Collaborative Computing Services at Ground, Air, and Space: An Optimization Approach

Yan Kyaw Tun, *Member, IEEE*, Ki Tae Kim, Luyao Zou, Zhu Han, *Fellow, IEEE*, György Dán, *Senior Member, IEEE*, and Choong Seon Hong, *Senior Member, IEEE*

***Abstract*—Multi-access edge computing (MEC)-enabled integrated space-air-ground networks have drawn much attention recently, as they can provide communication and computing services to wireless devices in areas that lack terrestrial base stations (TBSs). They could make it possible for battery-powered Internet of Things (IoT) devices to offload their computation tasks to MEC-enabled unmanned aerial vehicles (UAVs) assisted aerial networks and low earth orbit (LEO) satellites and thus reduce their energy consumption and allow them to complete the execution of tasks on time. However, due to the limited computation capacity of the MEC servers at UAVs and satellites, an efficient offloading decision and computation resource allocation scheme is essential. Therefore, this paper investigates the problem of minimizing the latency experienced by the wireless devices in the MEC-enabled integrated space-air-ground network by optimizing the offloading decision while assuring the energy constraints of both devices and UAVs. The problem is proved to be a non-convex problem, and the block successive upper-bound minimization (BSUM) method is proposed as a solution. Finally, extensive simulation results are presented to exhibit the effectiveness of the BSUM algorithm in solving the proposed problem.**

***Index Terms*—Multi-access edge computing (MEC), integrated space-air-ground networks, task offloading, resource allocation, block successive upper-bound minimization (BSUM).**

I. Introduction

MEC-enabled integrated space-air-ground networks have recently emerged as a potential technology for providing remote computation services to the Internet of Things (IoT) devices in the deep seas, in mountainous regions, and in disaster areas where terrestrial infrastructures (i.e., TBSs) do not exist [1], [2]. Integrated space-air-ground networks make it possible for energy-constrained IoT devices to offload their computation tasks thereby extending their service lifetime and allowing computationally demanding applications to be executed in remote areas [3], [4]. In order for computation offloading to cater for application requirements, however, there is a need for efficient offloading decision schemes that can ensure task execution within the deadlines of tasks.

This problem was addressed in [5] and [6], which proposed an optimization approach for efficient offloading and resource allocation schemes in integrated space-air-ground networks. Moreover, machine learning-based efficient task offloading and resource allocation problems in the integrated space-air-ground networks were studied in [7]. These existing studies assume that the UAVs in the considered integrated space-air-ground networks operate independently (i.e., no collaboration) [8]–[11]. In other words, if a UAV's computation capacity is insufficient to compute the offloading tasks of its associated devices, it will offload all tasks directly to the satellite, leaving underutilized neighboring UAVs with abundant computation capacity. Clearly, devices may experience lower latencies if their tasks could be relayed to nearby UAVs, but how to coordinate the offloading among UAVs in integrated space-air-ground networks is an intriguing problem yet unsolved.

To the best of our knowledge, this paper is the first to address the latency minimization problem in MEC-enabled integrated space-air-ground networks while incorporating UAV collaboration by concurrently optimizing task offloading decisions of both UAVs and devices. The main contributions of this paper are as follows:

- We first formulate the latency minimization problem in a MEC-enabled integrated space-air-ground network by optimizing the offloading decisions, i.e., decisions of both the devices and the UAVs, while satisfying their energy constraints.
- We then show that the formulated problem is a non-convex mixed integer programming problem due to the coupling of decision variables in the objective function and constraints. Therefore, to address the problem, we propose to relax the binary constraints and then propose to use the block successive upper-bound minimization (BSUM) algorithm to solve the relaxed problem.
- Finally, we demonstrate the convergence of the proposed algorithm by using extensive simulations. Furthermore, to show the effectiveness of our proposed algorithm, we compare the results of our proposed algorithm to baseline schemes.

The rest of this paper is organized as follows. The system model is described in Section II. Section III presents the

This work was partly supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00207816), and part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korea Government (MSIT) (Artificial Intelligence Innovation Hub) under Grant 2021-0-02068, (No. RS-2022-00155911, Artificial Intelligence Convergence Innovation Human Resources Development (Kyung Hee University)), and (No.2019-0-01287, Evolvable Deep Learning Model Generation Platform for Edge Computing), and the MSIT under the ITRC support program(IITP-2023-RS-2023-00258649) supervised by the IITP. Dr. Choong Seon Hong is the corresponding author.

This work was partly funded by the Swedish Research Council through project 2020-03860 and by the Swedish Foundation for Strategic Research through the CLAS project (grant RIT17-0046).

Yan Kyaw Tun, and György Dán are with Division of Network and Systems Engineering, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Brinellvägen 8, 114 28 Stockholm, Sweden, e-mail:{yktun, gyuri}@kth.se.

