Energy-Efficient Resource Management in UAV-Assisted Mobile Edge Computing

Yan Kyaw Tun, Yu Min Park, Nguyen H. Tran, *Senior Member, IEEE*, Walid Saad, *Fellow, IEEE*, Shashi Raj Pandey, and Choong Seon Hong, *Senior Member, IEEE*

***Abstract*—Unmanned aerial vehicles (UAVs) have been deployed to enhance the network capacity and provide services to mobile users with or without infrastructure coverage. At the same time, we have observed the exponential growth in Internet of Things (IoTs) devices and applications. However, as IoT devices have limited computation capacity and battery lifetime, it is challenging to process data locally on the devices. To this end, in this letter, a UAV-aided mobile edge computing system is proposed. The problem to jointly minimize the energy consumption at the IoT devices and the UAVs during task execution is studied by optimizing the task offloading decision, resource allocation mechanism and UAV's trajectory while considering the communication and computation latency requirements. A non-convex structure of the formulated problem is revealed and shown to be challenging to solve. To address this challenge, a block successive upper-bound minimization (BSUM) algorithm is introduced. Finally, simulation results are provided to show the efficiency of our proposed algorithm.**

***Index Terms*—Unmanned aerial vehicles (UAVs), mobile edge computing, tasks assignment, block successive upper-bound minimization (BSUM).**

I. Introduction

THE rapid proliferation of Internet of Things (IoT) devices, such as wearable devices, metering devices, traffic and other monitoring devices, is increasing the need to support computation-intensive applications, such as smart farming, face recognition, online gaming, virtual reality (VR), and augmented reality (AR) [1]. Furthermore, IoT devices are low power devices and have limited computation capacity. Therefore, it is challenging for such IoT devices to execute computational tasks locally. One promising approach to address this challenge is to deploy edge servers that bring the computation resources nearer to the devices and reduce the energy consumption of the devices by offloading their computation tasks to a local mobile edge computing (MEC) server [2]. However, in some applications, e.g., smart farming in a rural area, disaster rescue operation, and military operation, IoT devices will still be far from the mobile edge computing services and out of coverage of the cellular system which can prevent them from using MEC services. Recently, unmanned aerial vehicles (UAVs) have been widely deployed to extend the coverage area of the cellular networks and to provide network services to mobile devices where cellular infrastructures are not deployed yet [3]. In [4], authors have studied the computation efficiency maximization problem in the UAV-assisted mobile edge computing system and the work in [5] considered the task offloading strategy to its neighboring UAVs to enable a fog-like computing platform. Then, the work in [6] proposed a joint offloading and trajectory optimization design for the UAV-enabled mobile edge computing systems. In particular, the authors focused on minimizing the delays experienced by the mobile devices. However, in [6], the authors considered that each mobile device is transmitting with constant power and the computation resource allocation for task execution has been ignored. Meanwhile, in [7], the authors studied the impact of the latency on cell association. In [8], the authors considered the overall energy minimization of UAV by jointly optimizing the tasks scheduling and resource allocation. However, the energy consumption of the IoT devices has not been considered in [8].

The main contribution of this letter is to develop a new framework for minimizing the total energy consumption by optimizing task offloading decision, resource allocation mechanism, and the UAV's trajectory while satisfying the mobile devices' latency requirements. To address the formulated non-convex problem, we apply BSUM method. Simulation results demonstrate that the total energy consumption under our proposed algorithm is 10.54%, 18.49%, and 40.67% less than that of the Block Coordinate Descent (BCD) based solution, equal resource allocation and remote task execution schemes, respectively.

II. System Model

In Fig. 1, UAV-aided MEC system is pictured with a single UAV which functions as an aerial base station. The aerial base station is integrated with an edge server, and at the same time, it also serves a set $\mathcal{U}$ of U IoT devices. In order to provide computation services to the ground mobile devices, the UAV (aerial base station) is hovering above them. The UAV is capable of communicating and providing computation services to the ground mobile devices simultaneously at any given time during the UAV flight period, T. With a longer flight period T, the UAV gets more time to fly close to each ground device, which leads to achieving a better air-to-ground link. It is

Manuscript received July 29, 2020; revised September 8, 2020; accepted September 8, 2020. Date of publication September 23, 2020; date of current version January 8, 2021. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-01287, Evolvable Deep Learning Model Generation Platform for Edge Computing). The associate editor coordinating the review of this letter and approving it for publication was H. Ghazzai. *(Corresponding author: Choong Seon Hong.)*

Yan Kyaw Tun, Yu Min Park, Shashi Raj Pandey, and Choong Seon Hong are with the Department of Computer Science and Engineering, Kyung Hee University, Yongin 17104, South Korea (e-mail: ykyawtun7@khu.ac.kr; yumin0906@khu.ac.kr; shashiraj@khu.ac.kr; cshong@khu.ac.kr).

