

Joint Task Offloading and Resource Allocation in UAV-Enabled Mobile Edge Computing

Zhe Yu, *Student Member, IEEE*, Yanmin Gong^{ID}, *Member, IEEE*, Shimin Gong^{ID}, *Member, IEEE*,
and Yuanxiong Guo^{ID}, *Senior Member, IEEE*

Abstract—Mobile edge computing (MEC) is an emerging technology to support resource-intensive yet delay-sensitive applications using small cloud-computing platforms deployed at the mobile network edges. However, the existing MEC techniques are not applicable to the situation where the number of mobile users increases explosively or the network facilities are sparsely distributed. In view of this insufficiency, unmanned aerial vehicles (UAVs) have been employed to improve the connectivity of ground Internet of Things (IoT) devices due to their high altitude. This article proposes an innovative UAV-enabled MEC system involving the interactions among IoT devices, UAV, and edge clouds (ECs). The system deploys and operates a UAV properly to facilitate the MEC service provisioning to a set of IoT devices in regions where the existing ECs cannot be accessible to IoT devices due to terrestrial signal blockage or shadowing. The UAV and ECs in the system collaboratively provide MEC services to the IoT devices. For optimal service provisioning in this system, we formulate an optimization problem aiming at minimizing the weighted sum of the service delay of all IoT devices and UAV energy consumption by jointly optimizing UAV position, communication and computing resource allocation, and task splitting decisions. However, the resulting optimization problem is highly nonconvex and thus, difficult to solve optimally. To tackle this problem, we develop an efficient algorithm based on the successive convex approximation to obtain suboptimal solutions. Numerical experiments demonstrate that our proposed collaborative UAV-EC offloading scheme largely outperforms baseline schemes that solely rely on UAV or ECs for MEC in IoT.

Index Terms—Mobile edge computing (MEC), resource management, successive convex approximation, unmanned aerial vehicles (UAVs).

I. INTRODUCTION

AS THE number of wireless connected devices continues to grow vastly and rapidly, an enormous amounts

of data are collected from these devices at an exponential rate [1] and need to be transported from place to place for intelligent decision making, which has generated tremendous burden on our wireless communication infrastructure with the limited radio spectrum. It is estimated that 25 billion Internet of Things (IoT) devices will be in use by 2025 [2], and such multitudes of wireless connected devices are enabling many compelling new applications, such as real-time video analytics [3], [4], augmented/virtual reality [5], and smart cities [6], which are computation intensive and delay sensitive and rely on our ability to quickly process the data and extract useful information, precluding the traditional cloud-based data processing paradigm [7].

Mobile edge computing (MEC), in which computing and storage resources are placed at the mobile network edges (e.g., cellular base stations or WiFi access points) [8]–[10], has emerged as a prospective solution to resolve the network latency issue by pushing the frontier of data and services away from centralized cloud to the edge of the network, thereby enabling data analytics and functional operation in the proximity to the data sources. By moving resources to the network edge, close to where the data are being generated and acted upon, MEC can bring many benefits to users, such as lower service latency, reduced network congestion, and better service quality. Meanwhile, resource management becomes a key problem in MEC due to the much limited resources compared to remote clouds and the tight coupling of communication and computing. There has been substantial research on MEC resource management with the goal of optimizing system latency [11]–[14], energy consumption [15]–[17], and overall cost of system latency and/or energy consumption [18]–[21]. However, all of these studies assume wired or dedicated wireless connections with sufficient bandwidth among distributed edge resources deployed in a fixed fashion. Particularly, the existing MEC techniques are not applicable to the situation where the number of mobile users (MUs) increases explosively or the network facilities are sparsely distributed [22]. In view of this insufficiency, wireless networks enabled by unmanned aerial vehicles (UAVs) have recently been proposed as a promising solution to improve the connectivity of ground IoT devices.

UAVs, especially, low-cost quadcopters, are undergoing an explosive growth and a major regulation relaxation nowadays and have been widely used in civilian domains, such as traffic monitoring [23], public safety [24], search and rescue [25], and reconnaissance over disaster rescue and recovery [26]. UAVs

Manuscript received September 11, 2019; revised December 18, 2019; accepted December 28, 2019. Date of publication January 10, 2020; date of current version April 14, 2020. The work of Yanmin Gong was supported in part by the U.S. National Science Foundation under Grant CNS-1850523. (Corresponding author: Yuanxiong Guo.)

Zhe Yu is with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA (e-mail: zhe.yu@okstate.edu).

Yanmin Gong is with the Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: yanmin.gong@utsa.edu).

Shimin Gong is with the School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou 510275, China (e-mail: gongshm5@mail.sysu.edu.cn).

Yuanxiong Guo is with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: yuanxiong.guo@utsa.edu).

Digital Object Identifier 10.1109/JIOT.2020.2965898

2327-4662 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See <https://www.ieee.org/publications/rights/index.html> for more information.

not only provide extended coverage over wide geographical areas but also possess unique characteristics like fast deployment, easy programmability, and high scalability. Various payloads, such as IoT sensors (including cameras), miniaturized base stations, and embedded computing modules can be mounted on UAVs to enable different sensing, communication, and computing tasks [27], [28]. In particular, reliable and cost-effective wireless communication solutions for multitudes of real-world scenarios can be offered by UAVs if properly deployed and operated [28]. UAVs can act as wireless relays or aerial base stations for improving connectivity and extending coverage of ground wireless devices since the high altitude of UAV enables wireless devices to effectively establish line-of-sight (LoS) communication links thus mitigating the potential signal blockage and shadowing.

However, most prior works in the area of the UAV-enabled wireless networks (e.g., [29]–[32]) ignore the computing capability provided by UAVs and mainly focus on their communication aspect, and only a very few recent studies [22], [33]–[36] start to consider computing with UAVs' onboard resources. Hu *et al.* [22], Jeong *et al.* [33], and Zhou *et al.* [34] only considered communication and computation interactions between two types of entities where ground MUs offload the tasks to UAV for computation. Asheralieva and Niyato [35] proposed a game-theoretic and reinforcement learning approach in investigating the cooperation among UAVs and ground base stations. Hu *et al.* [36] studied a new UAV-enabled MEC system with interactions among a UAV, a set of ground user equipments, and an access point. To the best of our knowledge, UAV-enabled MEC systems involving MUs, UAVs, and edge clouds (ECs) with UAV-EC collaboration have not been studied.

In this article, we envision an innovative UAV-enabled MEC system where IoT devices offload computing tasks to ECs outside their communication range with the assistance of UAV, which are endowed with computing capability, to take the benefits of collaboration among UAV and ECs. Specifically, we consider the regions where the terrestrial wireless communication between IoT devices and ground cellular base stations or WiFi access points cannot be effectively established due to signal blockage and shadowing. Therefore, a UAV is deployed and operated to facilitate MEC service provisioning to a set of stationary IoT devices in such regions. The IoT devices perform some sensing tasks and need to process the generated data quickly. We assume that the sensing data analysis is not performed locally due to limited onboard communication, computing, and storage (CCS) resources but we seek to utilize those from the UAV and existing ground ECs. The UAV, equipped with miniaturized base stations and embedded computing modules, is placed properly to collect the generated sensing data from IoT devices and then, can further forward the computation tasks to more resourceful ground ECs nearby. We formulate the IoT task offloading process as a nonconvex optimization problem aiming at minimizing the weighted sum of the service delay of all IoT devices consisting of task offloading delay and computation delay, and UAV energy consumption consisting of transmission energy

and computation energy consumption, by jointly optimizing the task splitting decisions, UAV placement, communication bandwidth allocation, and computation resource allocation at the UAV and ECs.

The above-formulated optimization problem is challenging to solve due to the nonconvex objective function and constraints. To tackle that challenge, we implement an efficient algorithm by means of successive convex approximation [37], [38]. The basic idea of the proposed algorithm is to compute a suboptimal solution of the original nonconvex problem by solving a sequence of convex subproblems where the nonconvex objective function and constraints are replaced by suitable convex approximants. We first convert the nonconvex objective function and constraints into suitable convex approximants by introducing the initial feasible solutions, while the local first-order behavior of the original nonconvex problem is preserved. Then, we iteratively compute the local optimum of the resulting convex problem by updating the initial feasible solutions until a stationary solution of the original nonconvex problem is found. The convergence of the proposed algorithm is guaranteed if the step-size rule and termination criterion are properly chosen.

The main contributions of this article are summarized as follows.

- 1) We propose a novel UAV-enabled MEC system where a UAV is deployed to facilitate the provisioning of MEC services to IoT devices that cannot directly access ECs on the ground due to terrestrial signal blockage and shadowing.
- 2) Considering the stringent quality-of-service requirement of MEC services and the limited battery size of UAV, we formulate the joint IoT task offloading and UAV placement under the proposed system as an optimization problem with the goal of minimizing the service delay of IoT devices and maximizing the energy efficiency of UAV.
- 3) Given the nonconvexity of the formulated optimization problem, we reformulate it into tractable one using successive convex approximation, and then develop an efficient algorithm to find the suboptimal approximate solutions to the problem.
- 4) We conduct extensive simulations to evaluate the performance of our proposed collaborative UAV-EC scheme. Numerical experiments demonstrate that our proposed collaborative UAV-EC offloading scheme largely outperforms baseline schemes that solely rely on UAV or ECs for MEC in IoT.

The remainder of this article is organized as follows. Related work is reviewed in Section II. In Section III, we describe the system model and then formulate the optimal IoT task offloading processes as a nonconvex optimization problem. In Section IV, we reformulate the original problem as an approximated convex optimization problem and then solve it by means of successive convex approximation. The simulation results based on real-world traces are presented in Section V. Finally, the conclusion is given in Section VI.

II. RELATED WORK

In this section, we review the prior works most relevant to our article from two aspects: 1) resource management in MEC and 2) UAV-enabled MEC networks.

