Joint Computation Offloading and Resource Allocation for Low-Earth Orbit Satellites MEC Networks

Meng Wang[1,2], Yaqiong Wang[1](✉), Cheng Zhang[1,2], Hui Zhou[1],
and Longteng Yi[1,2]

[1] Institute of Telecommunication and Navigation Satellites, China Academy of Space Technology, Beijing 100094, China
`wangmeng_1996@hit.edu.cn`, `wangyaqiong83@163.com`
[2] Innovation center for Satellite Communication Systems, Beijing 100094, China

Abstract. This study addresses the challenges of joint computation offloading and resource allocation in low-earth orbit (LEO) satellite networks. To effectively manage the computational demands of LEO satellites, we propose a collaborative framework that allows each LEO satellite to offload tasks either to high-earth orbit (GEO) satellites or to multi-access edge computing (MEC) servers on the ground. Our goal is to jointly optimize the offloading ratio, computational frequency, transmission power, and bandwidth utilization of the LEO satellites, aiming to minimize the overall energy consumption while adhering to latency requirements. We formulate this challenge as a non-convex optimization problem and introduce an energy-efficient layered optimization approach to address it with reduced complexity. This involves breaking down the original problem into several manageable subproblems, which are solved sequentially to achieve a suboptimal solution. The simulation results confirm the effectiveness of our method, demonstrating its advantages over existing benchmark algorithms.

Keywords: Computation offloading · Resource allocation · Low-orbit Satellite network · Multi-access edge computing

1 Introduction

Multi-access Edge Computing (MEC) has emerged as an effective framework in the architecture of next-generation networks, facilitating efficient, low-latency data processing at the network's edge. By shifting computation-heavy tasks to remote MEC servers equipped with adequate processing capabilities, MEC can significantly lower the energy consumption of mobile users (MUs) [1]. Within the MEC framework, two primary challenges arise: computation offloading and

---

This work is supported by the pre-research project on Civil Aerospace Technologies, grant number D010203.

H.-H. Chen and W. Meng (Eds.): WiSATS 2024, LNICST 605, pp. 150–160, 2025.
https://doi.org/10.1007/978-3-031-86196-3_13

resource allocation. Computation offloading involves deciding where mobile users should execute their tasks, while resource allocation focuses on how these tasks are managed to ensure compliance with Quality of Service (QoS) standards [2].

In traditional Mobile Edge Computing (MEC) setups, users are mainly ground-based intelligent terminals, with MEC servers located in nearby base stations. However, the rapid advancement of satellite communication, particularly with low-Earth orbit (LEO) satellite constellations, has broadened MEC application scenarios. LEO satellites, especially remote sensing ones, handle computational tasks like image processing and large-scale routing. With the end-to-end transmission delay between LEO satellites and ground stations reduced to 4–30 ms, LEO satellites can act as users, while geostationary Earth orbit (GEO) satellites with ample computational resources serve as MEC servers. Integrating MEC with LEO satellite networks can enhance their performance, particularly for computationally intensive tasks. Nonetheless, joint computation offloading and resource allocation in LEO satellite MEC networks remain unresolved challenges.

Several previous studies have explored this topic [4–10]. Jing *et al.* [4] examined a single LEO-assisted MEC system, aiming to minimize the combined energy consumption of users and the LEO-MEC server by optimizing computation offloading decisions, utilizing a federated learning approach. Cao *et al.* [5] built on this work, focusing on maximizing the weighted sum computation rate through the optimization of radio and computational resources in both binary and partial offloading scenarios. However, neither study addressed task execution constraints. Dong *et al.* [6] took communication quality into account, investigating the joint optimization of offloading strategies and resource allocation in LEO-enabled MEC systems, with the goal of minimizing user delay in each time slot. In contrast, Zhang *et al.* [7] aimed to enhance the energy efficiency (EE) of LEOs, formulating a non-convex EE maximization problem under users' QoE requirements, and optimizing LEO positions, user transmit power, and computational load iteratively. Unlike the aforementioned studies that considered only a single LEO-MEC server, Minglei *et al.* [8] analyzed a multi-LEO framework, minimizing the overall power consumption of LEOs and users by jointly optimizing user associations, power control, computation capacity, and GEO positions. While the previous studies focused solely on LEOs as MEC servers, [9] and [10] introduced scenarios in which a BS-MEC server collaborates with LEO-MEC servers for data offloading. Specifically, Rodrigues *et al.* [9] explored a tethered LEO-assisted MEC system where tasks could be offloaded to both LEO-MEC and BS-MEC servers simultaneously, aiming to minimize the weighted-sum system delay of mobile users by optimizing LEO positions, time slot allocations, and task splitting ratios. Hao *et al.* [10] investigated a collaborative multi-LEO MEC system, minimizing the total execution latency for all users by jointly optimizing offloading decisions alongside communication and computing resources.