Nguyen H. Tran is with the School of Information Technologies, The University of Sydney, Sydney, NSW 2006, Australia (e-mail: nguyen.tran@sydney.edu.au).

Walid Saad is with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061 USA, and also with the Department of Computer Engineering, Kyung Hee University, Yongin 17104, South Korea (e-mail: walids@vt.edu).

Digital Object Identifier 10.1109/LCOMM.2020.3026033

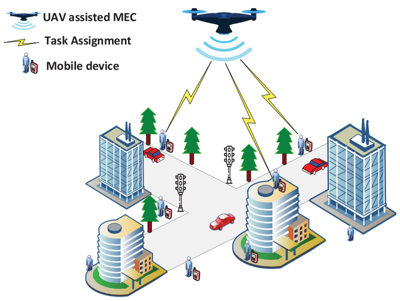

Fig. 1. Illustration of our system model.

assumed that the horizontal coordinate of each ground device is known ahead of time and fixed $\boldsymbol{o}_u = [x_u, y_u]^T, \forall u \in \mathcal{U}$. The UAV is flying at a fixed altitude H with time-varying horizontal coordinates $\boldsymbol{c}(t) = [x(t), y(t), H]^T, 0 \leq t \leq T$. Then, we discretize the UAV flight period T into a set $\mathcal{N}$ of N equally-spaced time slots with the length of each time slot set to $L = \frac{T}{N}$. Moreover, to provide the computation services to the ground mobile devices periodically and to recharge the UAV power, we assume that the UAV needs to return to its initial location at the end of the flight period, i.e., $\boldsymbol{c}(1) = \boldsymbol{c}(N)$. The maximum UAV's speed is denoted as V. Then, we have the following constraint which ensures that the speed of the UAV is below its maximum speed at each time slot:

$$\frac{||\boldsymbol{c}(n+1) - \boldsymbol{c}(n)||}{L} \leq V, \quad \forall n \in \mathcal{N}. \tag{1}$$

Moreover, we can see that LV is the maximum distance that UAV can be traveled by the UAV in each slot. Then, the energy consumption of the UAV for traveling from one location to another is related to the UAV's speed. In this context, the energy consumption due to the UAV flight at each time slot which is described in [9] as follows:

$$E^{\text{fly}}(n) = k\left(\frac{||\boldsymbol{c}(n+1) - \boldsymbol{c}(n))||}{L}\right), \quad \forall n \in \mathcal{N}, \tag{2}$$

where $k = 0.5M$ and M is the UAV's payload. Furthermore, the distance from the UAV to ground device u in time slot n is assumed to be a constant given by:

$$d_u(n) = \sqrt{H^2 + ||\boldsymbol{c}(n) - \boldsymbol{o}_u||^2}, \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}. \tag{3}$$

In this work, we consider a quasi-static block fading channel model between mobile devices and UAV. It means that the channel remains unchanged within the block and it may change over different blocks. Moreover, according to [10], there will be strong LoS links between ground mobile devices and UAV, and small-scale fading due to scattering when the UAV is at a sufficiently high altitude. Under the block fading channel model, we can model the channel coefficient of the device u at time slot n as $h_u(n) = \sqrt{\zeta_u(n)}\rho_u(n)$, where $\zeta_u(n)$ is the average channel power gain, and $\rho_u(n)$ is the small-scale channel coefficient. Then, the average channel power gain $\zeta_u(n)$ can be expressed as follows:

$$\zeta_u(n) = \zeta_0 d_u^{-\theta}(n) = \frac{\zeta_0}{(H^2 + ||\boldsymbol{c}(n) - \boldsymbol{o}_u||^2)^{\theta/2}}, \tag{4}$$