A. Resource Management in MEC

There is a rich literature on resource management in MEC that aims at optimizing system latency [11]–[14], energy consumption [15]–[17], and overall cost of system latency and/or energy consumption [18]–[21]. The tradeoff problem is studied in [11] for computing networks with fog node cooperation aiming at minimizing the response time of fog nodes under a given power efficiency constraint. Xu *et al.* [12] studied the joint service caching and task offloading problem in the dense network aiming at minimizing computation latency while keeping the total computation energy consumption low. Chen and Hao [13] investigated the MEC task offloading problem in the software-defined ultradense network aiming at minimizing the total task duration under energy budget constraints. Ren *et al.* [14] investigated a joint communication and computation resource allocation problem under the collaboration of cloud and edge computing for minimizing the system delay of all mobile devices. Sardellitti *et al.* [15] formulated the multicell MEC task offloading problem as a joint optimization of radio and computation resources aiming at minimizing the overall users' energy consumption, while meeting latency constraints. Zhang *et al.* [16] proposed an energy-efficient offloading scheme for MEC in 5G heterogeneous networks by formulating the optimization problem with the objective of minimizing the total system energy consumption. You *et al.* [17] studied the resource allocation problem for a multiuser MEC offloading system based on TDMA and OFDMA with the objective to minimize the weighted sum of mobile energy consumption. Chen *et al.* [18] formulated a multiuser computation offloading game to study the energy-delay tradeoff problem in a mobile-edge cloud computing architecture. Chen *et al.* [19], [20] jointly optimized the offloading decisions of all users and computing access point and resource allocation aiming at minimizing the overall energy cost and the maximum delay among all users. Zhang *et al.* [21] proposed a distributed joint computation offloading and resource allocation optimization scheme in heterogeneous networks with MEC to minimize the overhead of local energy consumption and execution time cost.

However, all of these studies assume wired or dedicated wireless connections with sufficient bandwidth among distributed edge resources deployed in a fixed fashion. Particularly, the existing MEC techniques are not applicable to the situation where the number of MUs increases explosively or the network facilities are sparsely distributed [22]. In view of the above limitations, we propose to deploy and operate a UAV to assist the IoT task offloading processes in a MEC system where ECs cannot be accessible to IoT devices due to terrestrial signal blockage and shadowing.

B. UAV-Enabled MEC Networks

Extensive research efforts have been made from the academia to employ UAVs as different kinds of wireless communication platforms [39]. For instance, UAVs equipped with base stations can be flexibly deployed at specific areas to provide reliable uplink and downlink communication for ground users. They can also serve as the mobile relaying nodes to connect two or more distant users [29], [30]. Moreover, UAVs can assist with information dissemination or data collection by flying over the specific areas [31], [32]. However, prior works in the area of the UAV-enabled wireless networks ignore the computing capability provided by UAVs and mainly focus on their communication aspect, and only a very few recent studies [22], [33]–[36] start to consider computing with UAVs' onboard resources. Hu *et al.* [22] investigated joint offloading and trajectory design for a MEC system where a UAV endowed with computing capability is deployed to serve the task offloading of MUs, aiming at minimizing the sum of the maximum delay among all the users in each time slot. Jeong *et al.* [33] studied the joint optimization of path planning and bit allocation for an MEC system where a UAV-mounted cloudlet is deployed to provide offloading opportunities to MUs, aiming at minimizing the mobile energy consumption while satisfying the quality-of-service requirements of offloaded applications. Zhou *et al.* [34] formulated the computation rate maximization problem under both partial and binary task offloading schemes in a UAV-enabled MEC wireless-powered system where the UAV can simultaneously transmit energy and perform computation. However, these works only consider communication and computation interactions between two types of entities where ground MUs offload the tasks to UAV for computation. Besides, Asheralieva and Niyato [35] presented a game-theoretic and reinforcement learning framework to study the computation offloading problem in UAV-enabled MEC networks with multiple service providers where UAV-based privately owned base stations are interacting with terrestrial privately owned and operator controlled base stations. Hu *et al.* [36] considered a UAV-aided MEC system where the cellular-connected UAV is served as a mobile computing server as well as a relay to help the user equipments complete their computing tasks or further offload their tasks to the AP for computing.

To the best of our knowledge, UAV-enabled MEC systems involving MUs, UAVs, and ECs have not been studied. Different from [35] which focuses on the user's perspective, we optimize the UAV energy-efficiency and IoT task service latency from the system operator's perspective with UAV-EC collaboration. Different from [36] which focuses on a single EC, we consider the scenario where multiple ECs and the UAV collaboratively provide MEC services to the IoT devices.

III. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we first introduce the system model for the UAV-enabled MEC system. After that, we formulate an optimization problem to model the optimal UAV-enabled IoT task offloading process.

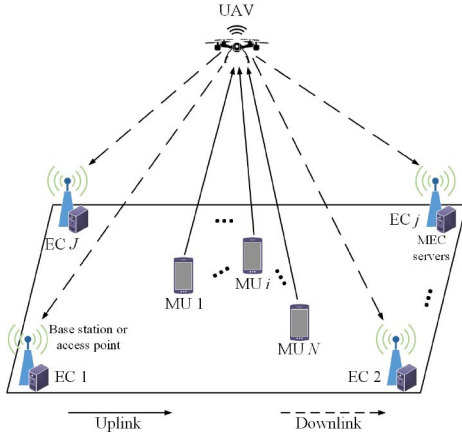


Fig. 1. Illustration of an exemplary UAV-enabled MEC system with N MUs, J ECs, and a UAV.

A. System Model

In this article, we consider the UAV-enabled MEC system as depicted in Fig. 1, which consists of a set of ground MUs¹ $i \in \mathcal{N} = \{1, 2, \dots, N\}$, a UAV, and a set of ground ECs $j \in \mathcal{J} = \{1, 2, \dots, J\}$. The UAV is deployed to facilitate the MEC service provisioning for ground MUs who cannot establish wireless communication with nearby cellular base stations or WiFi access points due to signal blockage and shadowing. In this scenario, ground-to-air (G2A) uplink communication is from MUs to the UAV while air-to-ground (A2G) downlink communication is from the UAV to ECs, which form a 3-D wireless communication network. For ease of reference, we list important notations in Table I.

We assume that the UAV is equipped with certain CCS resources but subject to the size, weight, and power (SWAP) limitations. The ECs are composed of ground MEC servers co-located with cellular base stations or WiFi access points that have more CCS resources compared to the UAV and MUs. Each MU i has periodical computation-intensive tasks to perform, which are modeled as a triplet $\mathcal{W}_i = \langle L_i, C_i, \lambda_i \rangle$, where L_i (in bits) denotes the input data size for processing the task, C_i (in CPU cycles/bit) denotes the number of CPU cycles required to process 1-bit of task data, and λ_i (in unit of #task per second) denotes the arrival rate of tasks.

In this article, we use the 3-D Cartesian coordinate system to represent the locations of MUs, UAV, and ECs. The position of the UAV is denoted by $Q^{UAV} = (x^{UAV}, y^{UAV}, H)$, where the height H is assumed to be fixed while the horizontal coordinates x^{UAV} and y^{UAV} affect the channel gain during data communication processes and need to be optimized in our problem. Besides, we assume the positions of MU i and EC j are fixed in our model, which are denoted as $Q_i^{MU} = (x_i^{MU}, y_i^{MU}, 0)$ and $Q_j^{EC} = (x_j^{EC}, y_j^{EC}, 0)$, respectively.

1) *Communication Model*: In the UAV-enabled network, the LoS links are much more dominant than other channel impairments, such as shadowing or small-scale fading due to the high altitude of the UAV. Therefore, the uplink channel

TABLE I
LIST OF NOTATIONS

Symbols	Definitions
Sets and Indices:	
\mathcal{N}	Index set of MUs $i \in \mathcal{N} = \{1, 2, \dots, N\}$
\mathcal{J}	Index set of ECs $j \in \mathcal{J} = \{1, 2, \dots, J\}$
k	Indices of iterations
Decision Variables:	
Q^{UAV}	Position of the UAV
β_{ij}, β_{i0}	Portion of received tasks from MU i to be processed at EC j and the UAV
f_i^{UAV}, f_{ij}^{EC}	Computation resource (in CPU cycles/second) allocated from the UAV to MU i and EC j to MU i
B_i^{UL}	Uplink bandwidth allocated to MU i
Functions:	
h_i^{UL}	Uplink channel gain from MU i to the UAV
h_j^{DL}	Downlink channel gain from the UAV to EC j
d_i^{UL}	Uplink distance from MU i to the UAV
d_j^{DL}	Downlink distance from the UAV to EC j
R_i^{UL}, R_j^{DL}	Achievable uplink/downlink transmission data rate (in bps) from MU i to the UAV and the UAV to EC j
t_i^{G2A}	G2A uplink transmission delay from MU i to the UAV
t_i^{UAV}	Computation delay at the UAV to process the offloaded tasks from MU i
t_{ij}^{A2G}	A2G downlink transmission delay from MU i to EC j via UAV
t_{ij}^{EC}	Computation delay at EC j to process the offloaded tasks from MU i to EC j via UAV
E_i^{CP}	Computation energy consumption of UAV when processing offloaded tasks from MU i
E_{ij}^{TX}, E_i^{RX}	Energy consumption of UAV when transmitting offloaded tasks from MU i to EC j and receiving the offloaded tasks from MU i
E_i^{UAV}	Total energy consumption of UAV when serving the task offloading and computation of MU i
T_i	Total service delay of MU i
Parameters:	
L_i	Input data size for processing tasks (in bits) of MU i
C_i	Number of CPU cycles required to process 1-bit of tasks (in CPU cycles/bit) of MU i
λ_i	Arrival rate of tasks (in number of tasks/second)
H	Height of the UAV
Q_i^{MU}, Q_j^{EC}	Position of MU i and EC j
α_0	Received power at the reference distance of 1 m for a transmission power of 1 W
σ^2	Noise power at the UAV
B^{UL}	Total uplink bandwidth
B^{DL}	Per-device bandwidth pre-assigned to EC j
P_i^{MU}	Transmit power of MU i
P_i^{UAV}, P_j^{UAV}	Transmit and receiving power of the UAV
F_i^{UAV}, F_j^{EC}	Total computation resources at the UAV and EC j
κ	Effective switched capacitance depending on the CPU architecture
ρ	Relative weight of energy and delay

gain from MU i to the UAV can be described by the free-space path loss model

$$h_i^{UL} \triangleq \alpha_0 (d_i^{UL})^{-2} = \frac{\alpha_0}{\|Q_i^{MU} - Q^{UAV}\|^2} \quad (1)$$

where α_0 represents the received power at the reference distance of 1 m for a transmission power of 1 W, d_i^{UL} denotes the uplink distance from MU i to the UAV, and $\|\cdot\|$ denotes the Euclidean norm of a vector. Similarly, the downlink channel

¹Note that we use mobile users and IoT devices interchangeably in this article.

gain from the UAV to EC j can be described as

$$h_j^{\text{DL}} \triangleq \alpha_0 \left(d_j^{\text{DL}} \right)^{-2} = \frac{\alpha_0}{\left\| Q^{\text{UAV}} - Q_j^{\text{EC}} \right\|^2} \quad (2)$$

where d_j^{DL} denotes the downlink distance from the UAV to the EC j .

