

Joint Laser Charging and DBS Placement for Drone-assisted Edge Computing

WeiQi Liu, *Student member, IEEE*, Shuai Zhang, *Student member, IEEE*, and Nirwan Ansari, *Fellow, IEEE*

Abstract—Deploying a drone-mounted base-station (DBS) to assist mobile edge computing can empower 5G and beyond networks with additional flexibility and maneuverability, and laser charging can potentially extend the DBS’s service time. A laser charging DBS framework is proposed in which a DBS is provisioning services for user equipments (UEs) on the ground and harvesting energy transmitted from the laser charging station mounted on the macro base station (MBS). Both MBS and DBS are equipped with servers. UEs can offload their tasks to either the MBS or the DBS. The DBS is to be placed at the optimal location to provide uplink communications and computing services for the ground UEs in each time slot. We thus formulate the joint user Association bandwidth and Computation Resource Assignment Laser charging (ACCRUAL) problem to jointly maximize the DBS service time and minimize the task completion time for all UEs. Since ACCRUAL is a mixed integer nonlinear problem, we decompose it into two sub-problems: the joint UE association computing Resource And bandwidth allocation Problem (REAP) and the DBS placement problem. An iterative algorithm is employed to solve the REAP problem and a placement algorithm based on counting sort is used to tackle the DBS placement problem. The performance of our algorithm is superior to the greedy algorithm and equally shared resource allocation algorithm upon which the total UE task offloading completion time is improved by more than 9% as compared to the greedy algorithm and the total DBS service time is improved by 20% when the laser power is 200 w and 40% of the DBS service time is improved when the laser power is 400 w.

Index Terms—wireless communications, drone mounted base-station (DBS), edge computing, resource allocation, laser charging

I. INTRODUCTION

With the proliferation of mobile devices and the deployment of 5G networks, many advanced mobile applications such as augmented reality, face recognition and mobile online games [1] are emerging. These applications require intensive computation and impose stringent task completion time. However, mobile devices are typically resource-constrained, due to their limited computation resources and battery life. Mobile Cloud Computing (MCC) has been proposed to address the challenges. However, offloading tasks to remote cloud servers may incur a long task completion time. The total number of connected devices in the world in 2020 was over 21.7 billion including 11.7 billion IoT devices and 5.6 billion mobile phones. The total number of connected devices in the world in

2025 is estimated to be over 41.2 billion including 30.9 billion IoT devices and 5.8 billion mobile phones [2]. Traffics and tasks generated by these devices will overwhelm the cloud; as a result, the network bandwidth will be throttled, thus leading to a larger data bottleneck. Mobile Edge Computing (MEC) has been proposed to alleviate this problem because MEC facilitates data processing to be closer to the source and can thus reduce task completion time as well as relieve the burden of the clouds.

Deploying drone-mounted base-stations (DBSs) to provision communications and computation services to user equipments (UEs) for temporary use cases in specific areas such as disaster areas and hot spots [4] [5] is promising. Since a DBS can be configured and deployed easily, it is convenient for service providers to deploy the DBS at a location where it may have a good channel condition, thus reducing the path loss between the DBS and UEs. Also, DBSs can bring the computing capability even closer to the UEs. Thus, the UEs can leverage the computing resources to handle their tasks without sending their tasks to the remote clouds. However, the limited DBS battery life curtails its service capabilities. A general portable base station weights from 1lb to 2.35lb [6]. For heavy lift drones, the flight usually lasts less than 20 minutes with their maximum payload [7]. The scant battery life and heavy payload restricts the DBS service range and time.

Charging the DBS via a wire is a solution to overcome the insufficient battery capacity issue. In 2017, 90 percent of the cellular sites were destroyed by Hurricane Maria [8] in Puerto Rico. To provision emergency communications services for victims, tethered DBSs have been deployed by AT&T [9]. However, the long wire (tether) increases the payload and the cost of deploying the DBS. Also, the wire restricts the service range and reduce the maneuverability of the DBS. Therefore, wireless charging can be an alternative solution, e.g., via radio frequency (RF) and laser. Since RF charging has a large divergence angle, it is only suitable to charge devices in a proximity range. The laser is able to travel over 500m in moderate fog, 1.9km in foggy weather, and 50km on a sunny day [10]. Owing to the laser’s long transmission distance, high energy transmission efficiency, and small divergence angle, laser charging is becoming a feasible solution to charge the DBS. To extend the time for DBS to provide computing services to UEs, we propose a laser charging enabled DBS-assisted MEC network structure.

II. STATE OF THE ART

Multiple types of research related to DBS communications, DBS-assisted edge computing, and wireless charging of drones

This work was supported in part by National Science Foundation under Grant CNS-1814748. (*Corresponding author: WeiQi Liu.*)

WeiQi Liu, Shuai Zhang and Nirwan Ansari are with the Advanced Networking Laboratory, Department of Electrical and Computing Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (email: wl296@njit.edu; sz355@njit.edu; nirwan.ansari@njit.edu).