where ζ_0 is the average channel power gain at the reference distance $d_0 = 1$ m, and θ is the path loss exponent. Moreover, $\boldsymbol{c}(n)$ and $\boldsymbol{o}_u$ represent, respectively, the location of the UAV and device u at time slot n. Then, due to the existence of strong LoS link, we can model small-scale fading by using Rician distribution (i.e., Rician fading) and it is as follows:

$$\rho_u(n) = \sqrt{\frac{K_u(n)}{K_u(n)+1}}\rho + \sqrt{\frac{1}{K_u(n)+1}}\hat{\rho}, \tag{5}$$

where ρ represents the deterministic LoS component with $|\rho| = 1$, $\hat{\rho}$ denotes the randomly scattered component which is assumed as a zero-mean unit-variance circularly symmetric complex Gaussian (CSCG) random variable, and $K_u(n)$ is the Rician factor.

A. Local Computing Model

Each mobile device has a task to be executed at every time slot n denoted by a tuple $\{\beta_u, a_u(n), T_u(n)\}$, where β_u is the required CPU cycles to compute 1-bit of input data, $a_u(n)$ is the total input data size and $T_u(n)$ is the tolerable amount of time to complete the task. Since the mobile devices are constrained by their power and computing capacity, it is impossible to execute all of their tasks locally. One promising solution is to assign a fraction of mobile device's task to the UAV which is attached with edge server on board and has more powerful computation capability. Let $l_u(n)$ and $(a_u(n) - l_u(n))$ be the fractions of the task executed remotely at UAV and locally at mobile device u, respectively. For every time slot n, the local computation latency/delay of device u is calculated by [6] as follows:

$$t_u^l(n) = \frac{\beta_u(a_u(n) - l_u(n))}{f_u^l}, \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{6}$$

where f_u^l is the maximum computation capacity (i.e., cycles/s) of user u. Then, the local energy consumption of device u at time slot n is given by [6] as follows:

$$E_u^l(n) = w(f_u^l)^2\beta_u(a_u(n) - l_u(n)), \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{7}$$

where w is a constant which depends on the chip architecture of the mobile device and we set $w = 5 \times 10^{-27}$ as defined in [6].

B. UAV-Aided Edge Computing Model

A fraction of the system bandwidth is allocated to each mobile device when its task is assigned to the UAV-assisted MEC server. We now introduce variables B and $\alpha_u(n)$ to denote, respectively, the total system bandwidth and the fraction of the system bandwidth allocated to device u at time slot n. Then, the instantaneous channel capacity of the device u at time slot n will be:

$$C_u(n) = \alpha_u(n)B\log_2\left(1 + \frac{p_u(n)|h_u(n)|^2}{\sigma^2}\right), \quad \forall u,\ \forall n, \tag{8}$$

where $p_u(n)$ is the transmit power of the user u at time slot n, and σ^2 is the additive Gaussian noise power. Then, the outage probability between the device u and UAV at time slot n is as follows:

$$\begin{aligned} P_u^{\text{out}}(n) &= \mathbb{P}\left(C_u(n) < R_u(n)\right) \\ &= F\left(\frac{\sigma^2(2^{R_u(n)/(\alpha_u(n)B)} - 1)}{\zeta_u(n)p_u(n)}\right), \end{aligned} \tag{9}$$

where $F(.)$ is the cumulative distribution function (CDF) of the random variable $|\rho_u(n)|^2$, and $P_u^{\text{out}}(n)$ is the non-decreasing function with respect to $R_u(n)$. Moreover, the outage probability $P_u^{\text{out}}(n) = \epsilon$, where ϵ is the maximum tolerable outage probability. Finally, the outage-aware achievable data rate of the device u at time slot n can be expressed as follows:

$$R_u(n)=\alpha_u(n)B\log_2\left(1+\frac{F^{-1}(\epsilon)p_u(n)\zeta_0}{\sigma^2(H^2+||\boldsymbol{c}(n)-\boldsymbol{o}_u||^2)^{\theta/2}}\right), \tag{10}$$

where $F^{-1}(.)$ is the complementary cumulative distribution function (i.e., the inverse of $F(.)$).