We assume the FDMA protocol for bandwidth sharing among MUs during the task offloading process. According to Shannon's capacity, the achievable uplink transmission data rate (in bps) from MU i to the UAV can be expressed as

$$R_i^{\text{UL}} = B_i^{\text{UL}} \log_2 \left(1 + \frac{h_i^{\text{UL}} P_i^{\text{MU}}}{\sigma^2} \right) \quad (3)$$

where B_i^{UL} , P_i^{MU} , and σ^2 represent the assigned bandwidth to MU i , transmit power of MU i , and the noise power at the UAV, respectively. For simplicity, we assume the noise power is the same at UAV and ECs [29]. However, it can be easily extended to the case when they are different. Similarly, the downlink transmission data rate (in bps) from the UAV to EC j can be computed as

$$R_j^{\text{DL}} = B_j^{\text{DL}} \log_2 \left(1 + \frac{h_j^{\text{DL}} P_{\text{TX}}^{\text{UAV}}}{\sigma^2} \right) \quad (4)$$

where B_j^{DL} and $P_{\text{TX}}^{\text{UAV}}$ represent the per-device bandwidth² pre-assigned to EC j and transmit power of the UAV, respectively.

2) *Delay Analysis:* In our model, we assume that MUs do not perform local computing due to their limited computation capacities. In contrast, tasks will be first offloaded to the UAV, and then, the UAV will determine the portion of tasks that are processed locally or further offloaded to ECs on the ground. Note that the decision time to split a task is very short compared to the entire communication and computation latency, and therefore, can be neglected. Besides, the output data size of the computation results is often very small compared to the input data size in many computation-intensive applications, such as face recognition and video analysis. Thus, the time needed to send the computation results back to MUs can be ignored as well.

In what follows, we will describe the four key components of the total delay for the offloading process: 1) G2A uplink transmission delay from MUs to the UAV; 2) computation delay at the UAV; 3) A2G downlink transmission delay from the UAV to the ECs; and 4) computation delay at the ECs.

G2A Uplink Transmission Delay From MUs to the UAV: As mentioned before, all the tasks will be offloaded to the UAV first via G2A links without any local computation. Therefore, the G2A transmission delay from MU i to the UAV is computed as the ratio of task input data size and the associated uplink transmission data rate

$$t_i^{\text{G2A}} = \frac{L_i}{R_i^{\text{UL}}}. \quad (5)$$

Computation Delay at the UAV: The UAV will decide the portion of the received tasks that will be processed locally at

the UAV or further offloaded to the ground ECs for processing. Denote $\{\beta_{ij} \in [0, 1], i \in \mathcal{N}, j \in \mathcal{J}\}$ and $\{\beta_{i0} \in [0, 1], i \in \mathcal{N}\}$ as the portion of received tasks from MU i to be processed at EC j and the UAV, respectively. Then, the computation delay at the UAV side to process the offloaded tasks from MU i can be calculated as

$$t_i^{\text{UAV}} = \frac{\beta_{i0} L_i C_i}{f_i^{\text{UAV}}} \quad (6)$$

where f_i^{UAV} (in CPU cycles/s) is the computation resource that the UAV allocates to MU i . Note that when β_{i0} equals 0, it means that no computation will be executed at the UAV side while when β_{i0} equals 1, it indicates that no further offloading will occur from the UAV to ECs.

A2G Downlink Transmission Delay From the UAV to ECs: The UAV may further offload the tasks to more powerful ECs on the ground to reduce the computation latency. Then, the A2G transmission delay from the MU i to EC j via UAV is described as the ratio of offloaded task input data size and the associated downlink transmission data rate

$$t_{ij}^{\text{A2G}} = \frac{\beta_{ij} L_i}{R_j^{\text{DL}}}. \quad (7)$$

Computation Delay at ECs: After receiving the offloaded task data from the UAV, ECs can start the computation process. Therefore, the computation delay at the EC side to process the offloaded task from the MU i to EC j via UAV is

$$t_{ij}^{\text{EC}} = \frac{\beta_{ij} L_i C_i}{f_{ij}^{\text{EC}}} \quad (8)$$

where f_{ij}^{EC} (in CPU cycles/s) is the computation resource that EC j allocates to MU i .

3) *UAV Energy Consumption Analysis:* To ensure service availability, it is important to manage the energy consumption of the UAV due to its limited battery size. In this article, we focus on computation and transmission energy consumption of UAV, and ignore the hovering power since it is independent of our decisions.

Computation Energy Consumption: Similar to [40], we model the power consumption of the CPU in UAV as $\kappa (f_i^{\text{UAV}})^3$, where κ denotes the effective switched capacitance depending on the CPU architecture. It follows that the corresponding energy consumption of UAV when processing tasks offloaded from MU i is given by the product of the power level and computation time

$$E_i^{\text{CP}} = \kappa \left(f_i^{\text{UAV}} \right)^3 t_i^{\text{UAV}} = \kappa \beta_{i0} L_i C_i \left(f_i^{\text{UAV}} \right)^2. \quad (9)$$

Transmission Energy Consumption: The transmission energy consumption of the UAV when receiving the task input data via the G2A uplink transmission channels from MU i is given by

$$E_i^{\text{RX}} = P_{\text{RX}}^{\text{UAV}} t_i^{\text{G2A}} = \frac{L_i P_{\text{RX}}^{\text{UAV}}}{R_i^{\text{UL}}} \quad (10)$$

where $P_{\text{RX}}^{\text{UAV}}$ is the receiving power of UAV. Besides, the transmission energy consumption of the UAV when offloading the

²We assume that each MU is assigned a certain bandwidth beforehand when they communicate with ECs via the UAV.

task input data of MU i via the A2G downlink transmission channels to EC j is given by

$$E_{ij}^{\text{TX}} = P_{\text{TX}}^{\text{UAV}} t_{ij}^{\text{A2G}} = \frac{\beta_{ij} L_i P_{\text{TX}}^{\text{UAV}}}{R_j^{\text{DL}}}. \quad (11)$$

Therefore, the total energy consumption of the UAV when serving the task offloading and computation of MU i is given by

$$E_i^{\text{UAV}} = \lambda_i \left(E_i^{\text{CP}} + E_i^{\text{RX}} + \sum_{j \in \mathcal{J}} E_{ij}^{\text{TX}} \right). \quad (12)$$

B. Problem Formulation

In this article, we are interested in minimizing the total energy consumption of the UAV when serving the computation and communication needs of the MUs and the total service delay of all MUs. To define the service delay of each MU, we make the following assumptions: 1) the UAV cannot partition a task until receiving its entire input data to ensure the accuracy of task splitting; 2) the UAV and ECs cannot start the processing of tasks until the end of the transmission between MUs and the UAV or the UAV and ECs to ensure the reliability of the computation results; and 3) the computation at the UAV can proceed simultaneously with the transmission of the tasks to each EC since the communication and computation modules are often separated at the UAV. Based on the above assumptions, the service delay of MU i can be represented as

$$T_i = t_i^{\text{G2A}} + \max_{j \in \mathcal{J}} \{t_i^{\text{UAV}}, t_{ij}^{\text{A2G}} + t_{ij}^{\text{EC}}\}. \quad (13)$$

Our problem becomes jointly optimizing the UAV position Q^{UAV} , G2A uplink communication resource allocation B_i^{UL} , task partition variables β_{i0} and β_{ij} , and computation resource allocation of the UAV f_i^{UAV} and ECs f_{ij}^{EC} with the goal of minimizing the weighted sum of total energy consumption of UAV and total service delay of all MUs. It can be formulated as the following optimization problem:

$$\min_{\substack{Q^{\text{UAV}}, B_i^{\text{UL}}, \beta_{i0}, \\ \beta_{ij}, f_i^{\text{UAV}}, f_{ij}^{\text{EC}}}} \sum_{i \in \mathcal{N}} E_i^{\text{UAV}} + \rho \sum_{i \in \mathcal{N}} T_i \quad (14a)$$

$$\text{s.t.} \quad \sum_{i \in \mathcal{N}} B_i^{\text{UL}} \leq B^{\text{UL}} \quad (14b)$$

$$\beta_{i0} + \sum_{j \in \mathcal{J}} \beta_{ij} = 1 \quad \forall i \quad (14c)$$

$$\sum_{i \in \mathcal{N}} f_i^{\text{UAV}} \leq F^{\text{UAV}} \quad (14d)$$

$$\sum_{i \in \mathcal{N}} f_{ij}^{\text{EC}} \leq F_j^{\text{EC}} \quad \forall j \quad (14e)$$

$$0 \leq \beta_{ij} \leq 1 \quad \forall i, j \quad (14f)$$

$$0 \leq \beta_{i0} \leq 1 \quad \forall i \quad (14g)$$

$$B_i^{\text{UL}}, f_i^{\text{UAV}} \geq 0 \quad \forall i \quad (14h)$$

$$f_{ij}^{\text{EC}} \geq 0 \quad \forall i, j \quad (14i)$$

where $\rho > 0$ is a parameter defining the relative weight of energy and delay, (14b), (14d), (14e), (14h), and (14i) ensure

that the allocated resources for uplink bandwidth, UAV and EC CPU frequencies are nonnegative and no more than their limits while (14c), (14f), and (14g) constrain that the offloading tasks of MUs are completely processed by UAV and ECs, and the values of partition variables are between 0 and 1.

IV. SOLUTION METHODOLOGY

Problem (14) is hard to solve due to the nonconvexity of the objective function and constraints. In what follows, we will first linearize the maximum term in (13) by leveraging auxiliary variables and reformulate the original optimization problem into a tractable one. Then, we develop an SCA-based algorithm to transform the nonconvex objective function and constraints into suitable convex approximants to iteratively solve the resulting optimization problem.