have been studied. Ansari *et al.* [11] first postulated the concept of enabling DBSs to simultaneously receive data and energy via the laser beams. Al-Hourani *et al.* [12] studied the line-of-sight (LoS) and non-LoS air-to-ground communications channels, and they elucidated a probabilistic LoS path loss model. Yang *et al.* [13] studied the IoT networks and proposed to implement multiple drones to assist the mobile edge computing to balance the load. Zhang *et al.* [14] proposed a $1-\epsilon$ approximation algorithm to minimize the latency of UAV-aided MEC networks. Mostafa *et al.* [15] constructed a wireless energy harvesting system for drones to extend the flight time. Wu *et al.* [16] jointly optimized the DBS trajectory and the user association scheduling to maximize the minimum average data rate among all users. Ouyang *et al.* [17] investigated the communications from the laser-powered DBS to the base station. A wireless charging system in which a DBS is implemented to energize the ground UEs via RF transmitter on the DBS was proposed by Jiang *et al.* [18], and they proposed an algorithm to maximize the uplink throughput from the UEs to the DBS. Sun *et al.* [19] investigated a solar-powered UAV communications network system and designed a 3D aerial trajectory to extend the UAV service time. They also proposed a resources allocation algorithm to maximize the throughput of the system. Moradi *et al.* [30] proposed a SkyCore architecture that pushes the evolved packet core (EPC) functionality to the edge to enhance the DBS-assisted edge computing network. They also deployed the SkyCore on a two-DBS LTE network and demonstrated its performance on control and data management as compared to other EPC in a DBS-assisted edge computing network. Fattore *et al.* [31] proposed to mount a low-cost system-on-chip on the drone and built a DBS to achieve the flying 5G user plane function, which enables the drone to relay the user data to the cellular base station or process the data on the drone. The performance of their drone-extended mobile core architecture was demonstrated through a field test. Zhang *et al.* [32] studied the computation-intensive Internet of Things Devices (IoTDs) network and proposed a DBS-assisted MEC architecture to provide services to IoTDs. They also proposed a $(1 + \epsilon)$ -approximation algorithm to minimize the operational cost of the network.

Different from the works mentioned above. The motivation of this work is to minimize the task completion time of all UEs while maximizing the DBS flight time. We propose a DBS assisted mobile edge computing framework to reduce the task offloading completion time that can greatly support computation and communications for temporary scenarios such as hot spots and disaster areas; a laser power source is implemented to charge the DBS, and the DBS is deployed to provision communications and computing for ground UEs; the users association, bandwidth assignment, DBS placement, the energy consumption of the DBS and the laser charging rate are investigated in this work.

In this article, the placement of the DBS is investigated in a time slot fashion. Here, one time slot is a fixed time duration. Within a time slot, the DBS provisions ground UEs at a fixed location. At the same time, the computing resources, bandwidth assignment and the UE association should

be delicately designed to minimize the completion time of all tasks in the network while extending the service time of the DBS. To achieve the objective, we formulate the joint user Association bandwidth and Computation Resource Assignment Laser charging (ACCRUAL) problem, and jointly minimize the task completion time and maximize the total DBS service time by optimizing the DBS placement, UE association, and limited bandwidth and computing resources assignment.

The main contributions of this article are delineated as follows:

- 1) A laser charged DBS assisted MEC framework is proposed where a laser power source is mounted on the MBS to energize a DBS. Both the DBS and MBS are provisioning uplink communications and computations for the ground UEs.
- 2) A charging model based on battery properties is proposed to estimate the battery usage time of the DBS.
- 3) An iterative algorithm is proposed to solve the joint UE association, bandwidth and computing resource allocation problem (REAP). The DBS placement problem is addressed by a proposed placement algorithm.

The remainder of this article is organized as follows. In Section II, the communications between UE to DBS and UE to MBS are defined. The task computing model and energy harvesting model are presented. In Section III, the problem formulation is elucidated. In Section IV, two heuristic algorithms are proposed. In Section V, the performance of the algorithms is evaluated and demonstrated with extensive simulation results. In Section VI, the conclusions are presented.

III. SYSTEM MODEL

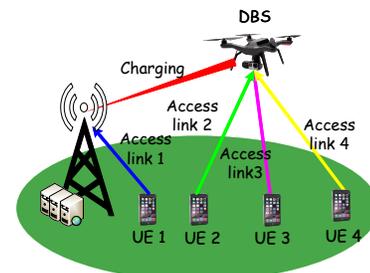


Fig. 1. System model.

A laser charged DBS assisted mobile edge computing network is proposed to provision the ground UEs for temporary events. As shown in Fig. 1, the MEC network system is composed of a DBS, an MBS, a remote laser power source mounted on the MBS and a group of ground UEs. We assume that multiple UEs can be served simultaneously via different frequency bands. Both MBS and DBS are equipped with servers. The servers are assumed to be multi-cores and able to handle multiple tasks simultaneously. The server on the MBS is much more powerful than that on the DBS. The UEs can offload their tasks to either MBS or DBS. The resources for communications and computations are limited. The DBS hovers at a fixed height and works as a base station to provide

uplink and computing services for the ground UEs. A laser power source is mounted on the MBS to energize the DBS to prolong its service time. Since the laser energy can only extend the DBS service time instead of exclusively powering the DBS, the DBS must retain adequate energy to fly back to the MBS. The positions of the UEs are assumed known by the DBS and the MBS. We further assume the UEs are with low mobility. The DBS may adjust its location to minimizing the task completion time of all UEs while harvesting enough energy to prolong its service time in every time slot.