We can then find the uplink transmission time of mobile device u when assigning the fraction of the task $l_u(n)$ to the UAV at time slot n which is described in [6] as follows:

$$t_u^{\text{up}}(n) = \frac{l_u(n)}{R_u(n)}, \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}. \tag{11}$$

Moreover, the energy consumption for uplink transmission (from the mobile device to the UAV) when the UAV is assigned a task fraction $l_u(n)$ at time slot n which is described in [6] as follows:

$$E_u^{\text{up}}(n) = \frac{p_u(n)l_u(n)}{R_u(n)}, \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}. \tag{12}$$

Then, the computation time/latency needed to execute the assigned task of user u at the UAV at time slot n which is described in [6] as follows:

$$t_u^{\text{comp}}(n) = \frac{\beta_u l_u(n)}{f_u^C(n)}, \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{13}$$

where $f_u^C(n)$ is the computation capacity (i.e., cycles/s) of the UAV-mounted server that is allocated to device u at time slot n. Then, the energy consumed by the UAV for executing the fraction of the task related to device u at time slot n which is described in [6] as follows:

$$E_u^{\text{exe}}(n) = q(f_u^C)^2\beta_u l_u(n), \quad \forall n \in \mathcal{N}, \tag{14}$$

where q is a constant which depends on the chip architecture of the edge server at the UAV, and we set $q = 5 \times 10^{-27}$ as defined in [6]. It is found that the size of the output data after the execution of the assigned task at the MEC server of the UAV is significantly smaller than that of the assigned task. On the other hand, the bandwidth allocated for the downlink transmission is considerably larger than that of the uplink transmission. Hence, the delay/latency and the energy consumption for the downlink transmission in sending the output data back to the device $u \in \mathcal{U}$ are considered negligible. Therefore, in this research, the downlink transmission from the UAV to the ground devices is omitted.

III. Energy Efficient Task Assignment and Resource Allocation

Given the model of Section II, our objective is to research the problem of joint task assignment and resource allocation in a UAV-aided mobile edge computing system with the goal of minimizing the energy consumption of both ground mobile devices and the UAV.

To the best of our knowledge, our work is the first to consider the energy minimization of both mobile devices and their serving UAV, by jointly optimizing the UAV's trajectory, communication and computation resource allocation, and task assignment. We can formally pose this problem as follows:

$$\min_{\boldsymbol{c},\boldsymbol{l},\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{f}} \left(\sum_{n=1}^{N}\sum_{u=1}^{U} E_u^l(n) + E_u^{\text{up}}(n)\right) + \sum_{n=1}^{N} E^{\text{fly}}(n) + \sum_{n=1}^{N}\sum_{u=1}^{U} E_u^{\text{exe}}(n) \tag{15}$$

$$\text{s.t. } t_u^{\text{up}}(n) + t_u^{\text{comp}}(n) \le T_u(n), \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{15a}$$

$$t_u^l(n) \le T_u(n), \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{15b}$$

$$l_u(n) \le a_u(n), \quad \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{15c}$$

$$\sum_{u=1}^{U} f_u^C(n) \le f^C(n), \quad \forall n \in \mathcal{N}, \tag{15d}$$

$$0 \le p_u(n) \le p_u^{\max}(n), \quad \forall n \in \mathcal{N},\ \forall u \in \mathcal{U}, \tag{15e}$$

$$\sum_{u=1}^{U} \alpha_u(n) \le 1, \quad 0 \le \alpha_u(n) \le 1,\ \forall u \in \mathcal{U},\ \forall n \in \mathcal{N}, \tag{15f}$$

$$\frac{||\mathbf{c}(n+1)-\mathbf{c}(n)||}{L} \le V, \quad \forall n \in \mathcal{N}, \tag{15g}$$

$$\boldsymbol{c}(1) = \boldsymbol{c}(N), \tag{15h}$$

where (15a) and (15b) represent the latency constraint of the task of each mobile device at each time slot, (15c) ensures that the fraction of the task assigned by device u to the UAV is always smaller than the total input data size of the task of the device. Then, (15d) and (15e) represent, respectively, the computation capacity constraint of the UAV and the power constraint of each ground mobile device. (15f) captures the fractions of the total bandwidth B allocated to the devices at each time slot n. Finally, (15g) is the velocity constraint for the UAV and (15h) ensures that UAV will return back to the initial location at the end of the flight period.