Unlike the previous studies, this paper introduces a novel MEC scenario where LEO satellites function as users, while GEO satellites and ground gateway stations serve as MEC servers. In this setup, LEOs operate with partial offloading, allowing them to either execute tasks locally or offload them to MEC servers. We jointly optimize the offloading ratio, transmit power, transmission bandwidth, and computational frequency to minimize energy consumption while meeting latency constraints. This leads to a non-convex optimization problem, which we address by decomposing it into several subproblems, solving each sequentially. Simulation results demonstrate that our approach significantly reduces energy consumption compared to benchmark algorithms.

2 System Model and Problem Formulation

We investigate a satellite-terrestrial cooperative MEC system comprising one BS-MEC server, M GEO-MEC server and N LEOs as depicted in Fig. 1. The set of LEOs is defined as $\mathcal{N} \triangleq \{1, \cdots, N\}$, while the MEC servers are defined as $\mathcal{M} \triangleq \{0, \cdots, M\}$, where the index '0' represents the BS-MEC server. To model dynamic characteristics, we consider a time-slotted system of T time slots, each of equal length ΔT, denoted as $\mathcal{T} \triangleq \{1, \cdots, T\}$. In this paper, we assume that LEOs hover at a fixed height H above the ground. To represent the positions of LEOs and GEO-MEC servers, we utilize a three-dimensional coordinate system. The BS-MEC server is positioned at $(X_0, Y_0, 0)$, while the coordinates of the j-th GEO-MEC server and the i-th LEO in the t-th time slot are denoted as $\mathbf{q}_j(t) = (X_j(t), Y_j(t), Z_j(t)), \forall j \in \mathcal{M}$ and $\mathbf{q}_i(t) = (x_i(t), y_i(t), z_i(t)), \forall i \in \mathcal{N}$, respectively. Following the approach in [10], we assume that the locations of LEOs and GEOs remain fixed within a time slot, but may vary between different time slots. In our model, the k-th task of the i-th LEO in the t-th time slot is characterized as $(S_{i,k}(t), C_{i,k}(t), T_{i,k}(t))$, where $S_{\min} \le S_{i,k}(t) \le S_{\max}, C_{\min} \le C_{i,k}(t) \le C_{\max}, T_{\min} \le T_{i,k}(t) \le T_{\max}$ denote the data size (bits), the number of CPU cycles required for executing a single bit and the maximum tolerable delay (ms) with respect to the k-th task of the i-th LEO, respectively. LEOs can partially offload tasks to the BS-MEC server and GEO-MEC servers via a wireless uplink channel in a partial offloading manner [4,6]. The offloading ratio of all LEOs in the t-th time slot are denoted as $\beta_{k,j,i}(t), k \in [1, A_i(t)], j \in \mathcal{M}, i \in \mathcal{N}$, where $0 \le \beta_{k,j,i}(t) \le 1$ indicates the k-th task's offloading ratio to the j-th MEC server in the t-th time slot.