A. Communications Model

We assume the communications channel between the i -th user equipment UE_i and the DBS is probabilistic LoS [12] expressed as:

$$p^{los} = [1 + be^{-a(\theta_i-b)}]^{-1}, \quad (1)$$

where the environmental parameters a and b are corresponding to different terrains e.g., urban or rural [20]. The elevation angle between UE_i and the DBS is denoted as θ_i , which can be calculated by $\theta_i = \arctan(h/i)$. Here, i and h are the horizontal and vertical distance between UE_i and the DBS, respectively.

The non-LoS and LoS path loss between UE_i and the DBS are expressed as:

$$\varphi_i^{nlos} = \zeta^{nlos} + \pi^{nlos} \log_{10}(\sqrt{l_i^2 + h^2}), \quad (2)$$

$$\varphi_i^{los} = \zeta^{los} + \pi^{los} \log_{10}(\sqrt{l_i^2 + h^2}), \quad (3)$$

where ζ^{los} and ζ^{nlos} are the LoS and non-LoS path loss at a reference distance. π^{los} and π^{nlos} are the LoS and non-LoS path exponent, respectively [21], [22]. Since we know the probability density functions of the non-LoS and LoS path loss, we can derive the average data rate between UE_i and the DBS as:

$$R_{id} = b_i \log_2 \left(1 + \frac{p_i \varphi_i^{los}}{\sigma^2} \right) p^{los} + b_i \log_2 \left(1 + \frac{p_i \varphi_i^{nlos}}{\sigma^2} \right) p^{nlos}. \quad (4)$$

Here, b_i is the bandwidth assigned to UE_i . $p^{nlos} = 1 - p^{los}$ is the probability of non-LoS connection between UE_i and the DBS. p_i is the communication power of UE_i and σ^2 is the noise power. The path loss between the UE_i and MBS can be written as:

$$\varphi_{im} = \zeta_{im} + \pi_{im} \log_{10}(\sqrt{l_{im}^2 + h_{im}^2}), \quad (5)$$

where ζ_{im} and π_{im} are the path loss at the reference distance and path exponent between UE_i and the MBS, respectively. h_{im} and l_{im} are the vertical and horizontal distance between UE_i and the MBS, respectively. The data rate of UE_i toward the MBS is expressed as:

$$R_{im} = b_i \log_2(1 + \gamma_{im}), \quad (6)$$

where $\gamma_{im} = p_i \varphi_{im} / \sigma^2$ is the SNR from UE_i to the MBS. Let j indicates whether UE_i is associating with MBS or DBS.

Thus, the data rate of UE_i when associating with base station j can be summarized as:

$$R_{ij} = \begin{cases} R_{im}, & j = 1, \\ R_{id}, & j \neq 1. \end{cases} \quad (7)$$

B. Computing Model

Both the DBS and MBS are assumed equipped with servers to provide computing service, and the servers are assumed to be multiple cores capable of processing multiple tasks simultaneously. The server in the MBS is more powerful than that mounted on the DBS. The task of UE_i can be executed either by the server mounted on the DBS or by the server equipped on the MBS. Let d_i be the data size of UE_i . C_d and C_m are the computing capacities of the DBS and MBS, respectively. The time for UE_i to complete its tasks by BS j , denoted as T_{ij} , includes the propagation time and computing time:

$$T_{ij} = \frac{d_i}{R_{ij}} + \frac{r_i d_i}{C_i}, \quad (8)$$

where r_i is the number of CPU cycles required to compute each bit of task d_i and C_i is the computing resources assigned to UE_i . Then, the completion time of all UEs can be calculated as: $W = \sum_{ij} T_{ij} \omega_{ij}$, where ω_{ij} is a binary indicator representing whether UE_i is associated with DBS or MBS.

C. Received Laser Power Model

The laser transmission between the laser power source and the DBS is assumed to be free space optical propagation. The received laser power can be expressed as [10]:

$$P_r = P_r G_r G_t \tau_r \tau_t \tau_e P, \quad (9)$$

where P_t is the transmission power of the laser source; G_t and G_r are the transmitter and receiver gain. Here, $G_t = 16/\Theta^2$, where Θ is the divergence angle of the laser. $G_r = (\pi D/\lambda)^2$, where D is the diameter of the receiver and λ is the wavelength of the laser beam; τ_t and τ_r are the transmitter and receiver efficiency; the environmental attenuation τ_e is calculated by $10^{-\alpha L/10}$. The environmental factor α equals to 6.9 dB/km and 0.19 dB/km in foggy weather and clear weather, respectively. L is the distance between the laser power source and the DBS; $P = 1/16 \cdot (\pi L/\lambda)^{-2}$ is the free space optical path loss between the transmitter and receiver [11]. We substitute the parameters and re-express Eq. (9) as:

$$P_r = \tau_r \tau_t P_t (\Theta L/D)^{-2} 10^{-\alpha L/10}. \quad (10)$$

D. Charging Model

Assume that a single battery is used to support the DBS services. The quantity of electrical charge in the fully charged battery is denoted as Q . According to the Coulomb counting method, the state of charge of the battery can be written as [23]:

$$S(t) = \frac{Q - \int_0^t I(\xi) d\xi}{Q}, \quad (11)$$

where $\int_0^t I(\xi)d\xi$ represents the amount of charge consumed by the DBS from 0 to t seconds.