It is challenging to solve the optimization problem in (15) by using convex optimization techniques due to the coupling between the high number of decision variables in the objective function, the nonlinear constraints, and its non-convex structure. Therefore, we apply BSUM algorithm to solve (15) [11]. BSUM continuously minimizes a sequence of tight upper bound of the objective function and sequentially updates the variables $\boldsymbol{c}$, $\boldsymbol{l}$, $\boldsymbol{\alpha}$, $\boldsymbol{p}$, and $\boldsymbol{f}$. Moreover, BSUM is guaranteed to converge to the set of stationary points of the objective function in (15). To apply the BSUM method, we can rewrite the optimization problem expressed in (15) more concisely as follows:

$$\min_{\substack{\boldsymbol{c}\in\mathcal{C},\boldsymbol{l}\in\mathcal{L},\boldsymbol{\alpha}\in\alpha,\\ \boldsymbol{p}\in\mathcal{P},\boldsymbol{f}\in\mathcal{F}}} \mathcal{O}(\boldsymbol{c},\boldsymbol{l},\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{f}) \tag{16}$$

where $\mathcal{O}(\boldsymbol{c},\boldsymbol{l},\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{f}) = \left(\sum_{n=1}^{N}\sum_{u=1}^{U} E_u^l(n) + E_u^{\text{up}}(n)\right) + \sum_{n=1}^{N} E^{\text{fly}}(n) + \sum_{n=1}^{N}\sum_{u=1}^{U} E_u^{\text{exe}}(n)$. Furthermore,

$$\mathcal{C} \triangleq \{\boldsymbol{c} : t_u^{\text{up}}(n) + t_u^{\text{comp}}(n) \le T_u(n), \frac{||\mathbf{c}(n+1)-\mathbf{c}(n)||}{L} \le V, \forall u \in \mathcal{U}, \forall n \in \mathcal{N}\},$$

$$\mathcal{L} \triangleq \{\boldsymbol{l} : t_u^{\text{up}}(n) + t_u^{\text{comp}}(n) \le T_u(n), t_u^l(n) \le T_u(n), l_u(n) \le a_u(n), \forall u \in \mathcal{U}, \forall n \in \mathcal{N}\},$$

$$\alpha \triangleq \{\boldsymbol{\alpha} : t_u^{\text{up}}(n) + t_u^{\text{comp}}(n) \leq T_u(n), \sum_{u=1}^{U} \alpha_u(n) \leq 1, 0 \leq \alpha_u(n) \leq 1, \forall u \in \mathcal{U}, \forall n \in \mathcal{N}\},$$

$$\mathcal{P} \triangleq \{\boldsymbol{p} : t_u^{\text{up}}(n) + t_u^{\text{comp}}(n) \leq T_u(n), 0 \leq p_u(n) \leq p_u^{\max}(n), \forall n, \forall u \in \mathcal{U}\},$$

$$\mathcal{F} \triangleq \{\boldsymbol{f} : t_u^{\text{up}}(n) + t_u^{\text{comp}}(n) \leq T_u(n), \sum_{u=1}^{U} f_u^C(n) \leq f^C(n), \forall u, \forall n \in \mathcal{N}\},$$

are the feasible sets of $\boldsymbol{c}$, $\boldsymbol{l}$, $\boldsymbol{\alpha}$, $\boldsymbol{p}$, and $\boldsymbol{f}$, respectively. At each iteration k, $\forall i \in \mathcal{I}^k$ where $\mathcal{I}$ is the set of indexes, we define the proximal upper-bound function $\mathcal{O}_i$ of the objective function in (16) and adding the quadratic penalty term helps to keep the proximal upper-bound function convex, as follows:

$$\mathcal{O}_i(\boldsymbol{c}_i; \boldsymbol{c}^k, \boldsymbol{l}^k, \boldsymbol{\alpha}^k, \boldsymbol{p}^k, \boldsymbol{f}^k) = \mathcal{O}(\boldsymbol{c}_i; \tilde{\boldsymbol{c}}, \tilde{\boldsymbol{l}}, \tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{p}}, \tilde{\boldsymbol{f}}) + \frac{\mu_i}{2} \|(\boldsymbol{c}_i - \tilde{\boldsymbol{c}})\|^2, \tag{17}$$

where μ_i is a positive penalty parameter, and it can be deployed for the other vectors of variables $\boldsymbol{l}_i$, $\boldsymbol{\alpha}_i$, $\boldsymbol{p}_i$, and $\boldsymbol{f}_i$, respectively. Furthermore, the proximal upper-bound function in (17) has a unique minimizer vectors $\tilde{\boldsymbol{c}}$, $\tilde{\boldsymbol{l}}$, $\tilde{\boldsymbol{\alpha}}$, $\tilde{\boldsymbol{p}}$, and $\tilde{\boldsymbol{f}}$ with respect to $\boldsymbol{c}_i$, $\boldsymbol{l}_i$, $\boldsymbol{\alpha}_i$, $\boldsymbol{p}_i$, and $\boldsymbol{f}_i$ at each iteration k, which are considered as the solution of the previous iteration $(k-1)$. Then, the solution at each iteration $(k+1)$ can be updated by solving the following subproblems:

$$\boldsymbol{c}_i^{(k+1)} \in \min_{\boldsymbol{c}_i \in \mathcal{C}} \mathcal{O}_i\Big(\boldsymbol{c}_i; \boldsymbol{c}^{(k)}, \boldsymbol{l}^{(k)}, \boldsymbol{\alpha}^{(k)}, \boldsymbol{p}^{(k)}, \boldsymbol{f}^{(k)}\Big), \tag{18}$$

$$\boldsymbol{l}_i^{(k+1)} \in \min_{\boldsymbol{l}_i \in \mathcal{L}} \mathcal{O}_i\Big(\boldsymbol{l}_i; \boldsymbol{l}^{(k)}, \boldsymbol{c}^{(k+1)}, \boldsymbol{\alpha}^{(k)}, \boldsymbol{p}^{(k)}, \boldsymbol{f}^{(k)}\Big), \tag{19}$$

$$\boldsymbol{\alpha}_i^{(k+1)} \in \min_{\boldsymbol{\alpha}_i \in \alpha} \mathcal{O}_i\Big(\boldsymbol{\alpha}_i; \boldsymbol{\alpha}^k, \boldsymbol{c}^{(k+1)}, \boldsymbol{l}^{(k+1)}, \boldsymbol{p}^{(k)}, \boldsymbol{f}^{(k)}\Big), \tag{20}$$

$$\boldsymbol{p}_i^{(k+1)} \in \min_{\boldsymbol{p}_i \in \mathcal{P}} \mathcal{O}_i\Big(\boldsymbol{p}_i; \boldsymbol{p}^{(k)}, \boldsymbol{c}^{(k+1)}, \boldsymbol{l}^{(k+1)}, \boldsymbol{\alpha}^{(k+1)}, \boldsymbol{f}^{(k)}\Big), \tag{21}$$

$$\boldsymbol{f}_i^{(k+1)} \in \min_{\boldsymbol{f}_i \in \mathcal{F}} \mathcal{O}_i\Big(\boldsymbol{f}_i; \boldsymbol{f}^{(k)}, \boldsymbol{c}^{(k+1)}, \boldsymbol{l}^{(k+1)}, \boldsymbol{\alpha}^{(k+1)}, \boldsymbol{p}^{(k+1)}\Big) \tag{22}$$

Additionally, the subproblems in (18)-(20) can be solved by using BSUM algorithm.

IV. Simulation Results

In our simulation, we consider that the UAV attached with the MEC server is flying at the fixed altitude $H = 20$ m with the maximum speed $V = 7$ m/s over a coverage area of (70×70) m^2. $U = 4$ mobile devices, are randomly positioned in that coverage area. The UAV begins and finishes the flight at the same coordinate, $\boldsymbol{c}(1) = \boldsymbol{c}(N) = (0, 0)$. The total flight time and the weight of the UAV are 40 s and 6 kg, respectively. The maximum computation capacity of the MEC server at the UAV is 1 GHz and the maximum computing power of the mobile devices is randomly generated between [0.1, 0.72] MHz. The input data size of the computation task of mobile devices and the required CPU cycles to execute one bit of input data are randomly generated between $[0.4 \times 10^5,\ 1 \times 10^5]$ bits and [5, 25] cycles, respectively. Then, the tolerable latency of the mobile devices is between [5, 20] s. The maximum available

Algorithm 1 BSUM Algorithm for an Energy-Efficient Resource Management in UAV-Assisted Mobile Edge Computing

