

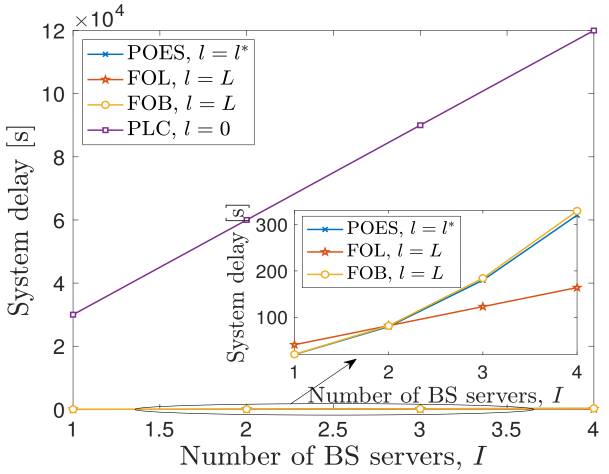


Fig. 9: System delay vs. the number of BS servers.

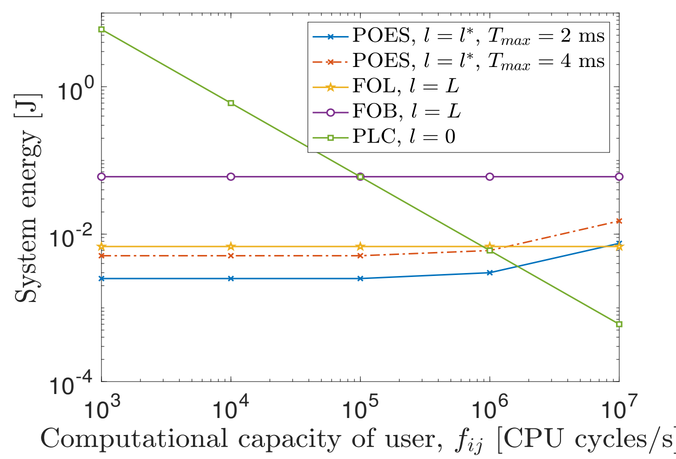


Fig. 10: System energy vs. the computational capability of users.

scheme is the lowest when the local computational capability of users increases from 10^3 CPU cycles/s to 10^7 CPU cycles/s. In contrast to the FOB and FOL schemes, in the POES scheme, the partial tasks can be offloaded either to the LEOS edge or to the BS server for processing remotely, while the remaining tasks can also be processed in parallel locally.

Figure 8 evaluates the system delay as the number of users in each cell increases. It is observed that the system delay of the PLC scheme increases significantly with the number of users in each cell increasing. By contrast, the system delay of the other schemes remains low and increases gracefully. This delay escalation is observed for the PLC scheme, because as the number of users in each cell increases, the PLC scheme requires more time to process the proliferation tasks of more users. In this case, the system delay will escalate when the computational capability of users is low. In contrast to the PLC scheme, the system delay of the FOB, the FOL and POES schemes remains relatively low because the ground users offload their tasks either to the BS or to the LEOS edge.

Figure 9 evaluates the system delay as the number of BS servers increases, exhibiting a similar trend to Fig. 8 as the total number of users increases. In contrast to the results of Fig. 8, the fixed number of users supported by each BS server will lead to a different delay trend for the FOB, FOL and POES schemes. In particular, the delay of the FOL and POES schemes is lower than that of the FOB scheme when $I \leq 2$. But the FOB scheme has the edge as the number of BS servers increases. This result is indeed expected since the LEOS edge has a higher computational capability than each BS server.

3) Impact of T_{max}: In practice, different kinds of applications have different tolerable delays. In our proposed POES scheme, T_{max} indicates the tolerable delay of the served traffic. A smaller value of T_{max} indicates the lower delay constraint, i.e., the served traffic is more delay-sensitive. The system delay and system energy versus the user's computational capacity is shown in Fig. 10 and Fig. 11, respectively. It is observed that T_{max} has a significant impact on the system performance. Specifically, we observe from Fig. 10 that the proposed POES scheme is better than the other benchmark

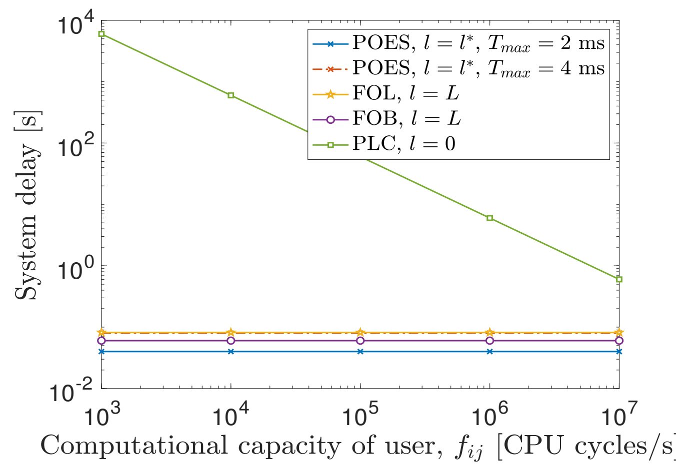


Fig. 11: System delay vs. the computational capability of user.

schemes in terms of the system energy. This is because the target of our proposed scheme is to minimize the system energy. In this context, the strategy with the lowest energy involved will be selected. In the contrast, Fig. 11 shows that the proposed POES scheme with $T_{max} = 2$ ms has the lowest delay but the POES scheme with $T_{max} = 4$ ms is not as good enough. The reason for this trend is that the system delay is considered a constraint in our formulated problem. In this case, if only the delay of each user does not exceed the delay constraint, an extra effort will be used for optimizing the system energy in the proposed POES scheme.