The remaining available power in the battery at time t can be written as [24]:

$$Q(t) = S(t)Q. \quad (12)$$

The remaining DBS service time at time t can be calculated by:

$$T(t) = \frac{\eta_e \eta_m}{I^d} Q(t), \quad (13)$$

where η_e and η_m are the discharging efficiency of the battery and the motor efficiency of the DBS. I^d denotes the DBS average working current, which can be obtained by the motor specification of the DBS.

We assume that the laser constantly provides energy to prolong DBS service time; the quantity of available electrical charge at time t while charging can be derived as the initial quantity of the electrical charge minus the quantity of charge consumed by the DBS hovering and movement plus the additional quantity of charge converted from the laser energy. Thus, the remaining DBS service time while charging can be calculated as:

$$T(t) = \frac{\eta_e \eta_m}{I^d} Q(t) + \frac{\eta_e \eta_m \eta_c}{V I^d} \int_0^t P_r(\xi) d\xi. \quad (14)$$

Here, V and η_c denote the working voltage of the DBS and the converting efficiency of the energizing circuit, which can both be obtained from the motor specification and energizing circuit specification, respectively. The first part of Eq. (14) is the remaining service time of the DBS after discharging. The second part of the Eq. (14) is the extended service time transformed from the harvested laser energy, where $\int_0^t P_r(\xi) d\xi$ represents the laser energy received by the DBS from 0 to t seconds.

Owing to the low mobility of the DBS, we assume that the harvest laser power remains the same within a time slot. Thus, we define $F[n]$ as the remaining service time at the end of the n -th time slot. Therefore, we can discretize and rewrite Eq. (14) as:

$$F[n+1] = F[n] - F^{used}[n+1] + F^{charged}[n+1]. \quad (15)$$

Here, the service time consumption caused by the hovering and movement of the DBS at the end of $(n+1)$ -th time slot can be calculated by:

$$F^{used}[n+1] = \tau_0 + \frac{\varepsilon[n+1]}{I^d V}, \quad (16)$$

where $\varepsilon[n+1] = \frac{M}{2} v^2[n+1]$ [25] denotes the kinetic energy consumption of the DBS in the $(n+1)$ -th time slot due to the movement of the DBS. Here, M and v are the mass and speed of the DBS, respectively. Thus, the service time reduction due to the DBS movement is calculated by $\frac{\varepsilon[n+1]}{I^d V}$. τ_0 represents the duration of a DBS service period.

The extended DBS service time converted from laser energy at the end of $(n+1)$ -th time slot can be calculated as:

$$F^{charged}[n+1] = \frac{\eta_e \eta_m \eta_c \tau_0}{V I^d} P_r[n+1]. \quad (17)$$

$P_r[n+1]$ represents the received laser energy in time slot $(n+1)$. $F[0]$ denotes the original service time of the DBS,

which can be obtained experimentally. For example, we can calculate the average DBS hovering time without charging.

The DBS maintains low mobility because the UEs will not change their locations frequently. The DBS service time reduction caused by DBS movement within a service period is negligible as compared to the service period. As a result, the DBS service time at the end of the $(n+1)$ -th time slot can be calculated by the remaining available service time at the end of the n -th time slot minus the reduced service time at the end of the $(n+1)$ -th time slot, plus the extended service time at the end of the $(n+1)$ -th time slot.

IV. PROBLEM FORMULATION

In this article, our objective is to maximize the DBS's service time and minimize the total task completion time of all UEs. In formulation \mathcal{P}_1 , $Z[n]$ is a binary variable to indicate whether the DBS has enough energy to serve time slot n . The summation of $Z[n]$ is the number of time slots the DBS has served UEs. As a result, it is equivalent to maximize the summation of $Z[n]$ in order to maximize the DBS service time. $W[n]$ is the time for completing tasks of all UEs at time slot n . Thus, we formulate the following multi-objective optimization problem to solve the ACCRUAL problem:

$$\begin{aligned} \mathcal{P}_1 : & \quad \max_{X[n], Y[n], b_i[n], C_i[n], \omega_{ij}[n]} \sum_n Z[n] \\ & \quad \min_{X[n], Y[n], b_i[n], C_i[n], \omega_{ij}[n], Z[n]} \sum_n W[n] \\ s.t. : & \quad C1 : \omega_{ij}[n] \leq Z[n], \forall i \in \{1, 2, \dots, I\}, \\ & \quad \forall n \in \{1, 2, \dots\}, \forall j \in \{1, 2\} \\ & \quad C2 : \sum_{j=1}^J \omega_{ij}[n] = 1, \forall i \in \{1, 2, \dots, I\}, \forall j \in \{1, 2\} \\ & \quad C3 : T_{ij} \omega_{ij}[n] \leq D_i, \forall i \in \{1, 2, \dots, I\}, \forall j \in \{1, 2\} \\ & \quad C4 : \sum_{i=1}^I \omega_{i2}[n] C_i[n] \leq C_d, \forall n \in \{1, 2, \dots\} \\ & \quad C5 : \sum_{i=1}^I \omega_{i1}[n] C_i[n] \leq C_m, \forall n \in \{1, 2, \dots\} \\ & \quad C6 : \sum_{i=1}^I \omega_{i2}[n] b_i[n] \leq \beta_d, \forall n \in \{1, 2, \dots\} \\ & \quad C7 : \sum_{i=1}^I \omega_{i1}[n] b_i[n] \leq \beta_m, \forall n \in \{1, 2, \dots\} \\ & \quad C8 : F[n] - F^{th} \geq (Z[n] - 1) \cdot f_0 \\ & \quad C9 : F[n] - F^{th} < Z[n] \cdot f_0 \\ & \quad C10 : 0 \leq X[n] \leq X_{max} \\ & \quad C11 : 0 \leq Y[n] \leq Y_{max} \end{aligned} \quad (18)$$