Zhu Han is with the Electrical and Computer Engineering Department, University of Houston, Houston, TX 77004, and the Department of Computer Science and Engineering, Kyung Hee University, Yongin-si, Gyeonggi-do 17104, Rep. of Korea, email{zhan2}@uh.edu.

Ki Tae Kim, Luyao Zou, and Choong Seon Hong are with the Department of Computer Science and Engineering, Kyung Hee University, Yongin-si, Gyeonggi-do 17104, Rep. of Korea, e-mails:{glideslope, zouluyao, cshong}@khu.ac.kr.

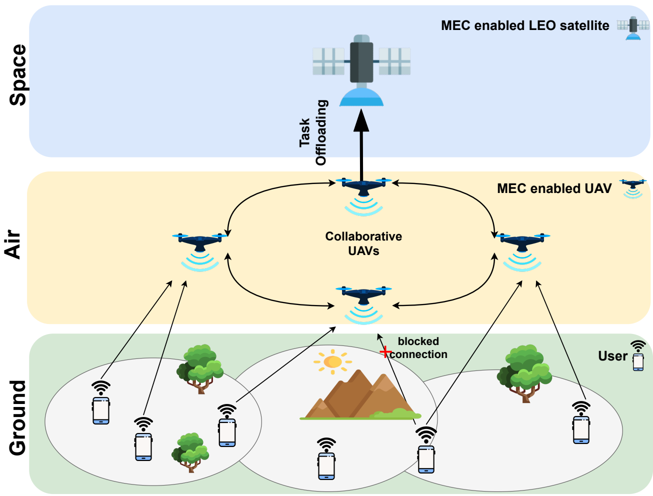


Fig. 1: A proposed system model.

problem formulation and the proposed solution. Simulation results are shown in Section IV. Section V concludes the paper.

II. System Model

As shown in Fig. 1, we consider a MEC system in the integrated space-air-ground network consisting of a set $\mathcal{J}$ of J wireless devices, a set $\mathcal{K}$ of K UAVs, and a LEO satellite. Each device $j \in \mathcal{J}$ in the considered network has a latency-sensitive computation task T_j, which can be represented by a tuple $T_j = \{\varphi_j, \alpha_j, A_j\}$, where φ_j is the maximum tolerable latency of the task, α_j is the required CPU cycles to process one bit of data, and A_j is the total input data size of the task. Wireless devices can either perform their computation tasks locally or they can offload tasks to MEC-enabled UAVs or to the satellite to be computed remotely. We assume that the 2 GHz frequency band is used for communication between devices and UAVs, while the 28 GHz (mmWave) frequency band is used for communication between UAVs and the satellite.

A. Communication Model

We consider that the wireless devices are outside of the coverage area of TBSs, hence, the wireless devices cannot offload their computation tasks to those. As a result, wireless devices offload tasks to the nearest UAV $k \in \mathcal{K}$ with the best channel quality. We define $w_j^k \in \{0,1\}$ as the task offloading decision variable, which represents whether or not the task of wireless device j is offloaded to UAV k by using a wireless link,

$$w_j^k = \begin{cases} 1, & \text{if computation task } T_j \text{ of device } j \text{ is offloaded} \\ & \text{to UAV } k, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Let us denote the horizontal coordinates of wireless device j and UAV k by $[x_j, y_j]^T$ and $[x_k, y_k]^T$, respectively. Moreover, h_k represents the hovering altitude of UAV k. Thus, the distance between device j and UAV k is

$$d_j^k = \sqrt{(x_k - x_j)^2 + (y_k - y_j)^2 + h_k^2}, \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}. \tag{2}$$

Then, by adopting the free-space path loss model, the channel gain between device j and UAV k is given by [12]

$$g_j^k = \frac{g_0}{(d_j^k)^{\vartheta}}, \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}, \tag{3}$$

where g_0 denotes the channel gain at the reference distance $d_0 = 1$ m, and ϑ is the path loss exponent. We can then calculate the spectrum efficiency of device j as

$$\gamma_j^k = \log_2\left(1 + \frac{P_j g_j^k}{I_j^k + \sigma^2}\right), \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}, \tag{4}$$

where P_j is the uplink transmission power of wireless device j, and g_j^k is the achievable channel gain between wireless device j and UAV k, σ^2 is the noise power and $I_j^k = \sum_{k' \in \mathcal{K}, k' \neq k} \sum_{j' \in \mathcal{J}, j' \neq j} P_{j'} g_{j'}^k$ is the interference at UAV k.

Finally, the instantaneous data rate achieved by wireless device j associated with UAV k can be calculated as

$$R_j^k = \frac{B^k}{|\mathcal{J}_k|}\gamma_j^k, \qquad \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}, \tag{5}$$

where $|\mathcal{J}_k|$ is the total number of wireless devices that offload their computation tasks to UAV k. Depending on the instantaneous data rate, the transmission latency experienced by device j when offloading its computation task to UAV k is given by [6] [13]

$$l_j^k = \frac{A_j}{R_j^k}, \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}. \tag{6}$$

The energy consumption for task offloading from wireless device j to UAV k is [6] [13]

$$E_j^k = \frac{P_j A_j}{R_j^k}, \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}. \tag{7}$$

After receiving the offloaded tasks from its associated users, UAV k decides to either process the tasks locally on its server or offload them to neighboring UAVs or the satellite, depending on its available computation capacity.