VI. Conclusions

An optimal offloading framework has been designed for LEOS edge-assisted multi-layer MEC systems. More explicitly, an optimal offloading access scheme and policy have been conceived for improving the computational performance of STINs in terms of computing latency, energy efficiency and coverage. We have solved the joint optimization problem of communication and computing resource allocation constructed for minimizing the system's energy dissipation while satisfying the computing latency requirement of users. To solve this optimization problem effectively, the original problem has been decomposed into the joint problem of offloading volume and mode as well as the joint optimization problem of computing resource allocation of the BS servers and the LEOS edge. Finally, our numerical results have characterized our multi-layer MEC framework in terms of its system energy and latency, demonstrating its improved energy efficiency and overall potential. We foresee some open issues when more LEOSs are involved, for example, the association between BSs and LEOSs in the multi-layer MEC system.

APPENDIX A
PROOF OF PROPOSITION 1

As seen in (13), we have $T_{ij} = \max\Big\{\frac{(L_{ij}-l_{ij})c_{ij}}{f_{ij}^{l}}, u_{ij}\Big(\frac{l_{ij}}{R_{ij}^{BS_i}}+\frac{l_{ij}c_{ij}}{f_{ij}^{BS_i}}\Big)+(1-u_{ij})\Big(\frac{l_{ij}}{R_{ij}^{LEO}}+\frac{l_{ij}c_{ij}}{f_{ij}^{LEO}}\Big)\Big\}$, yielding

$$\frac{(L_{ij}-l_{ij})\,c_{ij}}{f_{ij}^{l}} \le T_{ij}, \tag{48}$$

and

$$u_{ij}\left(\frac{l_{ij}}{R_{ij}^{BS_i}}+\frac{l_{ij}c_{ij}}{f_{ij}^{BS_i}}\right)+(1-u_{ij})\left(\frac{l_{ij}}{R_{ij}^{LEO}}+\frac{l_{ij}c_{ij}}{f_{ij}^{LEO}}\right)\le T_{ij}, \tag{49}$$

where $u_{ij}\in\{0,1\}$.

Thus, constraint (22a) is equivalent to

$$\begin{cases}\frac{(L_{ij}-l_{ij})c_{ij}}{f_{ij}^{l}}\le T_{ij}^{max},\\ u_{ij}\left(\frac{l_{ij}}{R_{ij}^{BS_i}}+\frac{l_{ij}c_{ij}}{f_{ij}^{BS_i}}\right)\le T_{ij}^{max},\\ (1-u_{ij})\left(\frac{l_{ij}}{R_{ij}^{LEO}}+\frac{l_{ij}c_{ij}}{f_{ij}^{LEO}}\right)\le T_{ij}^{max}.\end{cases} \tag{50}$$

Based on the above transformations, the non-linear constraint in (24) may be transformed to linear constrains imposed on u_{ij} and l_{ij}.

APPENDIX B
PROOF OF PROPOSITION 2

Let $\Lambda_{ij}^{E_L} = l_{ij}\Big(\frac{P_{ij}^{LEO}}{R_{ij}^{LEO}}+\frac{P_{LEO}c_{ij}}{f_{ij}^{LEO}}\Big)$, $\Lambda_{ij}^{U_L} = \frac{p_{ij}l_{ij}}{R_{ij}^{BS_i}}$, $\Lambda_{ij}^{T_L} = l_{ij}\Big(\frac{1}{R_{ij}^{LEO}}+\frac{c_{ij}}{f_{ij}^{LEO}}\Big)$, $\Lambda_{ij}^{E_B} = l_{ij}\Big(\frac{P_{ij}^{BS_i}}{R_{ij}^{BS_i}}+\frac{P_{BS_i}c_{ij}}{f_{ij}^{BS_i}}\Big)$, $\Lambda_{ij}^{U_B} = \frac{p_{ij}l_{ij}}{R_{ij}^{LEO}}$, and $\Lambda_{ij}^{T_B} = l_{ij}\Big(\frac{1}{R_{ij}^{LEO}}+\frac{c_{ij}}{f_{ij}^{LEO}}\Big)$. Problem $\mathcal{P}3a$ in (26) may then be transformed into an equivalent Quadratically Constrained Quadratic Program (QPQC) formulated as:

$$\min_{\mathbf{W}} \sum_{i=1}^{I}\sum_{j=1}^{J}\mathbf{b}_0^{\mathrm{T}}\mathbf{q} \tag{51}$$

$$\begin{aligned}
\text{s.t.}\quad & \mathbf{b}_1^{\mathrm{T}}\mathbf{q}\le f_{max}^{BS_i}, && (51a)\\
& \mathbf{b}_2^{\mathrm{T}}\mathbf{q}\le J, && (51b)\\
& \mathbf{b}_3^{\mathrm{T}}\mathbf{q}\le 0, && (51c)\\
& \mathbf{b}_4^{\mathrm{T}}\mathbf{q}\le E_{max}^{BS_i}, && (51d)\\
& \mathbf{b}_5^{\mathrm{T}}\mathbf{q}\le E_{max}^{LEO}-\sum_{i=1}^{I}\sum_{j=1}^{J}\Lambda_{ij}^{E_L}, && (51e)\\
& \mathbf{b}_6^{\mathrm{T}}\mathbf{q}\le T_{ij}^{max}, && (51f)\\
& \mathbf{b}_7^{\mathrm{T}}\mathbf{q}\le T_{ij}^{max}-\Lambda_{ij}^{E_T}, && (51g)\\
& \mathbf{q}^{\mathrm{T}}\mathrm{diag}(\mathbf{1}_{1\times IJ})\mathbf{q}-\mathbf{1}_{1\times IJ}^{\mathrm{T}}\mathbf{q}=0, && (51h)\\
& \mathbf{q}\ge 0, && (51i)
\end{aligned}$$