Here, $X[n]$ and $Y[n]$ are the location of the DBS at time slot n . $b_i[n]$ and $C_i[n]$ are the bandwidth and computing resource allocated to UE_i at time slot n . $\omega_{ij}[n]$ indicates with which base station UE_i is associated at time slot n . D_i is the deadline requirement of UE_i . C_d and C_m are the

computing resource capacities of the DBS and MBS. β_d and β_m are the bandwidth capacities of the DBS and MBS. F^{th} is the service time threshold of the DBS. X_{max} and Y_{max} are the maximum distance the DBS can travel on the horizontal plane. C1 is the battery power constraint, which imposes the DBS to serve UEs with enough battery power. C2 is the UE association constraint, which ensures one UE can associate with only one station. C3 is the UE task deadline constraint, which imposes the tasks to be completed in time. C4-C7 are the resource capacity constraints. C4-C5 are the computing resource constraints, which impose the computing resources allocated to UEs not to exceed the respective computing capacity of the MEC servers on the DBS and MBS. C6-C7 are the bandwidth resource constraints, which impose the bandwidth allocated to the ground UEs not to exceed the respective bandwidth resource capacity of the MBS and DBS. C8-C9 are the DBS energy constraints, where f_0 is a large positive number. Considering C8 and C9 together, $Z[n] = 1$ if the DBS has enough energy to return to the charging station at time slot n ; otherwise $Z[n] = 0$. As a result, C8 and C9 impose the DBS to retain adequate energy to fly back to the charging station. C10-C11 are the DBS placement constraints on the horizontal plane.

V. PROPOSED SOLUTION

The ACCRUAL problem is a mixed integer nonlinear problem due to the binary UE association indicator, the bandwidth and the computing resource allocation, and it is thus a very difficult problem. In order to solve the ACCRUAL problem, the original problem is decomposed into two sub-problems. The first sub-problem is the REAP problem. The second sub-problem is the DBS placement problem. To tackle the problem, the DBS is first placed in the middle of the ground UEs and allocated computing resources and bandwidth for UEs to achieve the minimum task completion time. Then, the DBS tries to find a location where it can maintain the total task completion time while maximizing the service time.

A. The REAP Problem

Note that \mathcal{P}_1 is a mixed integer nonlinear multi-objective problem which is difficult to solve. We can implement ϵ -Constraint Method, which is to incorporate one objective into the constraints. However, it is very difficult to determine the upper bound of the objective ϵ . Different ϵ will lead to completely different target results. Same as the weighted sum method, different weights of the objective can lead to totally different results. To tackle the problem, we decompose \mathcal{P}_1 into two sub-problems, the REAP problem and the DBS placement problem. For a given DBS placement, we first solve the REAP problem:

$$\begin{aligned} \mathcal{P}_2 : & \min_{X[n], Y[n], b_i[n], C_i[n], \omega_{ij}[n], Z[n]} \sum_n W[n] \\ s.t. : & \\ & C1 - C7 \end{aligned} \quad (19)$$

The REAP problem focuses on minimizing the completion time of the system with limited bandwidth and computing

resources. It is a mixed integer nonlinear programming, which is difficult to solve. To tackle \mathcal{P}_2 , we further decompose it into two sub-problems: the bandwidth and computing resource allocation problem and the UE association problem.

1) *Bandwidth and computing resource allocation*: For a given UE association, we try to assign the bandwidth and computing resources such that the deadline is satisfied. \mathcal{P}_2 is simplified as:

$$\begin{aligned} \mathcal{P}_3 : & \min_{b_i[n], C_i[n]} \sum_n W[n] \\ s.t. : & \\ & C3 - C7 \end{aligned} \quad (20)$$

Lemma 1. For a given UE association, \mathcal{P}_3 is convex.

Proof: Denote the Hessian Matrix of T_{ij} as $H = \begin{bmatrix} \frac{\partial^2 T}{\partial b_i^2} & \frac{\partial^2 T}{\partial b_i \partial C_i} \\ \frac{\partial^2 T}{\partial C_i \partial b_i} & \frac{\partial^2 T}{\partial C_i^2} \end{bmatrix} = \begin{bmatrix} \frac{2d_i}{b_i^3 \log(1+\gamma_i)^3} & 0 \\ 0 & \frac{2r_i d_i}{C_i^3} \end{bmatrix}$. Since r_i, d_i, b_i and C_i are all larger than 0, the Hessian Matrix H is positive definite, implying that T_{ij} is convex for a given UE association. Since the linear combination of convex functions is still convex, we can conclude that \mathcal{P}_3 is convex for a given UE association. ■

Since \mathcal{P}_3 is convex, we can easily solve it using CVX.