Thus, we define the binary decision variable $z_j^{k \to k'} \in \{0,1\}$, which indicates whether or not the computation task T_j is offloaded to UAV k'

$$z_j^{k \to k'} = \begin{cases} 1, & \text{if computation task } T_j \text{ is offloaded from} \\ & \text{UAV } k \text{ to UAV } k', \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

We make the reasonable assumption that there is a line-of-sight (LoS) communication link between UAVs. Thus, the achievable channel gain between UAV k and k' is [14]

$$\Gamma^{k \to k'} = 10^{L^{k \to k'}/10}, \forall k, k' \in \mathcal{K}, \tag{9}$$

where $L^{k \to k'} = \varsigma^{k \to k'} + \vartheta^{k \to k'}$ is the path loss between UAVs k and k', where $\varsigma^{k \to k'}$ is the attenuation factor for a LoS link, and $\vartheta^{k \to k'}$ is given by [14]

$$\vartheta^{k \to k'} = 20\log_{10}(d_k^{k'}) + 20\log_{10}(f_c) + 10\log_{10}\left[\left(\frac{2\pi}{c}\right)^2\right], \tag{10}$$

where f_c is the carrier frequency, c is the speed of light, and $d_k^{k'}$ is the distance between UAVs k and k'. Finally, we can calculate the achievable data rate between UAV k and k' as

$$R^{k \to k'} = B^{k \to k'} \log_2\left(1 + \Gamma^{k \to k'}\right), \forall k, k' \in \mathcal{K}, \tag{11}$$

where $B^{k \to k'}$ is the available bandwidth between UAV k and UAV k'. The transmission latency between UAV k and UAV k' is [6] [13]

$$l_j^{k\to k'} = \frac{\sum\limits_{j\in\mathcal{J}_k} z_j^{k\to k'} A_j}{R^{k\to k'}}, \forall k, k' \in \mathcal{K}. \quad (12)$$

Furthermore, the transmission energy of UAV k can be calculated as [6] [13]

$$E^{k\to k'} = P^{k\to k'} \left(\frac{\sum\limits_{j\in\mathcal{J}_k} z_j^{k\to k'} A_j}{R^{k\to k'}} \right), \forall k, k' \in \mathcal{K}. \quad (13)$$

Consequently, let us define the binary decision variable $z_j^{k\to s}$, which indicates whether or not the computation is offloaded to the satellite as

$$z_j^{k\to s} = \begin{cases} 1, & \text{if computation task } T_j \text{ is offloaded from} \\ & \text{UAV } k \text{ to the satellite,} \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

The achievable signal to noise ratio between UAV k and the satellite can be expressed as [15]

$$\Gamma^{k\to s} = \frac{P^{k\to s} g_k^{\text{tx}} g_s^{\text{rx}} L_r}{t_n a B_{\text{mm}}^{k\to s}} \left(\frac{c}{4\pi d_k^s f_c^{\text{mm}}} \right)^2, \quad (15)$$

where $P^{k\to s}$ is the transmit power of UAV k, g_k^{tx} and g_s^{rx} are the antenna gains of the transmitter and receiver, L_r is the attenuation factor, t_n is the noise temperature, a is the Boltzmann's constant, f_c^{mm} is the mmWave carrier frequency, and d_k^s reflects the distance between UAV k and the satellite. Then, the available mmWave backhaul link capacity between UAV k and the satellite, can be expressed as [15]

$$R^{k\to s} = B_{\text{mm}}^{k\to s} \log_2\left(1 + \Gamma^{k\to s}\right), \forall k \in \mathcal{K}, \quad (16)$$

where $B_{\text{mm}}^{k\to s}$ is the mmWave bandwidth between UAV k and the satellite. Therefore, the transmission latency between UAV k and the satellite can be expressed as [6] [13]

$$l_j^{k\to s} = \frac{\sum\limits_{j\in\mathcal{J}_k} z_j^{k\to s} A_j}{R^{k\to s}}, \forall k \in \mathcal{K}. \quad (17)$$

Furthermore, the transmission energy of UAV k can be expressed as [6] [13]

$$E^{k\to s} = P^{k\to s} \left(\frac{\sum\limits_{j\in\mathcal{J}_k} z_j^{k\to s} A_j}{R^{k\to s}} \right), \forall k \in \mathcal{K}. \quad (18)$$

B. Computing Model

1) Local Computing Model: If wireless device $j \in \mathcal{J}$ decides to perform its computation task locally (i.e., $w_j^k = 0$), the latency experienced by the wireless device to complete the task is given by

$$l_j^{\text{loc}} = \frac{\alpha_j A_j}{F_j}, \forall j \in \mathcal{J}, \quad (19)$$

where F_j is the computation capacity (i.e., cycle/s) of wireless device j. Then, the local energy consumption of wireless device j can be expressed as

$$E_j^{\text{loc}} = \varpi_j (F_j)^2 \alpha_j A_j, \forall j \in \mathcal{J}, \quad (20)$$

where ϖ_j is a constant that is dependent on the chip architecture of the wireless device [16].