where $\mathbf{b}_0$ to $\mathbf{b}_7$ are as follows

$$\begin{cases}
\mathbf{b}_0=[\Lambda_{11}^{E_B}+\Lambda_{11}^{U_B}-\Lambda_{11}^{E_L}-\Lambda_{11}^{U_L},\ldots,\Lambda_{IJ}^{E_B}+\Lambda_{IJ}^{U_B}-\Lambda_{IJ}^{E_L}-\Lambda_{JJ}^{U_L}]^{\mathrm{T}},\\
\mathbf{b}_1=[\mathbf{0}_{1\times J},\ldots(\sum_{j=1}^{J}f_{ij}^{BS_i})\mathbf{1}_{1\times J}^{i},\ldots,\mathbf{0}_{1\times J}]^{\mathrm{T}},\\
\mathbf{b}_2=[\mathbf{0}_{1\times J},\ldots,\mathbf{1}_{1\times J}^{i},\ldots,\mathbf{0}_{1\times J}]^{\mathrm{T}},\\
\mathbf{b}_3=[-\mathbf{1}_{1\times IJ}]^{\mathrm{T}},\\
\mathbf{b}_4=[\mathbf{0}_{1\times J},\ldots,(\sum_{j=1}^{J}\Lambda_{ij}^{E_B})\mathbf{1}_{1\times J}^{i},\ldots,\mathbf{0}_{1\times J}]^{\mathrm{T}},\\
\mathbf{b}_5=[-\sum_{i=1}^{I}\sum_{j=1}^{J}\Lambda_{11}^{E_L},\ldots,-\sum_{i=1}^{I}\sum_{j=1}^{J}\Lambda_{IJ}^{E_L}]^{\mathrm{T}},\\
\mathbf{b}_6=[\Lambda_{11}^{T_B},\ldots\Lambda_{ij}^{T_B},\ldots,\Lambda_{IJ}^{T_B}]^{\mathrm{T}},\\
\mathbf{b}_7=[-\Lambda_{11}^{T_L},\ldots-\Lambda_{ij}^{T_L},\ldots,-\Lambda_{IJ}^{T_L}]^{\mathrm{T}}.
\end{cases} \tag{52}$$

We now define $\mathbf{W}\triangleq[q^{\mathrm{T}},1]^{\mathrm{T}}[q^{\mathrm{T}},1]$ and let $\mathrm{rank}(\mathbf{W}^*)=1$. By using the classic SDP approach, the above QPQC problem in (51) is transformed to

$$\overline{\mathcal{P}}3a:\min_{\mathbf{W}} \sum_{i=1}^{I}\sum_{j=1}^{J}\mathrm{Tr}(\mathbf{A}_0\mathbf{W}) \tag{53}$$

$$\begin{aligned}
\text{s.t.}\quad & \mathrm{Tr}(\mathbf{A}_1\mathbf{W})\le f_{max}^{BS_i}, && (53a)\\
& \mathrm{Tr}(\mathbf{A}_2\mathbf{W})\le J, && (53b)\\
& \mathrm{Tr}(\mathbf{A}_3\mathbf{W})\le 0, && (53c)\\
& \mathrm{Tr}(\mathbf{A}_4\mathbf{W})\le E_{max}^{BS_i}, && (53d)\\
& \mathrm{Tr}(\mathbf{A}_5\mathbf{W})\le E_{max}^{LEO}-\sum_{i=1}^{I}\sum_{j=1}^{J}\Lambda_{ij}^{E_L}, && (53e)\\
& \mathrm{Tr}(\mathbf{A}_6\mathbf{W})\le T_{ij}^{max}, && (53f)\\
& \mathrm{Tr}(\mathbf{A}_7\mathbf{W})\le T_{ij}^{max}-\Lambda_{ij}^{T_L}, && (53g)\\
& \mathrm{Tr}(\mathbf{A}_8\mathbf{W})=0, && (53h)\\
& \mathbf{W}(IJ,IJ)=1, && (53i)\\
& \mathbf{W}\ge 0, && (53j)
\end{aligned}$$

where $\mathbf{A}_0$ to $\mathbf{A}_8$ are as follows

$$\begin{aligned}
&\mathbf{A}_0 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_0 \\ \frac{1}{2}\mathbf{b}_0 & 0 \end{bmatrix}, \qquad &&\mathbf{A}_1 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_1 \\ \frac{1}{2}\mathbf{b}_1 & 0 \end{bmatrix}, \\
&\mathbf{A}_2 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_2 \\ \frac{1}{2}\mathbf{b}_2 & 0 \end{bmatrix}, \qquad &&\mathbf{A}_3 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_3 \\ \frac{1}{2}\mathbf{b}_3 & 0 \end{bmatrix}, \\
&\mathbf{A}_4 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_4 \\ \frac{1}{2}\mathbf{b}_4 & 0 \end{bmatrix}, \qquad &&\mathbf{A}_5 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_5 \\ \frac{1}{2}\mathbf{b}_5 & 0 \end{bmatrix}, \\
&\mathbf{A}_6 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_6 \\ \frac{1}{2}\mathbf{b}_6 & 0 \end{bmatrix}, \qquad &&\mathbf{A}_7 = \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{b}_7 \\ \frac{1}{2}\mathbf{b}_7 & 0 \end{bmatrix}, \\
&\mathbf{A}_8 = \begin{bmatrix} \mathrm{diag}(\mathbf{1}_{1\times IJ}) & -\frac{1}{2}\mathbf{1}_{1\times IJ} \\ -\frac{1}{2}\mathbf{1}_{1\times IJ}^{\mathrm{T}} & 0 \end{bmatrix}. &&
\end{aligned} \tag{54}$$