1: **Initialization:** Set $k = 0$, $\epsilon_1 > 0$, and find initial feasible solutions $(\boldsymbol{c}^{(0)}, \boldsymbol{l}^{(0)}, \boldsymbol{\alpha}^{(0)}, \boldsymbol{p}^{(0)}, \boldsymbol{f}^{(0)})$;
2: **repeat**
3: Choose index set $\mathcal{I}^k$;
4: Let $\boldsymbol{c}_i^{(k+1)} \in \min_{\boldsymbol{c}_i \in \mathcal{C}} \mathcal{O}_i\big(\boldsymbol{c}_i; \boldsymbol{c}^{(k)}, \boldsymbol{l}^{(k)}, \boldsymbol{\alpha}^{(k)}, \boldsymbol{p}^{(k)}, \boldsymbol{f}^{(k)}\big)$;
5: Set $\boldsymbol{c}_j^{(k+1)} = \boldsymbol{c}_j^k, \forall j \notin \mathcal{I}^k$;
6: Find $\boldsymbol{l}_i^{(k+1)}$, $\boldsymbol{\alpha}_i^{(k+1)}$, $\boldsymbol{p}_i^{(k+1)}$, and $\boldsymbol{f}_i^{(k+1)}$ by solving (19), (20), (21), and (22);
7: $k = k + 1$;
8: **until** $\| \frac{\mathcal{O}_i^{(k)} - \mathcal{O}_i^{(k+1)}}{\mathcal{O}_i^{(k)}} \| \leq \epsilon_1$
9: Then, set $\big(\boldsymbol{c}_i^{(k+1)}, \boldsymbol{l}_i^{(k+1)}, \boldsymbol{\alpha}_i^{(k+1)}, \boldsymbol{p}_i^{(k+1)}, \boldsymbol{f}_i^{(k+1)}\big)$ as the desired solution.

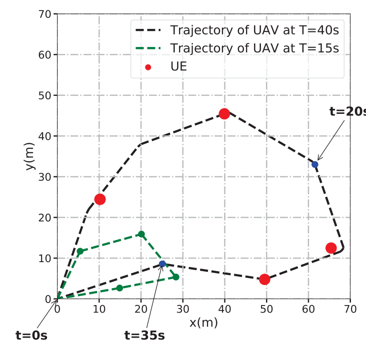

Fig. 2. Trajectories of UAV under different flight period T.

bandwidth is 30 MHz and the additive Gaussian noise power is -174 dBm/Hz. The maximum transmit power of the mobile devices is 100 mW.

In Fig. 2, we show the optimal UAV trajectory under our proposed BSUM algorithm, where the total UAV flight time, $T = 40$ s, is equally divided into 13 time slots with a flight duration of 5 s per slot. Then, we adjust the UAV flight time to 15 s and find the optimal UAV trajectory under our proposed algorithm. As a result, from Fig. 2, we note that when the flight time is sufficiently large, the UAV will fly close to the mobile devices.

In Fig. 3a, we demonstrated the total energy consumption of UAV such as flying energy and edge computing energy, and the energy consumption of IoT devices which includes uplink transmission energy and local computing energy. Moreover, we also compare the energy consumption of both UAV and IoT devices under our proposed algorithm with the Block Coordinate Descent (BCD) method and other two baseline schemes: 1) Equal resource allocation where the wireless bandwidth and the computing capacity of the MEC server deployed at the UAV are equally allocated to all devices, and 2) Remote task execution where mobile devices offload all of their computation tasks to the server at the UAV and execute tasks remotely. From the figure, it is clear that the energy consumption of UAV and IoT devices under our proposed algorithm is less than that of the others schemes. Fig. 3b presents the relationship between the energy consumption and

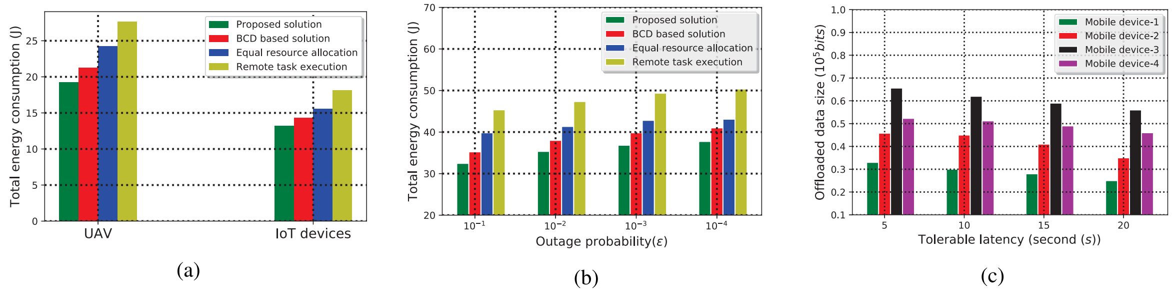

Fig. 3. 3(a) Shows energy consumption of UAV and IoT devices. 3(b) Shows energy consumption under different outage probabilities. 3(c) Shows offloaded data size of the task under different tolerable latency.