2) Remote Computing Model: We define the binary variable $z_j^k \in \{0, 1\}$ that indicates whether or not wireless device j's computation task is performed at UAV k, i.e.,

$$z_j^k = \begin{cases} 1, & \text{if computation task } T_j \text{ is computed at} \\ & \text{UAV } k, \\ 0, & \text{otherwise.} \end{cases} \quad (21)$$

If the task of wireless device j is computed at UAV k, i.e., $z_j^k = 1$, then the computation latency is

$$l_j^{k,\text{comp}} = \frac{\alpha_j A_j}{f_j^k}, \quad (22)$$

where f_j^k is the computation capacity of UAV k that is allocated to wireless device j and which can be formulated by using weighted proportional allocation [16] as

$$f_j^k = F_k^{\mathbf{max}} \frac{\alpha_j A_j}{\sum\limits_{i\in\mathcal{J}} z_i^k \alpha_i A_i}, \quad (23)$$

where $F_k^{\mathbf{max}}$ is the maximum computation capacity at UAV k. Consequently, the cumulative latency experienced by wireless device j when its computation task is offloaded to UAV k is

$$l_j^{k,e} = l_j^k + l_j^{k,\text{comp}}, \forall j \in \mathcal{J}_k, k \in \mathcal{K}. \quad (24)$$

Furthermore, the energy consumption at UAV k for executing the computation task T_j of wireless device j can be formulated as

$$E_j^{k,\text{comp}} = \varpi_k (f_j^k)^2 \alpha_j A_j, \forall j \in \mathcal{J}, \forall k \in \mathcal{K}, \quad (25)$$

where ϖ_k is a constant that depends on the chip architecture of the server at UAV k. Moreover, the total execution latency experienced by wireless device j when its computation task is offloaded to UAV k' is

$$l_j^{k\to k',e} = l_j^k + l_j^{k\to k'} + l_j^{k',\text{comp}}, \forall j \in \mathcal{J}_k, \text{and } k, k' \in \mathcal{K}. \quad (26)$$

Finally, let us denote the computation latency when the computation task of device j associated with UAV k is computed at the satellite by $l_j^{s,\text{comp}}$, which can be calculated based on (22). We consider that the satellite has a renewable energy resource, we can neglect the computation energy consumption at the satellite. Then, the total execution latency experienced by wireless device j when its computation task is offloaded to the satellite is

$$l_j^{k\to s,e} = l_j^k + l_j^{k\to s} + l_j^{s,\text{comp}} + 2 l_j^{k\to s,\text{pro}}, \forall j \in \mathcal{J}_k, \text{and } k \in \mathcal{K}, \quad (27)$$

where $l_j^{k\to s,\text{pro}} = \frac{d_k^s}{c}$ is the propagation delay between UAV k and the satellite. The downlink output data transmission latency and energy usage are insignificant compared to the uplink input data transmission because the output data size is relatively small compared to the input data size. As a consequence, the downlink data transmission is not covered in our model. Thus, the total latency experienced by wireless device j when its task is computed remotely is given by

$$l_j^{\text{off}} = z_j^k l_j^{k,e} + \sum_{k'\in\mathcal{K}, k'\neq k} z_j^{k\to k'} l_j^{k\to k',e} + z_j^{k\to s} l_j^{k\to s,e}, \quad (28)$$

Finally, the total energy consumption to complete the execution of the computation tasks of UAV k

$$E_k^{\text{Tot}} = \sum_{j\in\mathcal{J}} z_j^k E_j^{k,\text{comp}} + \sum_{k'\in\mathcal{K}, k'\neq k} E^{k\to k'} + E^{k\to s} + E^{k,\text{hov}}, \quad (29)$$

where $E^{k,\text{hov}}$ is the hovering energy and is given by

$$E^{k,\text{hov}} = P^{k,\text{hov}} l^{k,\text{hov}}, \forall k \in \mathcal{K}, \tag{30}$$

where $P^{k,\text{hov}}$ and $l^{k,\text{hov}} = \max_{j \in \mathcal{J}}(\varphi_j)$ are the hovering power and hovering time of UAV k, respectively.