Appendix C
Proof of proposition 3

The objective and constraint functions of Problem $\mathcal{P}4a$ in (30) be denoted by

$$g(f_{ij}^{BS_i}) = \sum_{i=1}^{I}\sum_{j=1}^{J} u_{ij}\left(\frac{P_{ij}^{BS_i} l_{ij}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} l_{ij} c_{ij}}{f_{ij}^{BS_i}}\right), \tag{55}$$

and

$$\begin{cases} g_0(f_{ij}^{BS_i}) = \sum_{j=1}^{J} f_{ij}^{BS_i} - f_{max}^{BS_i}, \\ g_1(f_{ij}^{BS_i}) = \sum_{j=1}^{J} u_{ij} l_{ij}\left(\frac{P_{ij}^{BS_i}}{R_{ij}^{BS_i}} + \frac{P_{BS_i} c_{ij}}{f_{ij}^{BS_i}}\right) - E_{max}^{BS_i}, \\ g_2(f_{ij}^{BS_i}) = u_{ij} l_{ij}\left(\frac{1}{R_{ij}^{BS_i}} + \frac{c_{ij}}{f_{ij}^{BS_i}}\right) - T_{ij}^{max}. \end{cases} \tag{56}$$

Then, the second derivative of $g(f_{ij}^{BS_i}), g_2(f_{ij}^{BS_i})$ and $g_3(f_{ij}^{BS_i})$ with respect to $f_{ij}^{BS_i}$ is formulated as

$$\frac{\mathrm{d}^2 g(f_{ij}^{BS_i})}{\mathrm{d}{f_{ij}^{BS_i}}^2} = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{2u_{ij} P_{BS_i} l_{ij} c_{ij}}{{f_{ij}^{BS_i}}^3}, \tag{57}$$

$$\frac{\mathrm{d}^2 g_1(f_{ij}^{BS_i})}{\mathrm{d}{f_{ij}^{BS_i}}^2} = \sum_{j=1}^{J} \frac{2u_{ij} P_{BS_i} l_{ij} c_{ij}}{{f_{ij}^{BS_i}}^3}, \tag{58}$$

$$\frac{\mathrm{d}^2 g_2(f_{ij}^{BS_i})}{\mathrm{d}{f_{ij}^{BS_i}}^2} = \frac{2u_{ij} l_{ij} c_{ij}}{{f_{ij}^{BS_i}}^3}. \tag{59}$$

Since the values of P_{BS_i} and c_{ij} are positive and $u_{ij} \in \{0,1\}, l_{ij} \in [0, L_{ij}]$ and $f_{ij} \geq 0$, we have $\frac{\mathrm{d}^2 g(f_{ij}^{BS_i})}{\mathrm{d}{f_{ij}^{BS_i}}^2} \geq 0$, $\frac{\mathrm{d}^2 g_1(f_{ij}^{BS_i})}{\mathrm{d}{f_{ij}^{BS_i}}^2} \geq 0$ and $\frac{\mathrm{d}^2 g_2(f_{ij}^{BS_i})}{\mathrm{d}{f_{ij}^{BS_i}}^2} \geq 0$. Hence, the objective and constraint functions, namely $g(f_{ij}^{BS_i})$, $g_1(f_{ij}^{BS_i})$ and $g_2(f_{ij}^{BS_i})$ are convex functions with respect to $f_{ij}^{BS_i}$, respectively. Moreover, the constraint $g_0(f_{ij}^{BS_i})$ is a linear function. Problem $\mathcal{P}4a$ in (30) is therefore a strictly convex problem.

Appendix D
Proof of proposition 4

Let the objective and constraint functions of Problem $\mathcal{P}4b$ in (39) be denoted by

$$h(f_{ij}^{BS_i}) = \sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij})\left(\frac{P_{ij}^{LEO} l_{ij}}{R_{ij}^{LEO}} + \frac{P_{LEO} l_{ij} c_{ij}}{f_{ij}^{LEO}}\right), \tag{60}$$

and

$$\begin{cases} h_0(f_{ij}^{LEO}) = \sum_{i=1}^{I}\sum_{j=1}^{J} f_{ij}^{LEO} - f_{max}^{LEO}, \\ h_1(f_{ij}^{LEO}) = \sum_{i=1}^{I}\sum_{j=1}^{J}(1-u_{ij}) l_{ij}\left(\frac{P_{ij}^{LEO}}{R_{ij}^{LEO}} + \frac{P_{LEO} c_{ij}}{f_{ij}^{LEO}}\right) \\ \qquad\qquad -E_{max}^{LEO}, \\ h_2(f_{ij}^{LEO}) = (1-u_{ij}) l_{ij}\left(\frac{1}{R_{ij}^{LEO}} + \frac{c_{ij}}{f_{ij}^{LEO}}\right) - T_{ij}^{max}. \end{cases} \tag{61}$$