2) *UE association*: Given the resource allocation strategy, the UE association problem can be mapped into a multiple-choice two-dimensional knapsack problem. A UE can be mapped into an item, the required bandwidth and computational resource can be mapped into the weight and the volume, the MBS and DBS can be mapped into two knapsacks, and the inverse of the task completion time can be mapped into the profit. The objective is to maximize the profit. To solve the association problem, we first define the profit of UE_i in associating with BS j as:

$$\varpi_{ij} = \frac{1}{T_{ij}} = \frac{1}{\frac{d_i}{R_{ij}} + \frac{r_i d_i}{C_i}}, \quad (21)$$

such that the profit ϖ_{ij} will decrease if the task completion time of UE_i in associating with BS j increases. By implementing the profit ϖ_{ij} , we transform the minimization problem into a maximization problem. Then, to solve the problem, we define C as the K -dimensional capacity constraints vector. Here, K is the number of capacity constraints including the bandwidth and computing resource constraints of the MBD and DBS. Let C^k be the k^{th} capacity constraint. We further define the required resource represented by a K -dimensional weight vector Φ_{ij} . Here, Φ_{ij}^k is the k^{th} required resource of UE_i associated with BS j . After having solved \mathcal{P}_3 , we try to obtain better UE association by solving the multi-choice multi-dimensional Knapsack Problem \mathcal{P}_4 :

$$\begin{aligned} \mathcal{P}_4 : P(\omega) = & \max_{\omega_{ij}[n], Z[n]} \sum_{i=1}^I \sum_{j=1}^2 \varpi_{ij}[n] \omega_{ij}[n] \\ s.t. : & \\ & C1 \ \& \ C2 \end{aligned}$$

$$C14 : \sum_{i=1}^I \sum_{j=1}^2 \Phi_{ij}^k [n] \omega_{ij} [n] \leq C^k \quad (22)$$

Here, ϖ_{ij} is the profit of UE_i to be served by BS j . C_i and b_i are the two costs of UE_i . The MBS and DBS are the two knapsacks. We want to maximize the total profit by putting UEs into two knapsacks (MBS vs DBS) without exceeding their respective capacities.

By letting the non-negative real vector λ correspond to the capacity constraint C14, \mathcal{P}_4 can be relaxed to the following Lagrangian:

$$\begin{aligned} \mathcal{P}_5 : L(\lambda) = & \max_{\omega_{ij}[n], Z[n]} \sum_{i=1}^I \sum_{j=1}^2 \varpi_{ij} [n] \omega_{ij} [n] \\ & - \sum_{k=1}^K \lambda_k \left(\sum_{i=1}^I \sum_{j=1}^2 \Phi_{ij}^k [n] \omega_{ij} [n] - C^k \right). \\ \text{s.t. :} \\ C1 \quad \& \quad C2. \end{aligned} \quad (23)$$

Here, λ is a K -dimensional Lagrange multiplier vector and λ_k is the Lagrange multiplier correspond to the k^{th} capacity constraint.

Lemma 2. For a given $\tilde{\lambda} \geq 0$, we can calculate the corresponding upper bound of \mathcal{P}_4 by solving $\mathcal{P}_5 : L(\tilde{\lambda})$.

Proof: Denote the optimal objective function value of \mathcal{P}_4 as z^* . Since the capacity constraint C14 of \mathcal{P}_4 is removed and $\sum_{k=1}^K \tilde{\lambda}_k (\sum_{i=1}^I \sum_{j=1}^2 \Phi_{ij}^k [n] \omega_{ij} [n] - C^k) \leq 0$. We have $L(\tilde{\lambda}) \geq z^*$. ■

By combining similar items in \mathcal{P}_5 , we get \mathcal{P}_6 . For a given $\tilde{\lambda} \geq 0$, we can calculate the corresponding upper bound by solving \mathcal{P}_6 :

$$\begin{aligned} \mathcal{P}_6 : L(\tilde{\lambda}) = & \max_{\omega_{ij}[n], Z[n]} \sum_{i=1}^I \sum_{j=1}^2 (\varpi_{ij} [n] - \sum_{k=1}^K \tilde{\lambda}_k \Phi_{ij}^k [n]) \\ & \cdot \omega_{ij} [n] + \sum_{k=1}^K \tilde{\lambda}_k C^k \\ \text{s.t. :} \\ C1 \quad \& \quad C2 \end{aligned} \quad (24)$$

Obviously for UE_i , given Lagrange multiplier vector $\tilde{\lambda}$, the summation of capacity constraints $\sum_{k=1}^K \tilde{\lambda}_k C^k$ is a constant. Considering C1 and C2, to maximize \mathcal{P}_6 , we just need to maximize $\sum_{i=1}^I \sum_{j=1}^2 (\varpi_{ij} [n] - \sum_{k=1}^K \tilde{\lambda}_k \Phi_{ij}^k [n]) \cdot \omega_{ij} [n]$. Thus, $\omega_{ij} = 1$ for UE_i to select BS j that maximizes $\varpi_{ij} [n] - \sum_{k=1}^K \tilde{\lambda}_k \Phi_{ij}^k [n]$; if $\varpi_{ij} [n] - \sum_{k=1}^K \tilde{\lambda}_k \Phi_{ij}^k [n] = 0, \forall j \in J, UE_i$ can select any BS.