To streamline the analysis, we focus on a single-input single-output (SISO) channel between the LEOs and the MEC servers. Additionally, we assume that different LEOs access the MEC servers using an orthogonal multiple access (OMA) scheme. The uplink channel capacity between the i-th LEO and the BS-MEC server in t-th time slot is given by

$$R_{i,g}(t) = \alpha_{0,i}(t) B_g \log\left(1 + \frac{|h_{i,g}(t)|^2 p_{i,0}(t)}{N_0 B_g \alpha_{0,i}(t)}\right), \tag{1}$$

where B_g is the total bandwidth allocated between LEOs and BS, N_0 is the white noise power spectrum density, $0 \le \alpha_{j,i}(t) \le 1, \forall j \in \mathcal{M}, \forall i \in \mathcal{N}$ is the ratio of the j-th MEC server's bandwidth allocated to the i-th LEO in the t-th time

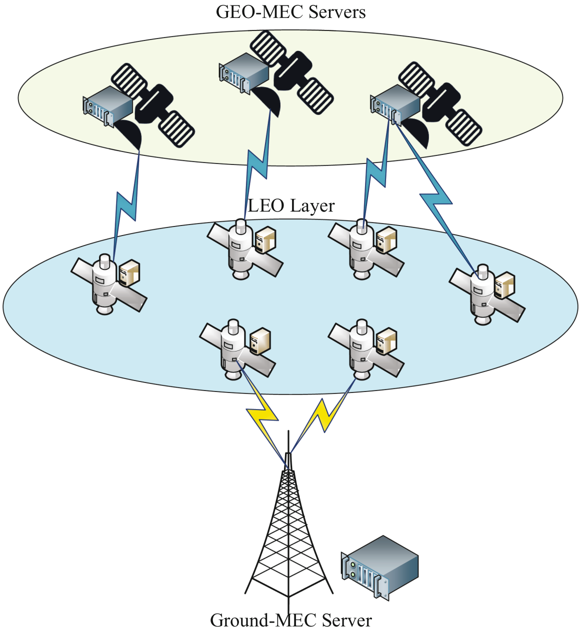

Fig. 1. An illustration of the considered satellite-terrestrial collaborative Networks.

slot, $0 \leq p_{i,j}(t) \leq p_{\max}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}$ is the transmit power of the i-th LEO to the j-th MEC server. The channel gain between the i-th LEO and the BS-MEC server in the t-th time slot is denoted as $h_{i,g}(t)$. Thus, the uplink offloading time for the k-th task of the i-th LEO in the t-th time slot is expressed as

$$T_{k,0,i}(t) = \frac{\beta_{k,0,i}(t) S_{i,k}(t)}{R_{i,g}(t)}, \forall k \in [1, A_i(t)]. \tag{2}$$

The maximum total transmit energy consumption of the i-th LEO during the t-th time slot can be expressed as

$$E_i^r(t) = \Delta T \sum_{j=0}^{M} p_{i,j}(t). \tag{3}$$

The channel between i-th LEO and j-th GEO can be modeled as a free space loss channel. The uplink channel capacity between i-th LEO and j-th GEO can be expressed as

$$R_{i,j}(t) = \alpha_{j,i}(t) B_u \log\left(1 + \frac{|h_{i,j}(t)|^2 p_{i,j}(t)}{N_0 B_u \alpha_{j,i}(t)}\right), \forall j \in [1, M], \tag{4}$$

where B_u is the total bandwidth between each LEO and GEOs, $h_{i,j}(t)$ is the channel gain. Therefore, the uplink offloading time for k-th task of i-th LEO in

t-th time slot is given by

$$T_{k,j,i}(t) = \frac{\beta_{k,j,i}(t) S_{i,k}(t)}{R_{i,j}(t)}, \forall k \in [1, A_i(t)], \tag{5}$$

According to [11], during the t-th time slot, the computational energy consumed by the i-th LEO is expressed as

$$E_i^l(t) = \Delta T \epsilon f_i(t)^3, \tag{6}$$

where ϵ is the CPU computation coefficient, and $f_i(t)$ represents the computational frequency of i-th LEO in t-th time slot, which can be dynamically adjusted based on the task load [12]. Given that this paper primarily focuses on the energy consumption of LEOs, we do not consider the computational energy in MEC servers. The local computing time consumed by i-th LEO when executing k-th task is expressed as

$$T_{i,k}^l(t) = \frac{\left(1 - \sum\limits_{j=0}^{M} \beta_{k,j,i}(t)\right) S_{i,k}(t) C_{i,k}(t)}{f_i(t)} \tag{7}$$