Then, the second derivative of $h(f_{ij}^{LEO}), h_2(f_{ij}^{LEO})$ and $h_3(f_{ij}^{LEO})$ with respect to f_{ij}^{LEO} is formulated as

$$\frac{\mathrm{d}^2 h(f_{ij}^{LEO})}{\mathrm{d}{f_{ij}^{LEO}}^2} = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{2(1-u_{ij}) P_{LEO} l_{ij} c_{ij}}{{f_{ij}^{LEO}}^3}, \tag{62}$$

$$\frac{\mathrm{d}^2 h_1(f_{ij}^{LEO})}{\mathrm{d}{f_{ij}^{LEO}}^2} = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{2(1-u_{ij}) P_{LEO} l_{ij} c_{ij}}{{f_{ij}^{LEO}}^3}, \tag{63}$$

$$\frac{\mathrm{d}^2 h_2(f_{ij}^{LEO})}{\mathrm{d}{f_{ij}^{LEO}}^2} = \frac{2(1-u_{ij}) l_{ij} c_{ij}}{{f_{ij}^{LEO}}^3}. \tag{64}$$

Since the values of P_{LEO} and c_{ij} are positive and $u_{ij} \in \{0,1\}, l_{ij} \in [0, L_{ij}]$ and $f_{ij} \geq 0$, we have $\frac{\mathrm{d}^2 h(f_{ij}^{LEO})}{\mathrm{d}{f_{ij}^{LEO}}^2} \geq 0$, $\frac{\mathrm{d}^2 h_1(f_{ij}^{LEO})}{\mathrm{d}{f_{ij}^{LEO}}^2} \geq 0$ and $\frac{\mathrm{d}^2 h_2(f_{ij}^{LEO})}{\mathrm{d}{f_{ij}^{LEO}}^2} \geq 0$. Hence, the objective and constraint functions, namely $h(f_{ij}^{LEO})$, $h_1(f_{ij}^{LEO})$ and $h_2(f_{ij}^{LEO})$ are convex function with respect to f_{ij}^{LEO}, respectively. Moreover, the constraint $h_0(f_{ij}^{LEO})$ is a linear function. Problem $\mathcal{P}4b$ in (39) is therefore a strictly convex problem.

References

[1] I. Del Portillo, B. G. Cameron, and E. F. Crawley, "A Technical Comparison of Three Low Earth Orbit Satellite Constellation Systems to Provide Gobal Broadband," *Acta Astronautica*, vol. 159, pp.123-135, Jun. 2019.

[2] T. Pfandzelter, J. Hasenburg, D. and Bermbach, "Towards a Computing Platform for the LEO Edge," in *Proc. ACM EdgeSys Workshop*, Edinburgh, UK, Apr. 2021, pp. 43-48.

[3] Z. Zhang, Y. Li, C. Huang, Q. Guo, L. Liu, C. Yuen, and Y. L. Guan, "User Activity Detection and Channel Estimation for Grant-Free Random Access in LEO Satellite-Enabled Internet-of-Things," *IEEE Internet of Things Journal*, vol. 7, Sep. 2020, pp. 8811-8825

[4] W. Saad, M. Bennis, and M. Chen, "A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems," *IEEE Network*, vol. 34, no. 3, pp. 134-142, May 2020.

[5] M. Giordani and M. Zorzi, "Non-Terrestrial Networks in the 6G Era: Challenges and Opportunities," *IEEE Network*, vol. 35, no. 2, pp. 244-251, Mar. 2021.

[6] N. U. L. Hassan, C. Huang, C. Yuen, A. Ahmad, and Y. Zhang, "Dense Small Satellite Networks for Modern Terrestrial Communication Systems: Benefits, Infrastructure, and Technologies," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 96-103, Oct. 2020.

[7] H. Yao, L. Wang, X. Wang, Z. Lu and Y. Liu, "The Space-Terrestrial Integrated Network: An Overview," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 178-185, Sept. 2018

[8] X. Cao, B. Yang, C. Yuen, and Z. Han, "HAP-Reserved Communications in Space-Air-Ground Integrated Networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 8286-8291, Aug. 2021.

[9] Z. Jia, Q. Wu, C. Dong, C. Yuen, and Z. Han, "Hierarchical Aerial Computing for Internet of Things via Cooperation of HAPs and UAVs," *IEEE Internet of Things Journal*, Feb. 2022.

[10] D. Bhattacherjee, S. Kassing, M. Licciardello, and A. Singla, "In-Orbit Computing: An Outlandish Thought Experiment?" in *Proc. ACM HotNets Workshop*, Nov. 2020, pp. 197–204.

[11] V. Bhosale, K. Bhardwaj, and A. Gavrilovska, "Toward Loosely Coupled Orchestration for the LEO Satellite Edge," in *Proc. USENIX HotEdge Workshop*, Jun. 2020.

[12] T. Pfandzelter and D. Bermbach, "Edge (of the Earth) Replication: Optimizing Content Delivery in Large LEO Satellite Communication Networks," in *Proc. IEEE/ACM CCGrid*, Melbourne, Australia, May 2021, pp. 565–575.

[13] D. Swinhoe, "Loft Orbital to Work with US Space Force on Satellite 'Edge Computing' Capabilities," May 2021. Online: https://www.datacenterdynamics.com/en/news/loft-orbital-to-work-with-us-space-force-on-satellite-edge-computing-capabilities/.