A. Problem Reformulation

We first define an auxiliary variable for each MU i as $z_i \triangleq \max_{j \in \mathcal{J}} \{t_i^{\text{UAV}}, t_{ij}^{\text{A2G}} + t_{ij}^{\text{EC}}\}$. Then, we linearize the service delay term in (14a) using z_i and reformulate the original optimization problem into the following:

$$\min_{\substack{z_i, Q^{\text{UAV}}, B_i^{\text{UL}}, \\ \beta_{i0}, \beta_{ij}, f_i^{\text{UAV}}, f_{ij}^{\text{EC}}}} \sum_{i \in \mathcal{N}} E_i^{\text{UAV}} + \rho \sum_{i \in \mathcal{N}} (t_i^{\text{G2A}} + z_i) \quad (15a)$$

$$\text{s.t.} \quad z_i \geq t_i^{\text{UAV}} \quad \forall i \quad (15b)$$

$$z_i \geq t_{ij}^{\text{A2G}} + t_{ij}^{\text{EC}} \quad \forall i, j \quad (15c)$$

$$(14b)-(14i). \quad (15d)$$

However, the reformulated optimization problem is still difficult to solve due to the nonconvex objective function (15a) and nonconvex constraints (15b) and (15c). Note that both the uplink and downlink transmission data rate functions (3) and (4) are nonconvex with respect to the UAV position Q^{UAV} .

B. Successive Convex Approximation

In this section, we will show how to build the convex approximation for the nonconvex objective function and nonconvex constraints in the reformulated problem (15) while preserving the local first-order behavior of the original nonconvex problem and solve the resulting problem iteratively to obtain suboptimal solutions by means of SCA. Before we develop the SCA-based algorithm, we first present the background of SCA.

1) *Background of SCA:* Consider the following optimization problem:

$$\mathcal{P}: \min_{\mathbf{x}} U(\mathbf{x}) \quad (16a)$$

$$\text{s.t.} \quad g_l(\mathbf{x}) \leq 0 \quad \forall l = 1, \dots, m \quad (16b)$$

$$\mathbf{x} \in \mathcal{K} \quad (16c)$$

where the objective function $U: \mathcal{K} \rightarrow \mathbb{R}$ is smooth (possibly nonconvex) and $g_l: \mathcal{K} \rightarrow \mathbb{R}$ is smooth (possibly nonconvex), for all $l = 1, \dots, m$; the feasible set is denoted as \mathcal{K} . A widely used method for solving this specific problem is SCA (also known as majorization minimization) where at each iteration, a convex approximation of the original problem is solved via

Algorithm 1 SCA Algorithm for Problem \mathcal{P}

Find a feasible solution $\mathbf{x} \in \mathcal{X}$ in \mathcal{P} , choose a step size $\theta \in (0, 1]$ and set $k = 0$.

Repeat

- 1) Compute $\hat{\mathbf{x}}(\mathbf{x}^k)$, the solution of $\mathcal{P}_{\mathbf{x}^k}$;
- 2) Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \theta(\hat{\mathbf{x}}(\mathbf{x}^k) - \mathbf{x}^k)$;
- 3) Set $k \leftarrow k + 1$.

Until some convergence criterion is met.

replacing the nonconvex objective function and constraints by suitable convex approximants. The convex approximation of the original problem can be stated as follows: given $\mathbf{x}^k \in \mathcal{X}$

$$\mathcal{P}_{\mathbf{x}^k}: \min_{\mathbf{x}} \quad \tilde{U}(\mathbf{x}; \mathbf{x}^k) \quad (17a)$$

$$\text{s.t.} \quad \tilde{g}_l(\mathbf{x}; \mathbf{x}^k) \leq 0 \quad \forall l = 1, \dots, m \quad (17b)$$

$$\mathbf{x} \in \mathcal{K} \quad (17c)$$

where $\tilde{U}(\mathbf{x}; \mathbf{x}^k)$ and $\tilde{g}_l(\mathbf{x}; \mathbf{x}^k)$ represent the approximants of $U(\mathbf{x})$ and $g_l(\mathbf{x})$ at current iterate \mathbf{x}^k , respectively; the feasible set is denoted as $\mathcal{X}(\mathbf{x}^k)$. More specifically, we consider the SCA method presented in Algorithm 1. It is assumed that at each iteration, some original functions $U(\mathbf{x})$ and $g_l(\mathbf{x})$ are approximated by their upper bounds where the same first-order behavior is preserved [41].

2) *SCA-Based Algorithm*: Scutari *et al.* [38] proposed a framework that unifies several existing SCA-based algorithms to solve the problem \mathcal{P} in a parallel and distributed fashion. It also offers much flexibility in the choice of the convex approximation functions, and the objective function U need not be an upper bound of itself at any feasible point. Multiple examples are summarized to find the candidate approximants $\tilde{g}_l(\mathbf{x})$ and $\tilde{U}(\mathbf{x})$ while necessary assumptions are satisfied to develop the SCA-based algorithm. We first present the assumptions and examples that we will utilize to approximate the nonconvex terms in our problem as follows.

Assumption 1: The key assumptions on the choice of the approximated function $\tilde{g}_l: \mathcal{K} \times \mathcal{X} \rightarrow \mathbb{R}$ are given as follows.

- A1) $\tilde{g}_l(\bullet; \mathbf{y})$ is convex on \mathcal{K} for all $\mathbf{y} \in \mathcal{X}$.
- A2) *Upper Bound*: $g_l(\mathbf{x}) \leq \tilde{g}_l(\mathbf{x}; \mathbf{y}) \quad \forall \mathbf{x} \in \mathcal{K}, \mathbf{y} \in \mathcal{X}$.
- A3) *Function Value Consistency*: $\tilde{g}_l(\mathbf{y}; \mathbf{y}) = g_l(\mathbf{y})$, for all $\mathbf{y} \in \mathcal{X}$.
- A4) $\tilde{g}_l(\bullet; \bullet)$ is continuous on $\mathcal{K} \times \mathcal{X}$.
- A5) $\nabla_{\mathbf{x}} \tilde{g}_l(\bullet; \bullet)$ is continuous on $\mathcal{K} \times \mathcal{X}$.
- A6) *Gradient Consistency*: $\nabla_{\mathbf{x}} \tilde{g}_l(\mathbf{y}; \mathbf{y}) = \nabla_{\mathbf{x}} g_l(\mathbf{y})$, for all $\mathbf{y} \in \mathcal{X}$, where $\nabla_{\mathbf{x}} \tilde{g}_l(\mathbf{y}; \mathbf{y})$ denotes the partial gradient of the function \tilde{g}_l with respect to the argument \mathbf{x} evaluated at $(\mathbf{y}; \mathbf{y})$.

Assumption 2: The key assumptions on the choice of the approximated function $\tilde{U}: \mathcal{K} \times \mathcal{X} \rightarrow \mathbb{R}$ are given as follows.

- B1) $\tilde{U}(\bullet; \mathbf{y})$ is uniformly strongly convex on \mathcal{K} with constant $\mu > 0$, i.e., for all $\mathbf{x}, \mathbf{z} \in \mathcal{K}$ and $\mathbf{y} \in \mathcal{X}$:

$$(\mathbf{x} - \mathbf{z})^\top (\nabla_{\mathbf{x}} \tilde{U}(\mathbf{x}; \mathbf{y}) - \nabla_{\mathbf{x}} \tilde{U}(\mathbf{z}; \mathbf{y})) \geq \mu \|\mathbf{x} - \mathbf{z}\|^2.$$

- B2) *Gradient Consistency*: $\nabla_{\mathbf{x}} \tilde{U}(\mathbf{y}; \mathbf{y}) = \nabla_{\mathbf{x}} U(\mathbf{y})$, for all $\mathbf{y} \in \mathcal{X}$.

- B3) $\nabla_{\mathbf{x}} \tilde{U}(\bullet; \bullet)$ is continuous on $\mathcal{K} \times \mathcal{X}$, where $\nabla_{\mathbf{x}} \tilde{U}(\mathbf{u}; \mathbf{v})$ denotes the partial gradient of the function \tilde{U} with

respect to the argument \mathbf{x} evaluated at $(\mathbf{u}; \mathbf{v})$. Note that A1) and B1) make the problem $\mathcal{P}_{\mathbf{x}^k}$ strongly convex while A2) and A3) guarantee the iterate feasibility that $\mathbf{x}^k \in \mathcal{X}(\mathbf{x}^k) \subseteq \mathcal{X}$.

Example 1 (Approximation of $g_l(\mathbf{x})$ [38, Example 3]): Suppose that g_l has a difference of convex (DC) structure, i.e., $g_l(\mathbf{x}) = g_l^+(\mathbf{x}) - g_l^-(\mathbf{x})$ with both g_l^+ and g_l^- being convex and continuously differentiable. By linearizing the concave part g_l^- , we obtain the convex upper approximation of g_l as follows: for all $\mathbf{x} \in \mathcal{K}$ and $\mathbf{y} \in \mathcal{X}$,

$$\tilde{g}_l(\mathbf{x}; \mathbf{y}) \triangleq g_l^+(\mathbf{x}) - g_l^-(\mathbf{y}) - \nabla_{\mathbf{x}} g_l^-(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) \geq g_l(\mathbf{x}). \quad (18)$$

Example 2 (Approximation of $g_l(\mathbf{x})$ [38, Example 4]): Suppose that $g_l(\mathbf{x})$ has a product of functions (PF) structure, i.e., $g_l(\mathbf{x}) = f_1(\mathbf{x})f_2(\mathbf{x})$ with both f_1 and f_2 being convex and nonnegative. Observe that $g_l(\mathbf{x})$ can be rewritten as a function with the DC structure

$$g_l(\mathbf{x}) = \frac{1}{2}(f_1(\mathbf{x}) + f_2(\mathbf{x}))^2 - \frac{1}{2}(f_1^2(\mathbf{x}) + f_2^2(\mathbf{x})). \quad (19)$$

Then, the convex upper approximation of g_l can be obtained by linearizing the concave part in (19): for any $\mathbf{y} \in \mathcal{X}$

$$\begin{aligned} \tilde{g}_l(\mathbf{x}; \mathbf{y}) &\triangleq \frac{1}{2}(f_1(\mathbf{x}) + f_2(\mathbf{x}))^2 - \frac{1}{2}(f_1^2(\mathbf{y}) + f_2^2(\mathbf{y})) \\ &\quad - f_1(\mathbf{y})f_1'(\mathbf{y})(\mathbf{x} - \mathbf{y}) - f_2(\mathbf{y})f_2'(\mathbf{y})(\mathbf{x} - \mathbf{y}) \geq g_l(\mathbf{x}). \end{aligned} \quad (20)$$

Example 3 (Approximation of $U(\mathbf{x})$ [38, Example 8]): Suppose that $U(\mathbf{x})$ has a PF structure, i.e., $U(\mathbf{x}) = h_1(\mathbf{x})h_2(\mathbf{x})$ with both h_1 and h_2 being convex and nonnegative. For any $\mathbf{y} \in \mathcal{X}$, a convex approximation of $U(\mathbf{x})$ is given by

$$\begin{aligned} \tilde{U}(\mathbf{x}; \mathbf{y}) &= h_1(\mathbf{x})h_2(\mathbf{y}) + h_1(\mathbf{y})h_2(\mathbf{x}) \\ &\quad + \frac{\tau}{2}(\mathbf{x} - \mathbf{y})^\top \mathbf{H}(\mathbf{y})(\mathbf{x} - \mathbf{y}) \end{aligned} \quad (21)$$

where $\tau > 0$ is a positive constant, and $\mathbf{H}(\mathbf{y})$ is a uniformly positive-definite matrix.