We aim to minimize the total energy consumption of all LEOs while satisfying the latency constraints. Therefore, the optimization problem can be formulated as

$$(\mathbf{P1}): \min_{\{\mathbf{X}(t)\}} \sum_{t=1}^{T} \sum_{i=1}^{N} \left\{ E_i^l(t) + E_i^r(t) \right\}, \tag{8a}$$

$$\text{s.t. } 0 \leq f_i(t) \leq f_{\max}, \forall i \in \mathcal{N}, \forall t \in \mathcal{T} \tag{8b}$$

$$0 \leq \beta_{k,j,i}(t) \leq 1, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, k \in [1, A_i(t)], \tag{8c}$$

$$\sum_{j=0}^{M} \beta_{k,j,i}(t) \leq 1, t \in \mathcal{T}, \forall i \in \mathcal{N}, k \in [1, A_i(t)], \tag{8d}$$

$$0 \leq p_{i,j}(t) \leq p_{\max}, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \tag{8e}$$

$$0 \leq \alpha_{j,i}(t) \leq 1, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \tag{8f}$$

$$\sum_{i=1}^{N} \alpha_{j,i}(t) \leq 1, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \tag{8g}$$

$$\frac{\left(1 - \sum\limits_{j=0}^{M} \beta_{k,j,i}(t)\right) C_{i,k}(t) S_{i,k}(t)}{f_i(t)} \leq T_{i,k}(t), \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, k \in [1, A_i(t)], \tag{8h}$$

$$\frac{\beta_{k,0,i}(t) S_{i,k}(t)}{R_{i,g}(t)} + \sum_{j=1}^{M} \frac{\beta_{k,j,i}(t) S_{i,k}(t)}{R_{i,j}(t)} \leq T_{i,k}(t), \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, k \in [1, A_i(t)], \tag{8i}$$

We denote the system operation at the t-th time slot as $\mathbf{X}(t) \triangleq [\boldsymbol{f}(t), \boldsymbol{\beta}(t), \boldsymbol{P}(t), \boldsymbol{\alpha}(t)]$, where $\boldsymbol{f}(t) = \{f_i(t)|, \forall i \in \mathcal{N}\}$ denotes the computational frequency allocation factor, $\boldsymbol{\beta}(t) = \{\beta_{k,j,i}(t)|, \forall k \in [1, A_i(t)], \forall j \in \mathcal{M}, \forall i \in \mathcal{N}\}$ denotes the task offloading decision, $\boldsymbol{P}(t) = \{p_{i,j}(t)|, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}\}$ denotes the transmit power allocation factor and $\boldsymbol{\alpha}(t) = \{\alpha_{j,i}(t)|, \forall j \in M, \forall i \in N\}$ denotes the bandwidth resource allocation factor. (8b) gives the range constraint on computational frequency for all LEOs. (8c) and (8d) give the constraints on offloading ratio. (8e) indicates that the transmit power of all LEOs cannot exceed the maximum transmit power. (8f) and (8g) give constraints on the radio resource allocation factor. (8h) and (8i) show that the total local computing and remote offloading time of each task should be less than the maximum tolerant delay.

Note that (**P1**) is a non-convex optimization problem which is highly challenging to solve directly. Thus, it is necessary to design an algorithm to solve (**P1**) efficiently.

3 Proposed Energy-Efficient Layered Optimization Method

In this section, we present the solution to problem **(P1)** using the block coordinate descent method. We decompose **(P1)** into two subproblems: the optimization of computation resources and task offloading, and the optimization of radio resources. For each subproblem, we derive the optimal solution.