[14] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y. A. Zhang, "The Roadmap to 6G: AI Empowered Wireless Networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 84-90, Aug. 2019.

[15] B. Yang, X. Cao, K. Xiong, C. Yuen, Y. L. Guan, S. Leng, L. Qian, and Z. Han, "Edge Intelligence for Autonomous Driving in 6G Wireless System: Design Challenges and Solutions," *IEEE Wireless Communications*, vol. 28, no. 2, pp. 40-47, Apr. 2021.

[16] S. Venkatesh, et al., "A High-Speed Programmable and Scalable Terahertz Holographic Metasurface Based on Tiled CMOS Chips," *Nature Electronics,* vol. 3, pp. 785-793, Dec. 2020.

[17] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657-1681, May 2017.

[18] Y. Yang, K. Wang, G. Zhang, X. Chen, X. Luo, and M. -T. Zhou, "MEETS: Maximal Energy Efficient Task Scheduling in Homogeneous Fog Networks," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 4076-4087, Oct. 2018.

[19] K. Wang, Y. Zhou, Z. Liu, Z. Shao, X. Luo, and Y. Yang, "Online Task Scheduling and Resource Allocation for Intelligent NOMA-Based Industrial Internet of Things," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 803-815, May 2020.

[20] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint Computation Offloading and User Association in Multi-Task Mobile Edge Computing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 12, pp. 12313-12325, Dec. 2018.

[21] K. Wang, Y. Zhou, J. Li, L. Shi, W. Chen, and L. Hanzo, "Energy-Efficient Task Offloading in Massive MIMO-Aided Multi-Pair Fog-Computing Networks," *IEEE Transactions on Communications*, vol. 69, no. 4, pp. 2123-2137, Apr. 2021.

[22] C. Liaskos, A. Tsioliaridou, A. Pitsillides, S. Ioannidis, and I. F. Akyildiz, "Using any Surface to Realize a New Paradigm for Wireless Communications," *Communications of the ACM*, vol. 61, no. 11, pp. 30-33, Nov. 2018.

[23] M. Di Renzo *et al.*, "Smart Radio Environments Empowered by Reconfigurable AI Meta-surfaces: An Idea Whose Time Has Come," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 1-20, Dec. 2019.

[24] C. Huang, S. Hu, G. C. Alexandropoulos, A. Zappone, C. Yuen, R. Zhang, M. Di Renzo, and M. Debbah, "Holographic MIMO Surfaces for 6G Wireless Networks: Opportunities, Challenges, and Trends," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 118-125, Oct. 2020.

[25] X. Cao, B. Yang, C. Huang, C. Yuen, M. Di Renzo, D. Niyato, and Z. Han, "Reconfigurable Intelligent Surface-Assisted Aerial-Terrestrial Communications via Multi-Task Learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 10, pp. 3035-3050, Jun. 2021.

[26] X. Cao, B. Yang, C. Huang, C. Yuen, M. Di Renzo, Z. Han, D. Niyao, H. V. Poor, and L. Hanzo, "AI-Assisted MAC for Reconfigurable Intelligent Surface-Aided Wireless Networks: Challenges and Opportunities," *IEEE Communications Magazine*, vol. 59, no. 6, pp. 21-27, Jun. 2021.

[27] X. Cao B. Yang, C. Huang, G. C. Alexandropoulos, C. Yuen, Z. Han, H. V. Poor, and L. Hanzo, "Massive Access of Static and Mobile Users via Reconfigurable Intelligent Surfaces: Protocol Design and Performance Analysis," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 4, pp. 1253-1269, Apr. 2022.

[28] T. Bai, C. Pan, Y. Deng, M. Elkashlan, A. Nallanathan, and L. Hanzo, "Latency Minimization for Intelligent Reflecting Surface Aided Mobile Edge Computing," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2666-2682, Nov. 2020.

[29] X. Lai, L. Fan, X. Lei, Y. Deng, G. K. Karagiannidis, and A. Nallanathan, "Secure Mobile Edge Computing Networks in the Presence of Multiple Eavesdroppers," *IEEE Transactions on Communications*, vol. 70, no. 1, pp. 500-513, Jan. 2022.

[30] B. Yang, X. Cao, X. Li, Q. Zhang, and L. Qian, "Mobile-Edge-Computing-Based Hierarchical Machine Learning Tasks Distribution for IIoT," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2169-2180, Mar. 2020.

[31] B. Yang, X. Cao, J. Bassey, X. Li, and L. Qian, "Computation Offloading in Multi-Access Edge Computing: A Multi-Task Learning Approach," *IEEE Transactions on Mobile Computing*, vol. 20, no. 9, pp. 2745-2762, Sep. 2021.

[32] B. Yang, X. Cao, C. Yuen, and L. Qian, "Offloading Optimization in Edge Computing for Deep-Learning-Enabled Target Tracking by Internet of UAVs," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9878-9893, Jun. 2021.

[33] S. Tang, L. Chen, K. H. Xia, L. Fan, and A. Nallanathan, "Computational Intelligence and Deep Learning for Next-Generation Edge-Enabled Industrial IoT," arXiv preprint arXiv:2110.14937, 2021.

[34] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar and A. Y. Zomaya, "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457-7469, Aug. 2020.

[35] T. Zhang, Z. Shen, J. Jin, X. Zheng, A. Tagami and X. Cao, "Achieving Democracy in Edge Intelligence: A Fog-Based Collaborative Learning Scheme," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2751-2761, Feb. 2021.