Then, we transform the nonconvex constraints and nonconvex objective function in the reformulated problem (15) into suitable approximants by following the above examples. For constraint (15b), we observe that the nonconvex term t_i^{UAV} can be written as the product of convex and nonnegative functions³

$$t_i^{\text{UAV}} = L_i C_i g_l(\beta_{i0}, f_i^{\text{UAV}}) = L_i C_i f_1(\beta_{i0}) f_2(f_i^{\text{UAV}}) \quad (22)$$

where $f_1(\beta_{i0}) = \beta_{i0}$ and $f_2(f_i^{\text{UAV}}) = 1/f_i^{\text{UAV}}$. Then, given a feasible solution $\beta_{i0}(k)$ and $f_i^{\text{UAV}}(k)$ for the k th iteration of the SCA-based algorithm, we derive a convex upper approximation of t_i^{UAV} by using Example 2 as

$$\begin{aligned} t_i^{\text{UAV}} &\leq \tilde{t}_i^{\text{UAV}}(\beta_{i0}, f_i^{\text{UAV}}; \beta_{i0}(k), f_i^{\text{UAV}}(k)) \\ &\triangleq L_i C_i \left[\frac{1}{2} \left(\left(\beta_{i0} + \frac{1}{f_i^{\text{UAV}}} \right)^2 - (\beta_{i0}(k))^2 - \left(\frac{1}{f_i^{\text{UAV}}(k)} \right)^2 \right) \right] \end{aligned}$$

³Without loss of generality, we factorize the constants (L_i , C_i , etc.) out of the term since they will not affect the convexity.

$$- (\beta_{i0}(k)(\beta_{i0} - \beta_{i0}(k))) + \left(\frac{1}{f_i^{\text{UAV}}(k)} \right)^3 \left(\frac{1}{f_i^{\text{UAV}}} - \frac{1}{f_i^{\text{UAV}}(k)} \right) \Bigg]. \quad (23)$$

For constraint (15c), t_{ij}^{A2G} can be written as the product of L_i , β_{ij} , and $1/R_j^{\text{DL}}$. However, $1/R_j^{\text{DL}}$ is a nonconvex function with respect to the UAV location Q^{UAV} , and therefore, Example 2 cannot be directly applied to derive a convex upper approximation. To tackle the nonconvexity, we replace it by nonnegative auxiliary variables $\{\phi_j\}_{j \in \mathcal{J}}$. Then, the nonconvex term t_{ij}^{A2G} can be written as the product of convex and nonnegative functions

$$t_{ij}^{\text{A2G}} = L_i g_l(\beta_{ij}, \phi_j) = L_i f_1(\beta_{ij}) f_2(\phi_j) \quad (24)$$

where $f_1(\beta_{ij}) = \beta_{ij}$ and $f_2(\phi_j) = 1/\phi_j$ in (24). Similarly, the nonconvex term t_{ij}^{EC} in (15c) can be written as the product of convex and nonnegative functions

$$t_{ij}^{\text{EC}} = L_i C_i g_l(\beta_{ij}, f_{ij}^{\text{EC}}) = L_i C_i f_1(\beta_{ij}) f_2(f_{ij}^{\text{EC}}) \quad (25)$$

where $f_1(\beta_{ij}) = \beta_{ij}$ and $f_2(f_{ij}^{\text{EC}}) = 1/f_{ij}^{\text{EC}}$ in (25). Then, given a feasible solution $\beta_{ij}(k)$, $\phi_j(k)$, and $f_{ij}^{\text{EC}}(k)$ for the k th iteration of the SCA-based algorithm, we derive convex upper approximation of t_{ij}^{A2G} and t_{ij}^{EC} by using Example 2 as

$$\begin{aligned} t_{ij}^{\text{A2G}} &\leq \tilde{t}_{ij}^{\text{A2G}}(\beta_{ij}, \phi_j; \beta_{ij}(k), \phi_j(k)) \\ &\triangleq L_i \left[\frac{1}{2} \left(\left(\beta_{ij} + \frac{1}{\phi_j} \right)^2 - (\beta_{ij}(k))^2 - \left(\frac{1}{\phi_j(k)} \right)^2 \right) \right. \\ &\quad \left. - (\beta_{ij}(k)(\beta_{ij} - \beta_{ij}(k))) + \left(\frac{1}{\phi_j(k)} \right)^3 \left(\frac{1}{\phi_j} - \frac{1}{\phi_j(k)} \right) \right] \end{aligned} \quad (26)$$

and

$$\begin{aligned} t_{ij}^{\text{EC}} &\leq \tilde{t}_{ij}^{\text{EC}}(\beta_{ij}, f_{ij}^{\text{EC}}; \beta_{ij}(k), f_{ij}^{\text{EC}}(k)) \\ &\triangleq L_i C_i \left[\frac{1}{2} \left(\left(\beta_{ij} + \frac{1}{f_{ij}^{\text{EC}}} \right)^2 - (\beta_{ij}(k))^2 - \left(\frac{1}{f_{ij}^{\text{EC}}(k)} \right)^2 \right) \right. \\ &\quad \left. - (\beta_{ij}(k)(\beta_{ij} - \beta_{ij}(k))) \right. \\ &\quad \left. + \left(\frac{1}{f_{ij}^{\text{EC}}(k)} \right)^3 \left(\frac{1}{f_{ij}^{\text{EC}}} - \frac{1}{f_{ij}^{\text{EC}}(k)} \right) \right]. \end{aligned} \quad (27)$$

By defining $\bar{R}_i^{\text{UL}} \triangleq \log_2(1 + [(h_i^{\text{UL}} P_i^{\text{MU}})/(\sigma^2)])$, we replace $1/\bar{R}_i^{\text{UL}}$ in t_i^{G2A} by nonnegative auxiliary variables $\{\gamma_i\}_{i \in \mathcal{N}}$ since it is a nonconvex function with respect to the UAV location Q^{UAV} . Then, the nonconvex terms in objective function (15a) can be written as the product of convex and nonnegative functions

$$E_i^{\text{CP}} = \kappa L_i C_i h_1(\beta_{i0}) h_2(f_i^{\text{UAV}}) \quad (28)$$

$$E_{ij}^{\text{TX}} = L_i P_{\text{TX}}^{\text{UAV}} h_1(\beta_{ij}) h_3(\phi_j) \quad (29)$$

$$t_i^{\text{G2A}} = L_i h_3(B_i^{\text{UL}}) h_3(\gamma_i) \quad (30)$$

$$E_i^{\text{RX}} = P_{\text{RX}}^{\text{UAV}} t_i^{\text{G2A}} = P_{\text{RX}}^{\text{UAV}} L_i h_3(B_i^{\text{UL}}) h_3(\gamma_i) \quad (31)$$

where $h_1(\beta_{i0}) = \beta_{i0}$ and $h_2(f_i^{\text{UAV}}) = (f_i^{\text{UAV}})^2$ in (28), and $h_1(\beta_{ij}) = \beta_{ij}$ and $h_3(\phi_j) = 1/\phi_j$ in (29), while $h_3(B_i^{\text{UL}}) = 1/B_i^{\text{UL}}$ and $h_3(\gamma_i) = 1/\gamma_i$ in (30). Then, given a feasible solution $\beta_{i0}(k)$, $\beta_{ij}(k)$, $\phi_j(k)$, $\gamma_i(k)$, $B_i^{\text{UL}}(k)$, and $f_i^{\text{UAV}}(k)$ for the k th iteration of SCA-based algorithm, we derive convex approximation of E_i^{CP} , E_{ij}^{TX} , t_i^{G2A} , and E_i^{RX} by using Example 3 as

$$\begin{aligned} \tilde{E}_i^{\text{CP}}(\beta_{i0}, f_i^{\text{UAV}}; \beta_{i0}(k), f_i^{\text{UAV}}(k)) \\ \triangleq \kappa L_i C_i \left(\beta_{i0} (f_i^{\text{UAV}}(k))^2 + \beta_{i0}(k) (f_i^{\text{UAV}})^2 \right) \\ + \frac{\tau_{\beta_{i0}}}{2} (\beta_{i0} - \beta_{i0}(k))^2 + \frac{\tau_{f_i^{\text{UAV}}}}{2} (f_i^{\text{UAV}} - f_i^{\text{UAV}}(k))^2 \end{aligned} \quad (32)$$

$$\begin{aligned} \tilde{E}_{ij}^{\text{TX}}(\beta_{ij}, \phi_j; \beta_{ij}(k), \phi_j(k)) &\triangleq L_i P_{\text{TX}}^{\text{UAV}} \left(\frac{\beta_{ij}}{\phi_j(k)} + \frac{\beta_{ij}(k)}{\phi_j} \right) \\ &+ \frac{\tau_{\beta_{ij}}}{2} (\beta_{ij} - \beta_{ij}(k))^2 + \frac{\tau_{\phi_j}}{2} (\phi_j - \phi_j(k))^2 \end{aligned} \quad (33)$$

$$\begin{aligned} \tilde{t}_i^{\text{G2A}}(B_i^{\text{UL}}, \gamma_i; B_i^{\text{UL}}(k), \gamma_i(k)) &\triangleq L_i \left(\frac{1}{B_i^{\text{UL}} \gamma_i(k)} + \frac{1}{B_i^{\text{UL}}(k) \gamma_i} \right) \\ &+ \frac{\tau_{B_i^{\text{UL}}}}{2} (B_i^{\text{UL}} - B_i^{\text{UL}}(k))^2 + \frac{\tau_{\gamma_i}}{2} (\gamma_i - \gamma_i(k))^2 \end{aligned} \quad (34)$$

and

$$\begin{aligned} \tilde{E}_i^{\text{RX}}(B_i^{\text{UL}}, \gamma_i; B_i^{\text{UL}}(k), \gamma_i(k)) \\ \triangleq P_{\text{RX}}^{\text{UAV}} \tilde{t}_i^{\text{G2A}}(B_i^{\text{UL}}, \gamma_i; B_i^{\text{UL}}(k), \gamma_i(k)) \end{aligned} \quad (35)$$

where $\tau_{\beta_{i0}}, \tau_{\beta_{ij}}, \tau_{\phi_j}, \tau_{\gamma_i}, \tau_{B_i^{\text{UL}}}, \tau_{f_i^{\text{UAV}}} > 0$. Therefore, the convex surrogate objective function of (15a) can be denoted as the nonnegative weighted sum of convex functions

$$\sum_{i \in \mathcal{N}} \lambda_i \left(\tilde{E}_i^{\text{CP}} + \tilde{E}_i^{\text{RX}} + \sum_{j \in \mathcal{J}} \tilde{E}_{ij}^{\text{TX}} \right) + \rho \sum_{i \in \mathcal{N}} (\tilde{t}_i^{\text{G2A}} + z_i) \quad (36)$$

where the convexity is preserved.