3.1 Computation Resource and Task Offloading Optimization

By substracting terms related to $\boldsymbol{f}(t)$ and $\boldsymbol{\beta}(t)$ in (**P1**), the computation resource and task offloading optimization sub-problem is formulated as follows.

$$(\mathbf{P2}): \min_{\boldsymbol{f}(t), \boldsymbol{\beta}(t)} \sum_{t=1}^{T} \sum_{i=1}^{N} \epsilon \left(f_i(t)^3 \right) \tag{9a}$$

$$\text{s.t. } 0 \le f_i(t) \le f_{\max}, \forall i \in \mathcal{N}, \forall t \in \mathcal{T} \tag{9b}$$

$$0 \le \beta_{k,j,i}(t) \le 1, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, k \in [1, A_i(t)], \tag{9c}$$

$$\sum_{j=0}^{M} \beta_{k,j,i}(t) \le 1, t \in \mathcal{T}, \forall i \in \mathcal{N}, k \in [1, A_i(t)], \tag{9d}$$

$$\frac{(1 - \sum_{j=0}^{M} \beta_{i,j,k}(t)) C_{i,k}(t) S_{i,k}(t)}{f_i(t)} \le T_{i,k}(t), \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, k \in [1, A_i(t)]. \tag{9e}$$

$$\frac{\beta_{k,0,i}(t) S_{i,k}(t)}{R_{i,g}(t)} + \sum_{j=1}^{M} \frac{\beta_{k,j,i}(t) S_{i,k}(t)}{R_{i,j}(t)} \le T_{i,k}(t), \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, k \in [1, A_i(t)], \tag{9f}$$

It is observed that (9a) is a convex function with respect to $\boldsymbol{f}(t)$ and (9b), (9c), (9d), (9f) are all convex constraints. In (9e), $\boldsymbol{f}(t), \boldsymbol{\beta}(t)$ are coupled, however, since $f_i(t) \geq 0$, (9e) can be transformed as $(1 - \sum\limits_{j=0}^{M} \beta_{i,j,k}(t))C_{i,k}(t)S_{i,k}(t) - f_i(t)T_{i,k}(t) \leq 0, \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, k \in [1, A_i(t)]$, which is a linear constraint with respect to $\boldsymbol{f}(t), \boldsymbol{\beta}(t)$. Therefore, by reformulating constraint (9e), (**P2**) can be regarded as a convex optimization problem, thus can be solved by CVX using interior points method.

3.2 Radio Resource Optimization

In this paper, radio resource optimization includes bandwidth and transmit power allocation. For fixed $\boldsymbol{f}(t)$ and $\boldsymbol{\beta}(t)$, the joint optimization of $\boldsymbol{\alpha}(t)$ and $\boldsymbol{P}(t)$ can be formulated as

$$(\mathbf{P3}): \min_{\boldsymbol{P}(t), \boldsymbol{\alpha}(t)} \Delta T \sum_{i=1}^{N} \left(\sum_{j=0}^{M} p_{i,j}(t) \right) \tag{10a}$$

$$\text{s.t.}\, 0 \leq p_{i,j}(t) \leq p_{\max}, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \tag{10b}$$

$$0 \leq \alpha_{j,i}(t) \leq 1, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \tag{10c}$$

$$\sum_{i=1}^{N} \alpha_{j,i}(t) \leq 1, \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \tag{10d}$$

$$\frac{\beta_{k,0,i}(t)S_{i,k}(t)}{R_{i,g}(t)} + \sum_{j=1}^{M} \frac{\beta_{k,j,i}(t)S_{i,k}(t)}{R_{i,j}(t)} \leq T_{i,k}(t), \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, k \in [1, A_i(t)], \tag{10e}$$

It is observed that (10a) is convex function with respect to $\boldsymbol{P}(t)$, and (10b), (10c), (10d) are convex constraints. Therefore, the main difficulty is to deal with (10e). To address this issue, we present the following lemma:

Lemma 1. *$R_{i,g}(t)$ and $R_{i,j}(t)$ are all jointly concave functions with respect to $\boldsymbol{P}(t), \boldsymbol{\alpha}(t)$.*