[36] B. Yang, X. Cao, X. Li, C. Yuen, and L. Qian, "Lessons Learned from Accident of Autonomous Vehicle Testing: An Edge Learning-aided Offloading Framework," *IEEE Wireless Communications Letters*, vol. 9, pp. 1182-1186, Aug. 2020.

[37] W. J. Larson and J. R. Wertz, "Space Mission Analysis and Design," Technical report, Torrance, CA (United States), Microcosm, Inc. 1992.

[38] J. A. Fraire, G. Nies, H. Hermanns, K. Bay, and M. Bisgaard, "Battery-Aware Contact Plan Design for LEO Satellite Constellations: The Ulloriaq Case Study," in *Proc. IEEE GLOBECOM*, Abu Dhabi, UAE, Dec. 2018.

[39] B. Di, H. Zhang, L. Song, Y. Li, and G. Y. Li, "Ultra-Dense LEO: Integrating Terrestrial-Satellite Networks Into 5G and Beyond for Data Offloading," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 47-62, Jan. 2019.

[40] R. Deng, B. Di, S. Chen, S. Sun, and L. Song, "Ultra-Dense LEO Satellite Offloading for Terrestrial Networks: How Much to Pay the Satellite Operator?" *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6240-6254, Oct. 2020.

[41] N. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "Space/Aerial-Assisted Computing Offloading for IoT Applications: A Learning-Based Approach," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1117-1129, May 2019.

[42] J. Li, K. Xue, D. S. L. Wei, J. Liu, and Y. Zhang, "Energy Efficiency and Traffic Offloading Optimization in Integrated Satellite/Terrestrial Radio Access Networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2367-2381, Apr. 2020.

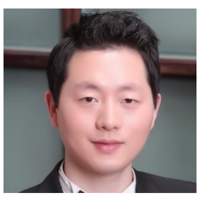
[43] Q. Tang, Z. Fei, B. Li, and Z. Han, "Computation Offloading in LEO Satellite Networks with Hybrid Cloud and Edge Computing," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9164-9176, Jun. 2021.

[44] X. Cao, B. Yang, C. Huang, C. Yuen, Y. Zhang, D. Niyato, and Z. Han, "Converged Reconfigurable Intelligent Surface and Mobile Edge Computing for Space Information Networks," *IEEE Network*, vol. 35, no. 4, pp. 42-48, Jul. 2021.

[45] D. B. Shmoys and E. Tardos, "An Approximation Algorithm for the Generalized Assignment Problem," Math. Program., vol. 62, no. 1–3, pp. 461–474, Feb. 1993.

[46] T. Bai, J. Wang, Y. Ren and L. Hanzo, "Energy-Efficient Computation Offloading for Secure UAV-Edge-Computing Systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 6074-6087, Jun. 2019.

Xuelin Cao (S'19-M'21) received her M.S. and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 2010 and 2019, respectively. From 2010 to 2012, she worked in NSN (Nokia Siemens Networks) as an engineer in the CDMA BTS Lab in Hangzhou, China. From Sep. 2017 to Jan. 2020, she was a visiting scholar with the Department of ECE at the University of Houston (UH), Texas, USA. She is currently with the School of Cyber Engineering, Xidian University, China, and also works as a Postdoctoral Fellow with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD), Singapore. Her research interests include MAC protocol design and modeling, 5G/6G, reconfigurable intelligent surface (RIS), and backscatter communications.



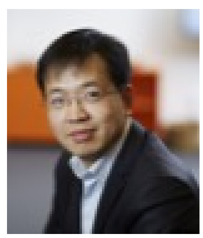
Bo Yang (M'19) received the B.E., M.S. and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 2007, 2010, and 2016, respectively.

From Oct. 2017 to Jan. 2020, he was a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, Prairie View A&M University (PVAMU), a member of the Texas A&M University System, Prairie View, TX, USA. From Feb. 2020 to Apr. 2022, He was a Research Fellow with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD), Singapore. He is currently a professor with the School of Computer Science, Northwestern Polytechnical University. His research focuses on machine learning, 5G/6G, multi-access/mobile edge computing, reconfigurable intelligent surface (RIS), and application of AI for wireless networks. He served as a session chair for IEEE VTC-Fall 2018 and a track chair for EAI IoTaas 2020.

Yulong Shen (IEEE Member) received the B.S. and M.S. degrees in computer science and the Ph.D. degree in cryptography from Xidian University, Xi'an, China, in 2002, 2005, and 2008, respectively. He is currently a Professor with the School of Computer Science and Technology, Xidian University, where he is also an Associate Director of the Shaanxi Key Laboratory of Network and System Security and a member of the State Key Laboratory of Integrated Services Networks. His research interests include wireless network security and cloud computing security. He has also served on the technical program committees of several international conferences, including ICEBE, INCoS, CIS, and SOWN.

Chau Yuen (S'04-M'06-SM'12-F'21) received the B.Eng. and Ph.D. degrees from Nanyang Technological University (NTU), Singapore, in 2000 and 2004, respectively. He was a Post-Doctoral Fellow with Lucent Technologies Bell Labs, Murray Hill, in 2005, and a Visiting Assistant Professor with The Hong Kong Polytechnic University in 2008. From 2006 to 2010, he was with the Institute for Infocomm Research (I2R), Singapore, where he was involved in an industrial project on developing an 802.11n Wireless LAN system and participated actively in 3Gpp long-term evolution (LTE) and LTE-Advanced (LTE-A) standardization. Since 2010, he has been with the Singapore University of Technology and Design.