Moreover, as we replace the nonconvex data rate functions in both objective function and constraints by the auxiliary variables $\{\phi_j\}_{j \in \mathcal{J}}$ and $\{\gamma_i\}_{i \in \mathcal{N}}$, we obtain equality constraints $\{\phi_j\}_{j \in \mathcal{J}} = 1/R_j^{\text{DL}}$ and $\{\gamma_i\}_{i \in \mathcal{N}} = 1/\bar{R}_i^{\text{UL}}$. To further address the nonconvexity, we first relax them as the following inequalities:

$$0 \leq \phi_j \leq R_j^{\text{DL}} \quad \forall j \quad (37)$$

$$0 \leq \gamma_i \leq \bar{R}_i^{\text{UL}} \quad \forall i \quad (38)$$

where the optimality is preserved since at optimal solutions the auxiliary variables will equate their upper bounds. The key observation is that in (37) and (38), although R_j^{DL} and \bar{R}_i^{UL} are not concave with respect to Q^{UAV} , they are convex functions with respect to $\|Q^{\text{UAV}} - Q_j^{\text{EC}}\|^2$ and $\|Q_i^{\text{MU}} - Q^{\text{UAV}}\|^2$, respectively. Recall that any convex function is globally lower bounded by its first-order Taylor expansion at any point [42]. Therefore, by taking the first-order Taylor expansion of R_j^{DL} and \bar{R}_i^{UL} with respect to $\|Q^{\text{UAV}} - Q_j^{\text{EC}}\|^2$ and $\|Q_i^{\text{MU}} - Q^{\text{UAV}}\|^2$, respectively, we obtain lower bounds of R_j^{DL} and \bar{R}_i^{UL} at local point $Q^{\text{UAV}}(k)$ for the k th iteration of the

SCA-based algorithm as follows:

$$R_j^{\text{DL}} \geq R_{j, \text{LB}}^{\text{DL}}(Q^{\text{UAV}}; Q^{\text{UAV}}(k)) \triangleq R_j^{\text{DL}}(Q^{\text{UAV}}(k)) - \frac{B_j^{\text{DL}} \eta (\|Q^{\text{UAV}} - Q_j^{\text{EC}}\|^2 - \|Q^{\text{UAV}}(k) - Q_j^{\text{EC}}\|^2)}{\ln 2 (\|Q^{\text{UAV}}(k) - Q_j^{\text{EC}}\|^2) (\eta + \|Q^{\text{UAV}}(k) - Q_j^{\text{EC}}\|^2)} \quad \forall j$$

and

$$\bar{R}_i^{\text{UL}} \geq \bar{R}_{i, \text{LB}}^{\text{UL}}(Q^{\text{UAV}}; Q^{\text{UAV}}(k)) \triangleq \bar{R}_i^{\text{UL}}(Q^{\text{UAV}}(k)) - \frac{\varepsilon_i (\|Q_i^{\text{MU}} - Q^{\text{UAV}}\|^2 - \|Q_i^{\text{MU}} - Q^{\text{UAV}}(k)\|^2)}{\ln 2 (\|Q_i^{\text{MU}} - Q^{\text{UAV}}(k)\|^2) (\varepsilon_i + \|Q_i^{\text{MU}} - Q^{\text{UAV}}(k)\|^2)} \quad \forall i$$

(39)

(40)

where $\eta \triangleq \alpha_0 P_{\text{TX}}^{\text{UAV}} / \sigma^2$ and $\varepsilon_i \triangleq \alpha_0 P_i^{\text{MU}} / \sigma^2$. Note that both $R_{j, \text{LB}}^{\text{DL}}$ and $\bar{R}_{i, \text{LB}}^{\text{UL}}$ are concave functions with respect to Q^{UAV} . Then, by replacing R_j^{DL} and \bar{R}_i^{UL} with their lower bounds, we obtain the approximated convex constraints as

$$0 \leq \phi_j \leq R_{j, \text{LB}}^{\text{DL}}(Q^{\text{UAV}}; Q^{\text{UAV}}(k)) \quad \forall j \quad (41)$$

$$0 \leq \gamma_i \leq \bar{R}_{i, \text{LB}}^{\text{UL}}(Q^{\text{UAV}}; Q^{\text{UAV}}(k)) \quad \forall i. \quad (42)$$

Finally, we denote the set of decision variables for our optimization problem as $\psi = (z_i, Q^{\text{UAV}}, B_i^{\text{UL}}, \beta_{i0}, \beta_{ij}, f_i^{\text{UAV}}, f_{ij}^{\text{EC}}, \phi_j, \gamma_i)$. The convex approximation of the reformulated problem (15) with a feasible solution $\psi(k)$ for the k th iteration of the SCA-based algorithm is given by

$$\min_{\psi} \sum_{i \in \mathcal{N}} \lambda_i \left(\tilde{E}_i^{\text{CP}}(\psi; \psi(k)) + \tilde{E}_i^{\text{RX}}(\psi; \psi(k)) + \sum_{j \in \mathcal{J}} \tilde{E}_{ij}^{\text{TX}}(\psi; \psi(k)) \right) + \rho \sum_{i \in \mathcal{N}} (\tilde{t}_i^{\text{G2A}}(\psi; \psi(k)) + z_i) \quad (43a)$$

$$\text{s.t. } z_i \geq \tilde{t}_i^{\text{UAV}}(\psi; \psi(k)) \quad \forall i \quad (43b)$$

$$z_i \geq \tilde{t}_{ij}^{\text{A2G}}(\psi; \psi(k)) + \tilde{t}_{ij}^{\text{EC}}(\psi; \psi(k)) \quad \forall i, j \quad (43c)$$

$$0 \leq \phi_j \leq R_{j, \text{LB}}^{\text{DL}}(\psi; \psi(k)) \quad \forall j \quad (43d)$$

$$0 \leq \gamma_i \leq \bar{R}_{i, \text{LB}}^{\text{UL}}(\psi; \psi(k)) \quad \forall i \quad (43e)$$

$$(14b)-(14i) \quad (43f)$$

which has a unique solution denoted by $\hat{\psi}(\psi(k))$. The above optimization problem (43) is convex, and the SCA-based algorithm is summarized in Algorithm 2.

Note that a diminishing step-size rule is applied in step 2), which is numerically more efficient than a constant one. The convergence of Algorithm 1 is guaranteed if the step size $\theta(k)$ is chosen so that $\theta(k) \in (0, 1]$, $\theta(k) \rightarrow 0$, and $\sum_v \theta(k) = \infty$, then $\psi(k)$ is bounded and at least one of its limit points is stationary [38]. For the termination criterion, it is very convenient to use $\|\hat{\psi}(\psi(k)) - \psi(k)\|$, which is a measure of stationarity. Thus, a reliable termination rule is to check $\|\hat{\psi}(\psi(k)) - \psi(k)\| \leq \zeta$, where ζ is the desired accuracy.

Algorithm 2 SCA-Based Algorithm for Problem (43)

Input: $\psi(0) = (z_i, Q^{\text{UAV}}(0), B_i^{\text{UL}}(0), \beta_{i0}(0), \beta_{ij}(0), f_i^{\text{UAV}}(0), f_{ij}^{\text{EC}}(0), \phi_j(0), \gamma_i(0))$, and $\tau_{\beta_{i0}}, \tau_{\beta_{ij}}, \tau_{\phi_j}, \tau_{\gamma_i}, \tau_{B_i^{\text{UL}}}$, $\tau_{f_i^{\text{UAV}}} > 0$ for $i \in \mathcal{N}$ and $j \in \mathcal{J}$, $\theta(k) \in (0, 1]$. Set $k = 0$, $\alpha = 0.5$.

Repeat

1) Compute $\hat{\psi}(\psi(k))$, the solution of (43);

2) Set $\psi(k+1) = \psi(k) + \theta(k)(\hat{\psi}(\psi(k)) - \psi(k))$, with $\theta(k) = \theta(k-1)(1 - \alpha\theta(k))$;

3) Set $k \leftarrow k + 1$.

Until $\psi(k)$ is a stationary solution of (14).

Output: $Q^{\text{UAV}}, B_i^{\text{UL}}, \beta_{i0}, \beta_{ij}, f_i^{\text{UAV}}$ and f_{ij}^{EC} .

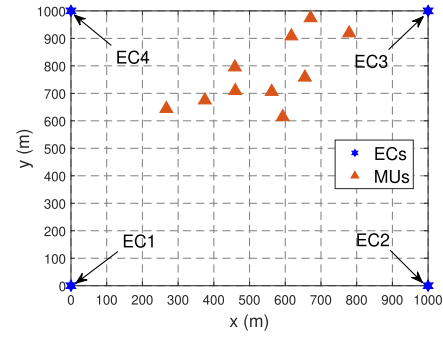


Fig. 2. Locations of 10 MUs and 4 ECs in the MEC system.