Proof. According to [13], we can use the property of perspective function to prove Lemma 1. Since $g(x,t) = tf(x/t)$, if f is a concave function, then $g(x,t)$ is a jointly concave function with respect to x, t. It is observed that $R_{i,g}(t)$ and $R_{i,j}(t)$ have the same mathematical property. Therefore, we only prove $R_{i,g}(t)$ for brevity. Specifically, by letting $x = p_{i,0}(t), t = \alpha_{0,i}(t), z = \dfrac{p_{i,0}(t)}{\alpha_{0,i}(t)}$, we have $f(z) = B_g \log\left(1 + \frac{z|h_{i,g}(t)|^2}{N_0 B_g}\right)$,. It is trivial to observe that $f(z)$ is a standard log function, and thus $f(z)$ is a concave function, according to the property of perspective function, $g(x,t)$ is a concave function, which indicates $R_{i,g}(t)$ is a concave function. Lemma 1 has been proved.

Algorithm 1. Block Coordinate Descent Method for (**P1**)

```
Observe ,$A_i(t)$, $h_{i,g}(t)$ and $h_{i,j}(t)$ in the current time slot.
Set $w = 0$ and the tolerance threshold $\xi$. Initialize task offloading matrix $\boldsymbol{f}^0(t), \boldsymbol{\beta}^0(t), \boldsymbol{\alpha}^0(t), \boldsymbol{P}^0(t)$.
repeat
    Find $\boldsymbol{f}^{w+1}(t), \boldsymbol{\beta}^{w+1}(t)$ by solving (P2) with given $\boldsymbol{\alpha}^w(t), \boldsymbol{P}^w(t)$.
    Obtain $\boldsymbol{\alpha}^{w+1}(t), \boldsymbol{P}^{w+1}(t)$ by solving (P3) with given $\boldsymbol{f}^{w+1}(t), \boldsymbol{\beta}^{w+1}(t)$.
    Denote the objective funtion of P1 as $O_{P_1}(t)$. Compute $O_{P_1}^{w+1}(t)$ using $\boldsymbol{f}^{w+1}(t)$, $\boldsymbol{\beta}^{w+1}(t)$, $\boldsymbol{\alpha}^{w+1}(t)$, $\boldsymbol{P}^{w+1}(t)$.
    $w \leftarrow w + 1$.
until $\left|O_{P_2}^{w+1}(t) - O_{P_2}^{w+1}(t)\right| \leq \xi$
Output: $\boldsymbol{f}^\star(t), \boldsymbol{\beta}^\star(t), \boldsymbol{P}^\star(t), \boldsymbol{\alpha}^\star(t)$.
```

Since $R_{i,g}(t)$ and $R_{i,j}(t)$ are all concave functions, $\frac{\beta_{k,0,i}(t)S_{i,k}(t)}{R_{i,g}(t)}$ and $\frac{\beta_{k,j,i}(t)S_{i,k}(t)}{R_{i,j}(t)}$ are all convex functions, and thus (10e) is a convex constraint, (**P3**) is a convex optimization problem with respect to $\boldsymbol{P}(t), \boldsymbol{\alpha}(t)$. Therefore, (**P3**) can be optimally solved using interial points method. By solving the above two subproblems in a iterally manner, we can obtain a near-optimal solution, the detailed procedure is given in Algorithm 1.

4 Simulation Results

The main parameters are given in Table 1.

To show the effectiveness of the proposed method, we compare our method with the following four benchmark algorithms.

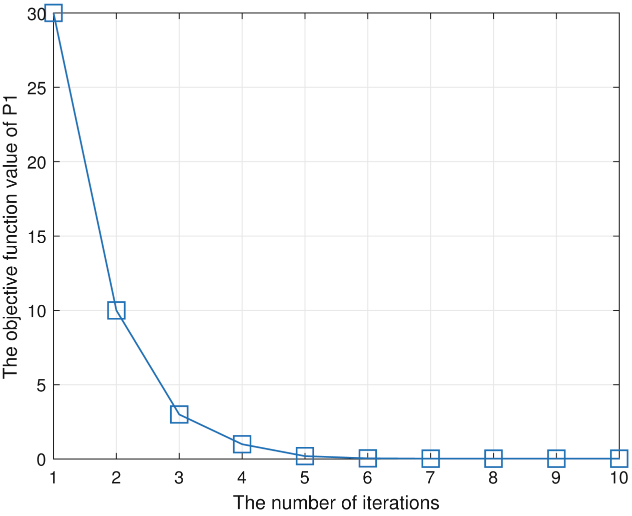

Fig. 2. The convergence performance of the proposed method.