III. Problem Formulation and Proposed Solution

We consider that the objective is to minimize the total latency experienced by devices while assuring the energy constraints of devices ($E_j^{\mathbf{max}}$) and of the UAVs ($E_k^{\mathbf{max}}$) are met. Our objective function can thus defined as $\mathbf{O}(\boldsymbol{w}, \boldsymbol{z}) = \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}} (1 - w_j^k) l_j^{\text{loc}} + w_j^k l_j^{\text{off}}$. Finally, we can formulate the latency minimization problem as

$$\mathbf{P:} \underset{\boldsymbol{w},\ \boldsymbol{z}}{\text{minimize}} \quad \mathbf{O}(\boldsymbol{w}, \boldsymbol{z}) \tag{31a}$$

$$\text{subject to} \quad E_j^{\text{loc}} + E_j^k \leq E_j^{\mathbf{max}}, \quad \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}, \tag{31b}$$

$$\sum_{j \in \mathcal{J}} z_j^k E_j^{k,\text{comp}} + \sum_{k' \in \mathcal{K}, k' \neq k} E_j^{k \to k'} + E^{k \to s} + E_k^{\text{hov}} \leq E_k^{\mathbf{max}}, \forall k \in \mathcal{K}, \tag{31c}$$

$$(1 - w_j^k) + w_j^k (z_j^k + \sum_{\substack{k' \in \mathcal{K}, \\ k' \neq k}} z_j^{k \to k'} + z_j^{k \to s}) = 1, \forall j \in \mathcal{J}_k, \tag{31d}$$

$$z_j^k \in \{0,1\}, z_j^{k \to k'} \in \{0,1\}, z_j^{k \to s} \in \{0,1\}, \forall k \in \mathcal{K}, \tag{31e}$$

$$w_j^k \in \{0,1\}, \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}, \tag{31f}$$

where (31b) and (31c) express the energy constraints of both wireless devices and UAVs. (31d) denotes that the computation task of a wireless device can only be executed in one location. Finally, (31e) and (31f) represent the binary decision variables of wireless devices and UAVs.

Due to the coupling between decision variables in the objective function, the nonlinear and binary constraints, and the non-convex structure, it is impossible to solve the optimization problem in (31) via convex optimization techniques. Therefore, we propose to use the BSUM method to solve the proposed problem [17]. BSUM is a novel approach for solving non-convex and non-smooth optimization problems by decomposing the problem into small subproblems. In the BSUM method, the upper bound of the objective function is minimized by iteratively updating the decision variables $\boldsymbol{w}$, and $\boldsymbol{z}$. Furthermore, BSUM can guarantee to converge to the stationary points of the objective function in (31). To apply the BSUM approach, firstly, we relax the binary constraints in (31e) into continuous ones, i.e., $w_j^k \in [0,1]$, $z_j^k \in [0,1]$, $z_j^{k \to k'} \in [0,1]$, and $z_j^{k \to s} \in [0,1]$. Then, we can introduce the feasible sets of $\boldsymbol{w}$, and $\boldsymbol{z}$ as

$$\mathcal{W} \triangleq \{\boldsymbol{w} : E_j^{\text{loc}} + E_j^k \leq E_j^{\mathbf{max}}, (1 - w_j^k) + w_j^k (z_j^k + \sum_{\substack{k' \in \mathcal{K}, \\ k' \neq k}} z_j^{k \to k'} + z_j^{k \to s}) = 1, w_j^k \in [0,1], \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}\},$$

TABLE I: Simulation Parameters.

Parameter	Value	Parameter	Value
g_0	-50 dB	P_j	23 dBm
σ^2	-174 dBm	A_j	[10, 50] Mbits
α_j	[10, 50] Cycles	f_j	[0.5, 3] MHz
ϖ_k, ϖ_j	5×10^{-27}	B^k	3 MHz
ϑ	2	$P^{k \to k'}$	30 dBm
$B^{k \to k'}$	1.7 MHz	$B_{\text{mm}}^{k \to s}$	1.8 MHz
$E_j^{\mathbf{max}}$	10 kJ	$E_k^{\mathbf{max}}$	50 kJ
f_c^{mm}	28 GHz	$P^{k \to s}$	30 dBm
θ_n	0.5		

$$\mathcal{Z} \triangleq \{\boldsymbol{z} : \sum_{j \in \mathcal{J}} z_j^k E_j^{k,\text{comp}} + \sum_{k' \in \mathcal{K}, k' \neq k} E_j^{k \to k'} + E^{k \to s} + E_k^{\text{hov}} \leq E_k^{\mathbf{max}}, (1 - w_j^k) + w_j^k (z_j^k + \sum_{\substack{k' \in \mathcal{K}, \\ k' \neq k}} z_j^{k \to k'} + z_j^{k \to s}) = 1, z_j^k, z_j^{k \to k'}, z_j^{k \to s} \in [0,1], \forall j \in \mathcal{J}_k, \forall k \in \mathcal{K}\},$$