TABLE II
SIMULATION PARAMETERS

Parameters	Values	Parameters	Values
H	100 m	$P_{\text{TX}}^{\text{UAV}}$	1 W
α_0	-50 dB	L_i	[1, 5] Mbits
σ^2	-100 dBm	C_i	[100, 200] CPU cycles/bit
κ	10^{-28} [43], [44]	B_i^{UL}	10 MHz
λ_i	30 tasks/min	B_j^{DL}	0.5 MHz
P_i^{MU}	0.1 W	F_i^{UAV}	3 GHz
$P_{\text{RX}}^{\text{UAV}}$	0.1 W	F_j^{EC}	[6, 9] GHz
ρ	5		

V. NUMERICAL EXPERIMENTS

In this section, we validate the effectiveness of our proposed SCA-based algorithm via extensive numerical experiments. All the experiments are implemented in MATLAB R2018a using CVX on a desktop computer with an Intel Core i7-4790 3.60-GHz CPU and 16-GB RAM. The convergence tolerance threshold ζ for the proposed algorithm is set to be 10^{-2} .

A. Simulation Setup

We consider a UAV-enabled MEC system with 4 ground ECs placed at each vertex and 10 ground MUs that are randomly distributed within a 2-D area of $1000 \times 1000 \text{ m}^2$, as illustrated in Fig. 2. The UAV is deployed and operated to facilitate the MEC service provisioning, and the optimal 3-D location of UAV can be found using our proposed SCA-based algorithm. The simulation parameter settings are summarized in Table II unless otherwise stated.

As mentioned in Section I, our system settings involving the interactions among IoT devices, UAV, and ECs are different from prior works. The approaches proposed in their studies are not directly applicable to our settings. Therefore, we consider the following intuitive methods as baselines.

- 1) *Random UAV Location Scheme*: The task splitting and resource allocation decisions are optimized while the UAV location is randomly selected without optimization.
- 2) *UAV-Only Scheme*: All tasks are offloaded and processed at the UAV without further offloading to any ECs.
- 3) *EC-Only Scheme*: All tasks are first offloaded to the UAV without any computations and further offloaded to ECs for processing.
- 4) *Fixed UAV-EC Scheme*: Half of the tasks is processed at the UAV while the other half is processed at ECs.

Note that the UAV is deployed at the optimal position for the last three baselines similar to our proposed method. To investigate the importance of UAV location optimization, we name our proposed method as *optimized UAV location scheme* and compare it with the random UAV location scheme. To study the benefits of utilizing computing capacity at both UAV and ECs, we rename our proposed method as *collaborative UAV-EC scheme* and compare it with the UAV-only, EC-only, and fixed UAV-EC schemes.

B. Experimental Results

In this section, we first simulate and analyze how UAV position and per-device bandwidth in the downlink communication will affect the system cost of the studied UAV-enabled MEC system. Then, we compare the performances of our proposed collaborative UAV-EC scheme with UAV-only, EC-only, and fixed UAV-EC schemes to verify the effectiveness of our method in reducing the overall system cost as well as the benefits of UAV-EC collaboration. We set the simulation parameters F_j^{EC} ($j = 1, 2, 3, 4$) and L_i ($i = 1, 2, \dots, 10$) to be [8, 9, 6, 7] GHz and [3, 5, 2, 3, 5, 1, 1, 5, 4, 5] Mbits, respectively.

1) *Importance of Optimizing the UAV Position*: In this part, we compare the performances of our proposed optimized UAV location scheme with the random UAV location scheme where the location of UAV is randomly assigned without optimization in terms of reducing the system cost. The results are summarized in Table III. It is shown that under the optimized UAV location scheme, the optimal 3-D position $(x_*^{\text{UAV}}, y_*^{\text{UAV}}, H)$ found for the UAV is at (558.11, 724.52, 100) m. The system cost of the optimized UAV location scheme is 20.83, which is the best compared with randomly selected UAV locations (at the center or near each EC), and our proposed scheme can achieve high-cost saving as 13.39%. The rationale behind the system cost difference is that for our proposed approach, the UAV location is optimized to obtain better channel condition when providing the offloading opportunities for ground MUs while for the random UAV location scheme, the UAV location is randomly assigned beforehand without optimization. Besides, the optimal task splitting ratios of MUs for UAV and ECs are shown in Fig. 3. We observe that for MUs 2, 5, 8, 9, and 10 with large amount of input data size, 32.12% tasks in

TABLE III
SYSTEM COST COMPARISON FOR OPTIMIZED UAV LOCATION AND RANDOM UAV LOCATION SCHEMES

Optimal UAV location (558.11, 724.52, 100) m	System cost 20.83	
Random UAV location (500, 500, 100) m	System cost 22.19	Cost saving percentage 6.13%
(100, 100, 100) m	24.05	13.39%
(900, 100, 100) m	23.85	12.67%
(900, 900, 100) m	22.44	7.17%
(100, 900, 100) m	23.07	9.71%

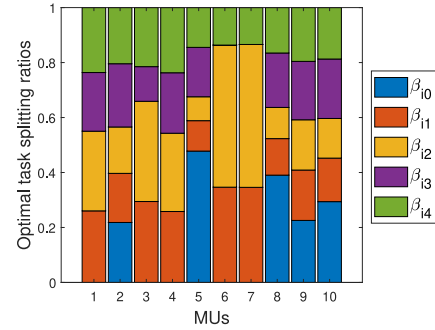


Fig. 3. Optimal task splitting ratios of the UAV β_{i0} ($i = 1, 2, \dots, 10$) and ECs β_{ij} ($j = 1, 2, 3, 4$) for MUs.

average are first processed at the UAV (i.e., β_{i0}) to reduce the data size, and then the remaining tasks are distributed to ECs for further processing.

2) *Impact of the Per-Device Bandwidth*: In this part, we first study how the per-device bandwidth will affect the optimal task splitting ratios at ECs. As mentioned before, per-device bandwidth is assigned to each MU beforehand, and it plays an important role in affecting the optimal task splitting ratios at ECs and system cost. To proceed, we increase the per-device bandwidth B_1^{DL} assigned to EC1 from 0.5 to 5 MHz while the other three B_2^{DL} , B_3^{DL} , and B_4^{DL} remain unchanged. In Fig. 4, we observe that the optimal task splitting ratio β_{11} for MU1 at EC1 is increasing while the other three β_{12} , β_{13} , and β_{14} are decreasing. The reason is that as B_1^{DL} increases, the A2G downlink transmission delay t_{11}^{A2G} from the UAV to the EC1 can be reduced, and then, more tasks will be offloaded to EC1 for further processing and therefore, the corresponding optimal task splitting ratio grows.

Next, we investigate how the per-device bandwidth will affect the system cost. To proceed, we increase the per-device bandwidth B_j^{DL} assigned to the j th EC ($j = 1, 2, 3, 4$) from 0.5 to 5 MHz while the other three remain at 0.5 MHz. In Fig. 5, we observe that the system cost reduces as the per-device bandwidth assigned to the j th EC increases. The reason is that as more bandwidth assigned to each MU when tasks are offloaded from the UAV to the j th EC, higher downlink transmission data rates can be achieved, and thus, the downlink transmission delay and downlink transmission energy consumption of the UAV can be reduced accordingly.

3) *Benefits of UAV-EC Collaboration*: In this part, we compare the performances of our proposed collaborative UAV-EC schemes with UAV-only, EC-only and fixed UAV-EC schemes

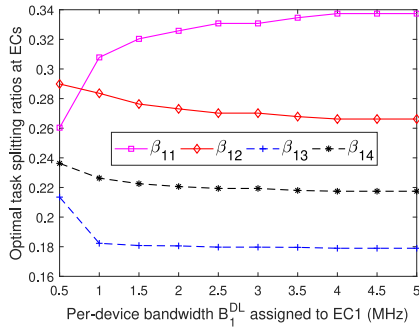


Fig. 4. Optimal task splitting ratios at each EC for MU1 as a function of per-device bandwidth B_1^{DL} assigned to EC1.

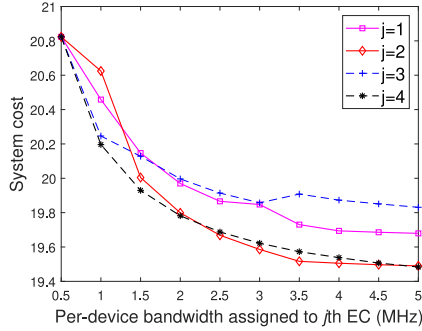


Fig. 5. System cost as a function of per-device bandwidth B_j^{DL} assigned to the j th EC ($j = 1, 2, 3, 4$) while fixing others at 0.5 MHz.

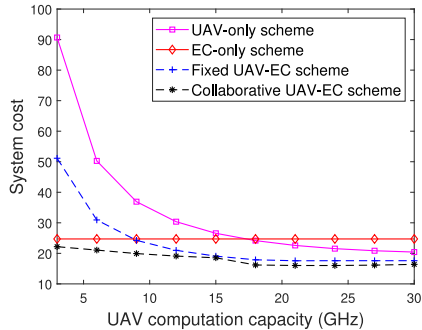


Fig. 6. System cost as a function of the UAV computation capacity F^{UAV} under four different offloading schemes.

in terms of reducing the system cost. Meanwhile, we investigate how system cost behaves as the UAV computation capacity and UAV transmission power change, respectively. First, we study how system cost behaves as the UAV computation capacity F^{UAV} increases from 3 to 30 GHz. As described in Fig. 6, the system cost of EC-only scheme does not change as F^{UAV} varies since this scheme prescribes that all MUs must offload their tasks to ECs without any computations at the UAV side. We further observe that for the other three offloading schemes, all system cost decreases as F^{UAV} increases since more computation resources are available to reduce the task computation delay at the UAV side. Then, we investigate how system cost will be affected as the UAV transmission power $P_{\text{TX}}^{\text{UAV}}$ increases from 1 to 10 W. As illustrated in Fig. 7, the system cost of the UAV-only scheme remains constant as $P_{\text{TX}}^{\text{UAV}}$ varies since this scheme indicates that all MUs must offload

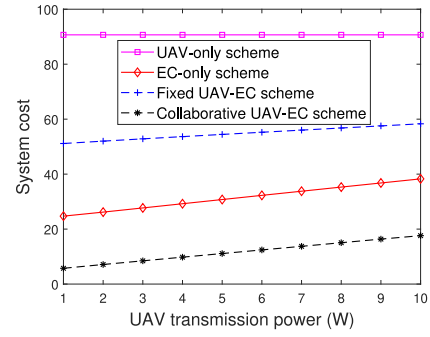


Fig. 7. System cost as a function of the UAV transmission power $P_{\text{TX}}^{\text{UAV}}$ under four different offloading schemes.

their tasks to the UAV for execution without further offloading to any ECs. We further observe that for the other three offloading schemes, system cost increases as $P_{\text{TX}}^{\text{UAV}}$ increases since the downlink transmission energy consumption of the UAV is an increasing function of $P_{\text{TX}}^{\text{UAV}}$. Under the above two scenarios, we observe that our proposed approach largely outperforms baseline schemes, such as UAV-only, EC-only, and fixed UAV-EC offloading schemes in terms of reducing the system cost, which verifies the benefits of UAV-EC collaboration in the task offloading processes.