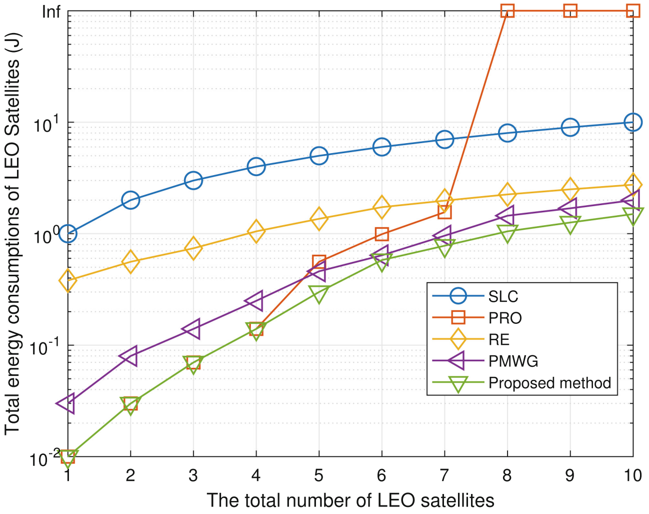

Fig. 3. The total energy consumptions of the proposed method compared with benchmark algorithms.

1. *Satellite local computing (SLC)*: This corresponds to the case where all computational tasks are executed locally. Specifically, we set $\boldsymbol{\beta}(t) = \mathbf{0}, \boldsymbol{P}(t) = \mathbf{0}, \boldsymbol{\alpha}(t) = \mathbf{0}$ and solve $(\mathbf{P1})$ in each time slot.
2. *Pure Remote offloading (PRO)*: In this case, the tasks are all offloaded to MEC servers. Specifically, we set $\sum_{j=0}^{M} \beta_{k,j,i}(t) = 1, \forall k \in [1, A_i(t)], \forall i \in \mathcal{N}, \forall t \in \mathcal{T}$ and solve $(\mathbf{P2})$ and $(\mathbf{P3})$ in each time slot.
3. *Random execution (RE)*: In this case, $\boldsymbol{f}(t), \boldsymbol{\beta}(t), \boldsymbol{P}(t), \boldsymbol{\alpha}(t)$ are set randomly in the feasible region. $(\mathbf{P2})$, $(\mathbf{P3})$ are solved according to Algorithm 1.
4. *The proposed method without GEO-MEC servers (PMWG)*: This method corresponds to the case where the LEOs are not assisted by GEO-MEC servers. Specifically, we set $M = 0$ and solve $(\mathbf{P2})$, $(\mathbf{P3})$ in each time slot.

Figure 2 illustrates the convergence performance of the proposed method. It is evident that the method converges after approximately 5 to 6 iterations. The simulation is conducted with $N = 5$, and upon convergence, the objective value of **(P1)** is 0.036. This indicates that the proposed method effectively reaches a near-optimal solution.

Figure 3 shows the performance of the proposed method compared with the 4 benchmark algorithms. It is observed that the proposed method outperforms the benchmark algorithms in terms of total energy consumptions. In Fig. 3, "Inf" means the constraints of P1 are not satisfied, resulting in a infeasible solution. With the increase of LEO satellites, the bandwidth resource is limited which limits the uplink channel capacity. Therefore, the PRO method may not satisfy the latency constraint, by contrast, the proposed method can adjust the offloading ratio according to the current state. It is also observed that the performance