Finally, we define the proximal upper-bound function, $\mathbf{O}_n$ of the objective function in (31a) for each iteration t, $\forall n \in \mathcal{N}$ where $\mathcal{N}$ is the index set. In order to ensure that the proximal upper-bound function $\mathbf{O}_n$ is convex, we apply a quadratic penalization to the objective function in (31a) as

$$\mathbf{O}_n(\boldsymbol{w}_n; \boldsymbol{w}^t, \boldsymbol{z}^t) = \mathbf{O}(\boldsymbol{w}_n; \boldsymbol{w}^{(t-1)}, \boldsymbol{z}^{(t-1)}) + \frac{\theta_n}{2} \parallel (\boldsymbol{w}_n - \boldsymbol{w}^{(t-1)}) \parallel^2 . \tag{32}$$

where θ_n is the penalty parameter, which can be used for the other vector of variables, $\boldsymbol{z}_n$. In addition, at each iteration, t, the proximal upper-bound function in (32) has the minimizer vectors $\boldsymbol{w}^{(t-1)}$, and $\boldsymbol{z}^{(t-1)}$ with regard to $\boldsymbol{w}_n$ and $\boldsymbol{z}_n$, which are taken to be the solution of the previous iteration. The solution at iteration $(t+1)$ can then be updated by solving the following optimization problems:

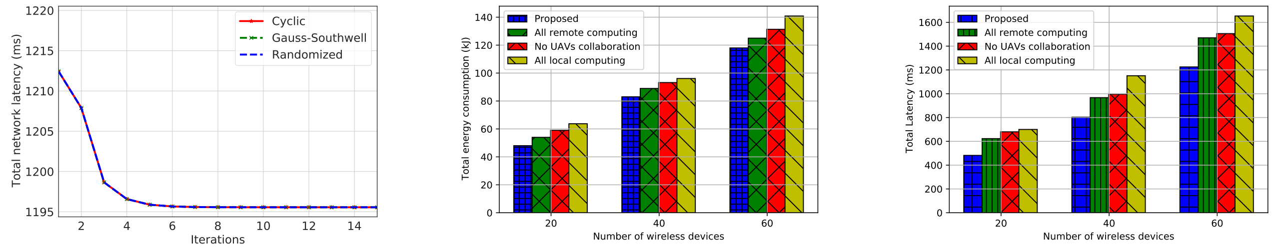
$$\boldsymbol{w}_n^{t+1} \in \underset{\boldsymbol{w}_n}{\text{argmin}} \quad \mathbf{O}_n\Big(\boldsymbol{w}_n; \boldsymbol{w}^{(t)}, \boldsymbol{z}^{(t)}\Big), \tag{33}$$

$$\boldsymbol{z}_n^{(t+1)} \in \underset{\boldsymbol{z}_n}{\text{argmin}} \quad \mathbf{O}_n\Big(\boldsymbol{z}_n; \boldsymbol{z}^{(t)}, \boldsymbol{w}^{(t+1)}\Big). \tag{34}$$

Then, the subproblems in (33) and (34) can be solved using the CVXPY toolkit. Finally, a summary of our proposed BSUM-based algorithm for task offloading decision in MEC-enabled integrated space-air-ground networks is presented in Algorithm 1. In our work, we utilize the CVXPY toolkit, which implements an Embedded Conic Solver (ECOS), to solve subproblems (33) and (34) [18]. For a problem with N variables, the computational complexity of the interior point method [18] used in the ECOS solver is $\mathcal{O}(N)^{3.5}$. The number of variables in subproblems (33) and (34) is J and $K(K+1)$, respectively. Thus, the computational complexity of the proposed BSUM-based task offloading decision in MEC-enabled integrated space-air-ground networks is $\mathcal{O}(J^{3.5}) + \mathcal{O}((K(K+1))^{3.5})$.

IV. Simulation results

To evaluate the proposed solution, we consider wireless devices distributed within an area of 400 m $\times$ 400 m. To provide computing services to the devices, 3 MEC-enabled



(a) (b) (c)

Fig. 2: (a) shows the convergence of the proposed algorithm. (b) shows the energy consumption under different number of devices.(c) shows total latency under different number of devices.

Algorithm 1 BSUM-Based Algorithm for Task Offloading Decision in MEC-enabled Integrated Space-Air-Ground Networks

1: **Initialization:** Set $t = 0$, $\epsilon_1 = 10^{-4}$, and find initial feasible solutions $(\boldsymbol{w}^{(0)}, \boldsymbol{z}^{(0)})$;
2: **repeat**
3: Choose index set $\mathcal{N}$;
4: Let $\boldsymbol{w}_n^{(t+1)} \in \underset{\boldsymbol{w}_n}{\text{argmin}}\ \mathbf{O}_n\big(\boldsymbol{w}_n; \boldsymbol{w}^{(t)}, \boldsymbol{z}^{(t)}\big)$;
5: Set $\boldsymbol{w}_m^{(t+1)} = \boldsymbol{w}_m^t, \forall m \notin \mathcal{N}$;
6: Find $\boldsymbol{z}_n^{(t+1)}$ by solving (34);
7: $t = t+1$;
8: **until** $\| \frac{\mathbf{O}_n^{(t)} - \mathbf{O}_n^{(t+1)}}{\mathbf{O}_n^{(t)}} \| \leq \epsilon_1$
9: Then, set $\big(\boldsymbol{w}_n^{(t+1)}, \boldsymbol{z}_n^{(t+1)}\big)$ as the desired solution.