VI. CONCLUSION

In this article, we have studied an innovative UAV-enabled MEC system involving the interactions among IoT devices, UAV, and ECs. We have proposed to deploy a UAV properly to facilitate the MEC service provisioning to a set of stationary IoT devices in regions where the existing ECs cannot be accessible to IoT devices due to terrestrial signal blockage and shadowing. The UAV and ECs in our system collaboratively provide MEC services to the IoT devices using aerial-to-ground communications. We have formulated a non-convex optimization problem with the goal of minimizing the weighted sum of the service delay of all IoT devices and UAV energy consumption by jointly optimizing UAV position, communication and computing resource allocation, and task splitting decisions. We have developed an SCA-based algorithm to tackle the nonconvexity of the original problem by first transforming the original nonconvex problem into its approximated convex form and then solve it efficiently. We have also conducted numerical experiments to verify that our proposed collaborative UAV-EC offloading scheme largely outperforms baseline schemes that solely rely on UAV or ECs for MEC in IoT. In the future, we will extend our work to the setting of multiple UAVs and investigate the task offloading and UAV swarm placement in the multihop MEC scenario.

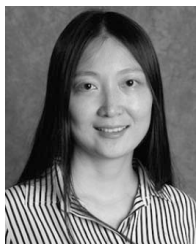
REFERENCES

- [1] *Internet of Things Data to Top 1.6 Zettabytes by 2020*. Accessed: Jan. 6, 2019. [Online]. Available: <https://campustechnology.com/articles/2015/04/15/internet-of-things-data-to-top-1-6-zettabytes-by-2020.aspx>
- [2] *GSMA Intelligence, the Mobile Economy 2018*. Accessed: Jan. 6, 2019. [Online]. Available: <https://www.gsma.com/mobileeconomy/wp-content/uploads/2018/02/The-Mobile-Economy-Global-2018.pdf>
- [3] G. Ananthanarayanan *et al.*, "Real-time video analytics: The killer app for edge computing," *Computer*, vol. 50, no. 10, pp. 58–67, 2017.

- [4] J. Wang, B. Amos, A. Das, P. Pillai, N. Sadeh, and M. Satyanarayanan, "A scalable and privacy-aware IoT service for live video analytics," in *Proc. ACM MMSys*, 2017, pp. 38–49.
- [5] M. Billinghurst *et al.*, "A survey of augmented reality," *Found. Trends[®] Human-Comput. Interact.*, vol. 8, nos. 2–3, pp. 73–272, 2015.
- [6] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
- [7] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.
- [8] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—A key technology towards 5G," vol. 11, ETSI, Sophia Antipolis, France, White Paper, pp. 1–16, 2015.
- [9] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.
- [10] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.
- [11] Y. Xiao and M. Krunz, "QoE and power efficiency tradeoff for fog computing networks with fog node cooperation," in *Proc. IEEE INFOCOM*, 2017, pp. 1–9.
- [12] J. Xu, L. Chen, and P. Zhou, "Joint service caching and task offloading for mobile edge computing in dense networks," in *Proc. IEEE INFOCOM*, 2018, pp. 207–215.
- [13] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.
- [14] J. Ren, G. Yu, Y. He, and G. Y. Li, "Collaborative cloud and edge computing for latency minimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 5031–5044, May 2019.
- [15] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.
- [16] K. Zhang *et al.*, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, 2016.
- [17] C. You, K. Huang, H. Chae, and B.-H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.
- [18] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
- [19] M.-H. Chen, B. Liang, and M. Dong, "Joint offloading and resource allocation for computation and communication in mobile cloud with computing access point," in *Proc. IEEE INFOCOM*, 2017, pp. 1–9.
- [20] M.-H. Chen, M. Dong, and B. Liang, "Resource sharing of a computing access point for multi-user mobile cloud offloading with delay constraints," *IEEE Trans. Mobile Comput.*, vol. 17, no. 12, pp. 2868–2881, Dec. 2018.
- [21] J. Zhang, W. Xia, F. Yan, and L. Shen, "Joint computation offloading and resource allocation optimization in heterogeneous networks with mobile edge computing," *IEEE Access*, vol. 6, pp. 19324–19337, 2018.
- [22] 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.
- [23] M. J. Neely, "Intelligent packet dropping for optimal energy-delay trade-offs in wireless downlinks," *IEEE Trans. Autom. Control*, vol. 54, no. 3, pp. 565–579, Mar. 2009.
- [24] H. Yu and M. J. Neely, "A new backpressure algorithm for joint rate control and routing with vanishing utility optimality gaps and finite queue lengths," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1605–1618, Aug. 2018.
- [25] G. Sharma, R. Mazumdar, and N. B. Shroff, "Delay and capacity trade-offs in mobile ad hoc networks: A global perspective," *IEEE/ACM Trans. Netw.*, vol. 15, no. 5, pp. 981–992, Oct. 2007.
- [26] Z. Mao, C. E. Koksal, and N. B. Shroff, "Near optimal power and rate control of multi-hop sensor networks with energy replenishment: Basic limitations with finite energy and data storage," *IEEE Trans. Autom. Control*, vol. 57, no. 4, pp. 815–829, Apr. 2012.
- [27] X. Cao, P. Yang, M. Alzenad, X. Xi, D. Wu, and H. Yanikomeroglu, "Airborne communication networks: A survey," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1907–1926, Sep. 2018.
- [28] 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.
- [29] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for UAV-enabled mobile relaying systems," *IEEE Trans. Commun.*, vol. 64, no. 12, pp. 4983–4996, Dec. 2016.
- [30] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 2109–2121, Mar. 2018.
- [31] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient Internet of Things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.
- [32] C. Zhan and Y. Zeng, "Completion time minimization for multi-UAV-enabled data collection," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4859–4872, Oct. 2019.
- [33] 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.
- [34] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, Sep. 2018.
- [35] A. Asheralieva and D. Niyato, "Hierarchical game-theoretic and reinforcement learning framework for computational offloading in UAV-enabled mobile edge computing networks with multiple service providers," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8753–8769, Oct. 2019.
- [36] X. Hu, K.-K. Wong, K. Yang, and Z. Zheng, "UAV-assisted relaying and edge computing: Scheduling and trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4738–4752, Mar. 2019.
- [37] G. Scutari, F. Facchinei, L. Lampariello, S. Sardellitti, and P. Song, "Parallel and distributed methods for constrained nonconvex optimization—Part II: Applications in communications and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1945–1960, Apr. 2017.
- [38] G. Scutari, F. Facchinei, and L. Lampariello, "Parallel and distributed methods for constrained nonconvex optimization—Part I: Theory," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1929–1944, Oct. 2017.
- [39] Q. Wu, L. Liu, and R. Zhang, "Fundamental trade-offs in communication and trajectory design for UAV-enabled wireless network," *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 36–44, Feb. 2019.
- [40] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.
- [41] M. Razaviyayn, "Successive convex approximation: Analysis and applications," Ph.D. dissertation, Faculty Grad. School, Univ. Minnesota, Minneapolis, MN, USA, 2014.
- [42] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [43] W. Yuan and K. Nahrstedt, "Energy-efficient soft real-time CPU scheduling for mobile multimedia systems," *ACM SIGOPS Oper. Syst. Rev.*, vol. 37, no. 5, pp. 149–163, 2003.
- [44] W. Yuan and K. Nahrstedt, "Energy-efficient CPU scheduling for multimedia applications," *ACM Trans. Comput. Syst.*, vol. 24, no. 3, pp. 292–331, 2006.



Zhe Yu (Student Member, IEEE) received the B.Eng. degree in communication engineering from the University of Science and Technology Beijing, Beijing, China, in 2014, and the M.S. degree in electrical engineering from Vanderbilt University, Nashville, TN, USA, in 2016. He is currently pursuing the Ph.D. degree in electrical and computer engineering with Oklahoma State University, Stillwater, OK, USA, where his research is on resource management for energy-efficient cloud and edge systems.



Yanmin Gong (Member, IEEE) received the B.Eng. degree in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2009, the M.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2012, and the Ph.D. degree in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 2016.

She is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, TX, USA. She was an Assistant Professor with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, USA, from 2016 to 2018. Her research interests include security and privacy in big data, machine learning, Internet of Things, cyber-physical systems, mobile computing, and wireless networks.

Dr. Gong was a recipient of the Best Paper Award of GLOBECOM 2017. She is serving as an Associate Editor for IEEE WIRELESS COMMUNICATIONS.



Yuanxiong Guo (Senior Member, IEEE) received the B.Eng. degree in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2009, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 2012 and 2014, respectively.

From 2014 to 2019, he was with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, USA. He has been an Assistant Professor with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX, USA, since August 2019. His current research interests include data analytics, security, and privacy with applications to Internet of Things, cyber-physical systems, and cloud/edge systems.

Dr. Guo is a recipient of the Best Paper Award in the IEEE Global Communications Conference 2011. He is currently serving as an Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.



Shimin Gong (Member, IEEE) received the B.E. and M.E. degrees in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2008 and 2012, respectively, and the Ph.D. degree in computer engineering from Nanyang Technological University, Singapore, in 2014.

He was an Associated Researcher with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Beijing, China. He is currently an Associate Professor with the School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou, China. He was a Visiting Scholar with the Chinese University of Hong Kong, Hong Kong, in 2011, and the University of Waterloo, Waterloo, ON, Canada, in 2012. His research interests include wireless powered IoT, deep reinforcement learning, backscatter communications, and networking.

Dr. Gong is a recipient of the Best Paper Award on MAC and Cross-layer Design in IEEE WCNC 2019. He has been the Lead Guest Editor of the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING—Special Issue on Deep Reinforcement Learning on Future Wireless Communication Networks.