We denote z_L as the value of $L(\lambda)$; z_L is minimized by the sub-gradient method from which the optimal Lagrange multiplier vector, denoted as λ^* , can be derived. Let

$$\Omega = \{\omega^t \text{ subject to C1 and C2}\}$$

be the finite set of feasible solutions of $L(\lambda)$, where $\lambda \geq 0$ and $\omega^t = (\omega_{1,1}^t, \dots, \omega_{i,j}^t, \dots, \omega_{I,2}^t)$ be the feasible solution at the

t^{th} iteration. The dual problem, which is to find the tightest upper bound of \mathcal{P}_4 , can be written as:

$$\begin{aligned} \mathcal{P}_7 : z_L(\lambda) = & \min_{\lambda} u \\ \text{s.t. :} \\ u \geq & \sum_{i=1}^I \sum_{j=1}^2 \varpi_{ij} [n] \omega_{ij}^t [n] + \sum_{k=1}^K \lambda_k g^t \\ \lambda_k \geq & 0, \forall k \in K \end{aligned} \quad (25)$$

where $g^t = (g_1^t, \dots, g_K^t)$ is a K -dimensional vector at the t^{th} iteration such that $g_k^t = \sum_{i=1}^I \sum_{j=1}^2 C^k - \Phi_{ij}^k [n] \omega_{ij}^t [n]$.

Theorem 1. The function $L(\lambda)$ of λ is piecewise-linear.

Proof: Denote S as any subset of UEs, and $a(S)$ and $c(S)$ denote the total required resources and the total profit of UEs in S , respectively. Then, we have $L(\lambda) = \max_{S \in \Omega} \{c(S) - \lambda(a(S) - C)\}$. Since $c(S) - \lambda(a(S) - C)$ is a linear function of λ for a given subset, we find that $L(\lambda)$ is the maximum of a finite set of linear functions, and thus it is piecewise-linear. ■

Since $L(\lambda)$ is piecewise-linear, it is not differentiable at some points. We use the sub-gradient method [29] to find the optimal λ that leads to the tightest upper bound. Recall that the K -dimensional vector g^t is a sub-gradient of ω^t on $\tilde{\lambda}$ for function $z_L(\lambda)$ when the condition

$$z_L(\lambda) \leq z_L(\tilde{\lambda}) + g^t (\lambda - \tilde{\lambda}), \forall \lambda \geq 0 \quad (26)$$

is satisfied, where ω^t is the optimal solution of $L(\tilde{\lambda})$.

The optimal Lagrange multiplier λ^* can be approached iteratively by generating a sequence of λ according to Inequality (26). Given an initial λ^1 , the sequence $\{\lambda^t\}$ can be determined iteratively by:

$$\lambda^{t+1} = \lambda^t + \gamma_s g^t, \quad (27)$$

where γ_s is the step size. λ^t is the Lagrange multiplier at the t^{th} iteration.

Lemma 3. The optimal Lagrange multiplier λ^* can be approached iteratively.

Proof: $\|\lambda^{t+1} - \lambda^*\|^2 = \|\lambda^t + \gamma_s g^t - \lambda^*\|^2$
 $= \|\lambda^t - \lambda^*\|^2 + \gamma_s^2 \|g^t\|^2 + 2\gamma_s g^t (\lambda^t - \lambda^*)$
 $\leq \|\lambda^t - \lambda^*\|^2 + \gamma_s^2 \|g^t\|^2 + 2\gamma_s (z_L(\lambda^t) - z_L(\lambda^*))$
 To guarantee $\|\lambda^{t+1} - \lambda^*\|^2 - \|\lambda^t - \lambda^*\|^2 \leq 0$, we should ensure $\gamma_s^2 \|g^t\|^2 + 2\gamma_s (z_L(\lambda^t) - z_L(\lambda^*)) < 0$. Thus, we can conclude that as long as:

$$\gamma_s \in \left(0, \frac{2(z_L(\lambda^t) - z_L(\lambda^*))}{\|g^t\|^2}\right), \quad (28)$$

λ^* can be progressively approached. ■

In practice, we set $z_L(\lambda^*)$ as the lower bound of \mathcal{P}_3 . For example, we can calculate the lower bound by the greedy method. The steps are summarized in Algorithm 1.

In Algorithm 2 Iterative Allocation for REAP (IA-REAP), we denote m as the number of iterations, the resource allocation of \mathcal{P}_3 as a and the objective value of the REAP problem as $W(\omega, a)$. Since \mathcal{P}_3 is convex for given UE association ω , we have $W(\omega^m, a^m) \geq W(\omega^m, a^{m+1})$. As

Algorithm 1: UE Association for REAP

Input: $K, \epsilon, \beta_d, \beta_m, UE, C_d, C_m, X[n], Y[n]$.
Output: $\omega^t, W[n]^*$.

- 1 Calculate b_i and C_i by solving \mathcal{P}_3 .
- 2 Initialize λ^t and set $t = 1$.
- 3 **repeat**
- 4 Obtain ω^t by solving \mathcal{P}_6 .
- 5 Calculate the sub-gradient g^t according to Eq.(26).
- 6 $\lambda^{t+1} = \lambda^t + \gamma_s g^t$.
- 7 Choose γ_s according to Eq.(28).
- 8 $t = t + 1$.
- 9 **until** $P(\omega^{t+1}) - P(\omega^t) \leq \epsilon$.

Algorithm 2: Iterative Allocation for REAP (IA-REAP)

Input: $K, \epsilon, \beta_d, \beta_m, UE, C_d, C_m, X[n], Y[n]$.
Output: $\omega^t, W[n]^*$.