Table 1. Main Simulation Parameters

Parameters	Value
Ground bandwidth B_g	20 MHz
Inter-satellite bandwidth B_u	20 MHz
Number of GEO-MEC Servers M	3
The height of LEO satellites H	600 km
Maximum transmission power $p_{\max}$	23 dBm
Noise power spectral density N_0	–174 dBm/Hz
Computational coefficient ϵ	10^{-26}
Maximum computational frequency $f_{\max}$	2 GHz
The size of the input tasks $S_{i,k}(t)$	$[5 \times 10^5, 5 \times 10^6]$ bits
The maximum tolerant latency $L_{i,k}(t)$	$[0.01, 0.1]$ s
CPU cycles for a unit bit $C_{i,k}(t)$	500 cycles/bit
The threshold of Algorithm 1 ξ	0.001

of the proposed method is better than PMWG, which indicates the importance of GEO-MEC servers.

5 Conclusion

In this paper, we address the joint computation offloading and resource allocation problem in LEO-MEC networks. We formulate the problem as a non-convex optimization problem and propose an energy-efficient layered optimization method to solve it effectively. Specifically, the original problem is decomposed into three subproblems: offloading ratio optimization, computation frequency optimization, and radio resource optimization. For each subproblem, we derive the optimal solution, and the final solution is obtained by iteratively solving these subproblems. Simulation results demonstrate the method's convergence and effectiveness.

For future work, we plan to extend the study to more realistic scenarios where LEO satellites can offload tasks to multiple GEO satellites and ground MEC servers. Additionally, we will explore using geometric methods to account for the visible time between LEO satellites, ground servers, and GEO satellites.

References

1. Abbas, N., Zhang, Y., Taherkordi, A., Skeie, T.: Mobile edge computing: a survey. IEEE Internet Things J. **5**(1), 450–465 (2018)
2. Taleb, T., et al.: On multi-access edge computing: a survey of the emerging 5G network edge cloud architecture and orchestration. IEEE Commun. Surv. Tutor. **19**(3), 1657–1681 (2017)

3. Song, Z., Hao, Y., Liu, Y., Sun, X.: Energy-efficient multiaccess edge computing for terrestrial-satellite Internet of Things. IEEE Internet Things J. **8**(18), 14202–14218 (2021)
4. Jing, Y., Wang, J., Jiang, C., Zhan, Y.: Satellite MEC with federated learning: architectures, technologies and challenges. IEEE Netw. **36**(5), 106–112 (2022)
5. Cao, X., et al.: Edge-assisted multi-layer offloading optimization of LEO satellite-terrestrial integrated networks. IEEE J. Sel. Areas Commun. **41**(2), 381–398 (2023)
6. Dong, Q., Xu, X., Han, S., Liu, R., Zhang, X.: ‘DDPG-based task offloading in satellite-terrestrial collaborative edge computing networks. In: 2023 IEEE International Conference on Communications Workshops (ICC Workshops), Rome, Italy (2023)
7. Zhang, X., et al.: Energy-efficient computation peer offloading in satellite edge computing networks. IEEE Trans. Mob. **23**(4), 3077–3091 (2024)
8. Minglei, T., Song, L., Wanjiang, H., Xiaoxiang, W.: Online learning-based offloading decision and resource allocation in mobile edge computing-enabled satellite-terrestrial networks. China Commun. **21**(3), 230–246 (2024)
9. Rodrigues, T.K., Kato, N.: Hybrid centralized and distributed learning for MEC-equipped satellite 6G networks. IEEE J. Sel. Areas Commun. **41**(4), 1201–1211 (2023)
10. Hao, Y., Song, Z., Zheng, Z., Zhang, Q., Miao, Z.: Joint communication, computing, and caching resource allocation in LEO satellite MEC networks. IEEE Access **11**, 6708–6716 (2023)
11. Yang, Z., Bi, S., Zhang, Y.-J.A.: Dynamic offloading and trajectory control for LEO-enabled mobile edge computing system with energy harvesting devices. IEEE Trans. Wirel. Commun. **21**(12), 10515–10528 (2022)
12. Wang, Y., Sheng, M., Wang, X., Wang, L., Li, J.: Mobile-edge computing: partial computation offloading using dynamic voltage scaling. IEEE Trans. Commun. **64**(10), 4268–4282 (2016)
13. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)