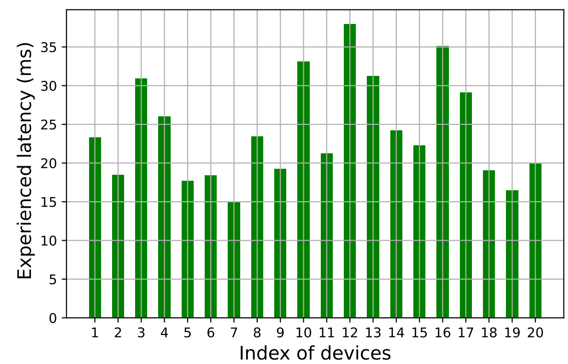


Fig. 3: Maximum latency experienced by each device.

UAVs hover at an altitude of 50 m. Additionally, a LEO satellite at an altitude of 780 km is taken into consideration to execute the devices' offloaded tasks that the UAVs cannot handle. The rest of the simulation parameters are shown in Table I. We executed the algorithms implemented in Python on a PC with Intel(R) Core(TM) i5-8500 CPU 3.00 GHz, 32.0 GB RAM, and NVIDIA GeForce GTX 1660 Ti. As a basis for comparison, we use three baseline schemes proposed in the recent literature [5] and [13], namely: 1) All remote computing scheme where devices offload all of their tasks UAVs to perform remote computing, 2) All local computing scheme where devices compute their tasks locally, and 3) No UAVs collaboration scheme where the UAV computation capacity is insufficient to execute the offloaded tasks of its associated devices, the UAV directly transferred the tasks of its devices to the satellite using mmWave backhaul links without checking its neighboring UAVs which have sufficient computation to execute its computation tasks.

We illustrate the convergence of the proposed BSUM algorithm in Fig. 2a. As shown in the figure, our proposed algorithm converges to the stationary solution in fewer than 10 iterations. As a result, our proposed algorithm can be applied in large-scale networks. Furthermore, we also compare the total latency experienced by wireless devices (i.e., the objective function of our optimization problem in (31a)) under various block selection rules, including Cyclic, Gauss-Southwell, and Randomized. The figure shows that all block selection rules achieve the same total latency and courage equally fast convergence. Fig. 2b shows the total energy consumption (i.e., energy consumption of both UAVs and wireless devices) for various number of wireless devices. The figure shows as the number of wireless devices increases, so does the total energy usage of the network. The figure also shows the energy consumption under three baseline schemes. As shown in Fig. 2b, the energy consumption under our proposed scheme is the lowest when compared with the benchmark schemes. Moreover, Fig. 2c shows the total network latency for different numbers of wireless devices. The figure shows that our proposed algorithm can achieve the lowest total latency when compared to the benchmark schemes. As a result, we conclude that our proposed algorithm is more effective for the considered space-air-ground networks than benchmark schemes. Fig. 3 shows the maximum latency experienced among devices (i.e., the latency experienced by each device) when there are 20 devices in the considered networks. From the figure, we observe that the latency experienced by device-12 is the highest. It is because device-12 has the largest input data size and the required CPU cycles to compute its data compared to other devices.

Finally, Fig. 4 shows the total latency for various number of UAVs and wireless devices. We can observe that when there are few wireless devices in the network, e.g., 20 devices, the total latency experienced by the wireless devices does not decrease considerably, even when more UAVs are deployed. However, when the network size increases, i.e., increase the number of wireless devices in the network, the total latency lowers dramatically when more UAVs are deployed. The rationale for this is that deploying a few UAVs is sufficient to handle the computation tasks of wireless devices when the

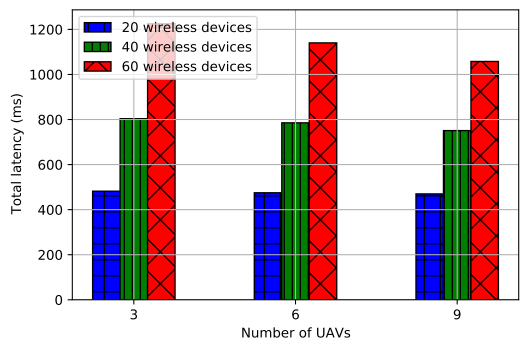


Fig. 4: Total latency under different number of UAVs.

network size is small. Furthermore, deploying more UAVs in a small network size can result in increased total energy consumption. However, as the network grows in size, it takes longer to complete users' tasks, resulting in increased total latency experienced by devices. As a result, more UAVs will be needed to ensure low total latency in this scenario.