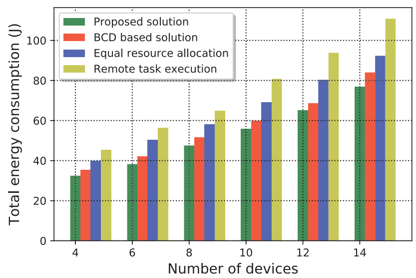

Fig. 4. Energy consumption under different number of mobile devices.

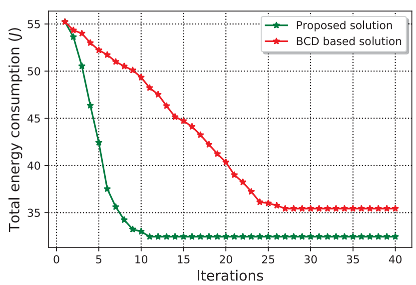

Fig. 5. Convergence of the proposed solution.

the tolerable outage probability. In this regard, the devices increase their transmission power to achieve required high data rate when the outage probability is higher. Thus, we observe the increase in energy consumption when we tighten the outage probability, as in Fig. 3b. Fig. 3c presents the impact of tolerable latency on the offloaded data size of each mobile device to the MEC server deployed at the UAV. As observed in Fig. 3c, each mobile device offloads less data to the MEC server when the tolerable latency increases, i.e., it has a sufficient amount of time to execute its computation task locally. However, the drawback is that it can increase the energy consumption of the device. In Fig. 4, the total energy consumption of the network (i.e., energy consumption of both mobile devices and UAV) for a different number of mobile devices is described. From Fig. 4, we observe that our proposed scheme outperforms the other baseline schemes. The performance gap between our proposed scheme and other schemes gets higher when the number of users increases. Therefore, our proposed scheme is more efficient when there is a large number of users in the system. Finally, Fig. 5 demonstrates our proposed solution converges to the sub-optimal solution within a few iterations. Furthermore, we also notice from the figure that our proposed solution converges faster than the BCD method.

V. CONCLUSION

In this letter, we have studied the problems of energy-efficient UAV trajectory optimization, resource allocation, and task offloading in a UAV-assisted mobile edge computing system where we aimed at minimizing not only the energy consumption of mobile devices but also UAV's propulsion and computing power. We have shown that the studied problem exhibit a non-convex structure, and, thus, it is challenging to solve by using traditional convex optimization techniques. To address this issue, we have introduced the BSUM algorithm, which is a powerful tool for non-convex and non-smooth problem. Finally, we presented the numerical results to show the efficiency of the proposed solution approach where it was clear that our proposed algorithm outperforms other baseline algorithms.

REFERENCES

[1] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May 2020.
[2] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, Mar. 2017.
[3] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2334–2360, 3rd Quart., 2019.
[4] J. Zhang *et al.*, "Computation-efficient offloading and trajectory scheduling for multi-uav assisted mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 2114–2125, Feb. 2019.
[5] G. Lee, W. Saad, and M. Bennis, "Online optimization for UAV-assisted distributed fog computing in smart factories of industry 4.0," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.
[6] Q. Hu, Y. Cai, G. Yu, Z. Qin, M. Zhao, and G. Y. Li, "Joint offloading and trajectory design for UAV-enabled mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1879–1892, Apr. 2019.
[7] M. Mozaffari, A. T. Z. Kasgari, W. Saad, M. Bennis, and M. Debbah, "Beyond 5G with UAVs: Foundations of a 3D wireless cellular network," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 357–372, Jan. 2019.
[8] Y. Du, K. Wang, K. Yang, and G. Zhang, "Energy-efficient resource allocation in UAV based MEC system for IoT devices," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.
[9] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2049–2063, Mar. 2018.
[10] W. Khawaja, I. Guvenc, D. W. Matolak, U.-C. Fiebig, and N. Schneckenburger, "A survey of air-to-ground propagation channel modeling for unmanned aerial vehicles," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2361–2391, 3rd Quart., 2019.
[11] 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 Process. Mag.*, vol. 33, no. 1, pp. 57–77, Jan. 2016.