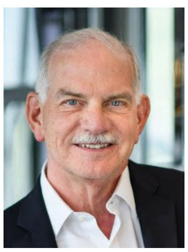
He was a recipient of the Lee Kuan Yew Gold Medal, the Institution of Electrical Engineers Book Prize, the Institute of Engineering of Singapore Gold Medal, the Merck Sharp and Dohme Gold Medal, and twice a recipient of the Hewlett Packard Prize. He received the IEEE Asia Pacific Outstanding Young Researcher Award in 2012 and IEEE VTS Singapore Chapter Outstanding Service Award on 2019. He serves as an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, and the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, where he was awarded as the Top Associate Editor from 2009 to 2015. He served as the guest editor for several special issues, including IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE COMMUNICATIONS MAGAZINE, IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING. He is a Distinguished Lecturer of IEEE Vehicular Technology Society.



Yan Zhang (IEEE Fellow'20) is currently a Full Professor with the Department of Informatics, University of Oslo, Norway. He received the Ph.D. degree from the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore. He received M.S. and B.S from Beihang University and Nanjing University of Post and Telecommunications, respectively. His research interests include next-generation wireless networks leading to 6G, green and secure cyber-physical systems (e.g., smart grid and transport). Dr. Zhang is an Editor (or Area Editor, Senior Editor, Associate Editor) for several IEEE transactions/magazine, including IEEE Network Magazine, IEEE Transactions on Network Science and Engineering, IEEE Transactions on Vehicular Technology, IEEE Transactions on Industrial Informatics, IEEE Transactions on Green Communications and Networking, IEEE Communications Survey and Tutorials, IEEE Internet of Things Journal, IEEE Systems Journal, IEEE Vehicular Technology Magazine, and IEEE Blockchain Technical Briefs. He is a symposium/track chair in a number of conferences, including IEEE ICC 2021, IEEE SmartGridComm 2021, and IEEE Globecom 2017. He is the Chair of IEEE Communications Society Technical Committee on Green Communications and Computing (TCGCC). He is an IEEE Communications Society Distinguished Lecturer and IEEE Vehicular Technology Society Distinguished Speaker. He was an IEEE Vehicular Technology Society Distinguished Lecturer during 2016-2020. He is CCF Senior Member, an elected member of CCF Technical Committee of Blockchain, and 2019 CCF Distinguished Speaker. Since 2018, Prof. Zhang was a recipient of the global "Highly Cited Researcher" Award (Web of Science top 1%most cited worldwide). He is Fellow of IEEE, Fellow of IET, elected member of Academia Europaea (MAE), elected member of the Royal Norwegian Society of Sciences and Letters (DKNVS), and elected member of Norwegian Academy of Technological Sciences (NTVA).

Zhu Han (S'01-M'04-SM'09-F'14) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively.

From 2000 to 2002, he was an R&D Engineer of JDSU, Germantown, Maryland. From 2003 to 2006, he was a Research Associate at the University of Maryland. From 2006 to 2008, he was an assistant professor at Boise State University, Idaho. Currently, he is a John and Rebecca Moores Professor in the Electrical and Computer Engineering Department as well as in the Computer Science Department at the University of Houston, Texas. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. Dr. Han received an NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the Journal on Advances in Signal Processing in 2015, IEEE Leonard G. Abraham Prize in the field of Communications Systems (best paper award in IEEE JSAC) in 2016, and several best paper awards in IEEE conferences. Dr. Han was an IEEE Communications Society Distinguished Lecturer from 2015-2018, AAAS fellow since 2019 and ACM distinguished Member since 2019. Dr. Han is 1% highly cited researcher since 2017 according to Web of Science. Dr. Han is also the winner of 2021 IEEE Kiyo Tomiyasu Award, for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: "for contributions to game theory and distributed management of autonomous communication networks."



H. Vincent Poor (S'72-M'77-SM'82-F'87) received the Ph.D. degree in EECS from Princeton University in 1977. From 1977 until 1990, he was on the faculty of the University of Illinois at Urbana-Champaign. Since 1990 he has been on the faculty at Princeton, where he is currently the Michael Henry Strater University Professor. During 2006 to 2016, he served as the dean of Princeton's School of Engineering and Applied Science. He has also held visiting appointments at several other universities, including most recently at Berkeley and Cambridge. His research interests are in the areas of information theory, machine learning and network science, and their applications in wireless networks, energy systems and related fields. Among his publications in these areas is the forthcoming book *Machine Learning and Wireless Communications*. (Cambridge University Press). Dr. Poor is a member of the National Academy of Engineering and the National Academy of Sciences and is a foreign member of the Chinese Academy of Sciences, the Royal Society, and other national and international academies. He received the IEEE Alexander Graham Bell Medal in 2017.

Lajos Hanzo (FIEEE'04) received his Master degree and Doctorate in 1976 and 1983, respectively from the Technical University (TU) of Budapest. He was also awarded the Doctor of Sciences (DSc) degree by the University of Southampton (2004) and Honorary Doctorates by the TU of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE Press. He has served several terms as Governor of both IEEE ComSoc and of VTS. He has published 2000+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 123 PhD students. Over 40 of them are Professors at various stages of their careers in academia and many of them are leading scientists in the wireless industry. He is also a Fellow of the Royal Academy of Engineering (FREng), of the IET and of EURASIP. He is the recipient of the 2022 Eric Sumner Field Award.