V. Conclusions

In this paper, we considered MEC-enabled integrated space-air-ground networks to provide computation services to wireless devices in remote areas. We formulated the offloading decisions problem for the considered network to minimize the total latency experienced by the wireless devices. In order to solve the formulated non-convex problem, we first relaxed the binary constraints into continuous ones and we proposed a BSUM method to address the relaxed problem. Finally, we conducted comprehensive simulations to demonstrate the effectiveness of the proposed algorithm, and it is found that when compared to benchmark schemes, our proposed method significantly reduces energy consumption and the total latency experienced by the wireless devices.

References

[1] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-air-ground integrated network: A survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2714–2741, Fourth Quarter 2018.

[2] C.-Q. Dai, J. Luo, S. Fu, J. Wu, and Q. Chen, "Dynamic user association for resilient backhauling in satellite–terrestrial integrated networks," *IEEE Systems Journal*, vol. 14, no. 4, pp. 5025–5036, Dec. 2020.

[3] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable iot architecture based on transparent computing," *IEEE Network*, vol. 31, no. 5, pp. 96–105, Aug. 2017.

[4] L. Liu, C. Chen, T. Qiu, M. Zhang, S. Li, and B. Zhou, "A data dissemination scheme based on clustering and probabilistic broadcasting in vanets," *Vehicular Communications*, vol. 13, pp. 78–88, Jul. 2018.

[5] S. Mao, S. He, and J. Wu, "Joint UAV position optimization and resource scheduling in space-air-ground integrated networks with mixed cloud-edge computing," *IEEE Systems Journal*, vol. 15, no. 3, pp. 3992–4002, Dec. 2020.

[6] Y. K. Tun, Y. M. Park, N. H. Tran, W. Saad, S. R. Pandey, and C. S. Hong, "Energy-efficient resource management in UAV-assisted mobile edge computing," *IEEE Communications Letters*, vol. 25, no. 1, pp. 249–253, Jan. 2021.

[7] H. Liao, Z. Zhou, X. Zhao, and Y. Wang, "Learning-based queue-aware task offloading and resource allocation for space–air–ground-integrated power IoT," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5250–5263, Apr. 2021.

[8] N. Waqar, S. A. Hassan, A. Mahmood, K. Dev, D.-T. Do, and M. Gidlund, "Computation offloading and resource allocation in mec-enabled integrated aerial-terrestrial vehicular networks: A reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 21 478–21 491, Jun. 2022.

[9] Y. Chen, B. Ai, Y. Niu, H. Zhang, and Z. Han, "Energy-constrained computation offloading in space-air-ground integrated networks using distributionally robust optimization," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 11, pp. 12 113–12 125, Nov. 2021.

[10] G. Wang, S. Zhou, and Z. Niu, "Radio resource allocation for bidirectional offloading in space-air-ground integrated vehicular network," *Journal of Communications and Information Networks*, vol. 4, no. 4, pp. 24–31, Dec. 2019.

[11] B. Chen, N. Li, Y. Li, X. Tao, and G. Sun, "Energy efficient hybrid offloading in space-air-ground integrated networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, Austin, TX, USA, Apr. 2022, pp. 1319–1324.

[12] Y. K. Tun, Y. M. Park, T. H. T. Le, Z. Han, and C. S. Hong, "A business model for resource sharing in cell-free uavs-assisted wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 8, pp. 8839–8852, Aug. 2022.

[13] N. N. Ei, M. Alsenwi, Y. K. Tun, Z. Han, and C. S. Hong, "Energy-efficient resource allocation in multi-uav-assisted two-stage edge computing for beyond 5g networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 16 421–16 432, Feb. 2022.

[14] U. Challita and W. Saad, "Network formation in the sky: Unmanned aerial vehicles for multi-hop wireless backhauling," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Singapore, Dec. 2017.

[15] Z. Jia, M. Sheng, J. Li, D. Zhou, and Z. Han, "Joint HAP access and LEO satellite backhaul in 6G: Matching game-based approaches," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 4, pp. 1147–1159, Apr. 2021.

[16] Y. K. Tun, N. H. Tran, D. T. Ngo, S. R. Pandey, Z. Han, and C. S. Hong, "Wireless network slicing: Generalized kelly mechanism-based resource allocation," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 8, pp. 1794–1807, Aug. 2019.

[17] M. Hong, M. Razaviyayn, Z.-Q. Luo, and J.-S. Pang, "A unified algorithmic framework for block-structured optimization involving big data: With applications in machine learning and signal processing," *IEEE Signal Processing Magazine*, vol. 33, no. 1, pp. 57–77, Jan. 2016.

[18] Y. Ye, *Interior point algorithms: theory and analysis*. John Wiley & Sons, Oct. 2011.