- 1 Initialize $m = 0, \omega^m$ and a^m .
- 2 **repeat**
- 3 Given $\{\omega^m, a^m\}$, find the optimal resource allocation a^{m+1} by solving \mathcal{P}_3
- 4 Given $\{\omega^m, a^{m+1}\}$, find the optimal UE association ω^{m+1} by Algorithm 1.
- 5 $m = m + 1$.
- 6 **until** the improvement is less than ϵ .

proven before, for given resource allocation a^{m+1} , we have $W(\omega^m, a^{m+1}) \geq W(\omega^{m+1}, a^{m+1})$. Therefore, we can conclude that $W(\omega^m, a^m) \geq W(\omega^{m+1}, a^{m+1})$.

B. DBS Placement Problem

Given the UE association and resource allocation, we try to solve the DBS placement problem:

$$\begin{aligned} \mathcal{P}_8 : \quad & \max_{X[n], Y[n]} \sum_n Z[n] \\ \text{s.t. :} \quad & \\ & C8 - C11 \end{aligned} \quad (29)$$

Since \mathcal{P}_8 is highly non-linear, to tackle the DBS placement problem, the horizontal plane is divided into several sub-planes to determine the DBS placement. We first calculate the completion time of all tasks at each sub-plane. Second, the sub-planes are sorted in ascending order according to the total task completion time. In Algorithm 3, the plane is divided into L sub-planes. $W_l[n]$ is the completion time of all tasks at the l -th sub-plane. $W[n]^*$ is the completion time of all tasks obtained from Algorithm 2. In lines 1-5, the completion time of all tasks at each sub-plane are rounded down. In lines 6-11, the maximum and minimum completion times among all sub-planes are found. $range$ is the variance of the task completion time among all sub-planes. $count$ is an auxiliary array in lines 12-14; it stores the number of sub-planes for each completion time. $index$ is used to count how many blocks can provision task completion time of $W_l[n]$. In lines 15-20, the blocks are sorted according to the completion time stored in

Algorithm 3: Counting Placement (CP)

Input: $\omega_{ij}, X_{max}, Y_{max}, F[n], F^{th}, W[n]^*$.
Output: X^*, Y^*, Z_n .

- 1 **for** $X \in X_{max}$ and $Y \in Y_{max}$ **do**
- 2 \lfloor obtain $W_l[n]$ for every X and Y by Algorithm 2.
- 3 **for** $l \leq L$ **do**
- 4 \lfloor $W[n]^* = \lfloor W[n]^* * 100 \rfloor$.
- 5 \lfloor $W_l[n] = \lfloor W_l[n] * 100 \rfloor$.
- 6 **for** $l \leq L$ **do**
- 7 **if** $W_l[n] > \max$ **then**
- 8 \lfloor $\max = W_l[n]$.
- 9 **if** $W_l[n] < \min$ **then**
- 10 \lfloor $\min = W_l[n]$.
- 11 $range = \max - \min + 1$.
- 12 **for** $l \leq L$ **do**
- 13 $index = W_l[n] - \min$.
- 14 $count(index) = count(index) + 1$.
- 15 $index = 1$.
- 16 **for** $l \leq range$ **do**
- 17 **while** $count(l) \geq 1$ **do**
- 18 $W_{index}[n] = l + \min$.
- 19 $index = index + 1$.
- 20 $count(l) = count(l) - 1$.
- 21 $l = L$.
- 22 **for** $l > \frac{3}{4}L$ **do**
- 23 **if** $W[n]^* \leq W_l[n]$ and $F \leq \dot{F}$ **then**
- 24 \lfloor $\max F = \dot{F}$.
- 25 \lfloor $l = l - 1$.
- 26 $(X^*, Y^*) = \arg \max F(X_l, Y_l)$.
- 27 **if** $F[n] - F^{th} > 0$ **then**
- 28 \lfloor $Z[n] = 1$.
- 29 **else**
- 30 \lfloor $Z[n] = 0$.
- 31 \lfloor $\omega_{ij} = 0 \forall i \in I$.

$count$. Here, $count$ stores the positions where the sub-planes with each completion time should be placed. Then, in lines 21-26, we compare $W[n]^*$ and total completion time of the sub-planes. The DBS will move to the location where it can maintain the completion time $W[n]^*$ obtained from Algorithm 2 and maximize the service time of the DBS. In lines 27-31, $Z[n] = 0$ if the DBS does not retain adequate battery power to fly back to the MBS charging station at the end of time slot n , in which case all ground UE association indicators ω_{ij} will be set to 0. The time complexity of Algorithm3 (CP) depends on the number of sub-planes. The complexity to calculate the task completion time at all candidate sub-planes is $O(L \cdot m)$, where m is the number of iterations of Algorithm 2. The complexity to sort sub-planes is $O(L + range)$. The complexity to compare the service time of the candidate sub-planes is $O(\frac{1}{4}L)$. Thus, the total complexity of the CP algorithm is $O((\frac{3}{4} + m)L + range)$.

VI. PERFORMANCE EVALUATION

Extensive simulations are run using MATLAB to obtain the results. 40 UEs are uniformly distributed in a $500 \times 500 \text{ m}^2$ area for every simulation. To simulate the scenario that the DBS helps serve the UEs far away from the MBS, the MBS is placed at the origin $(0,0)$ of the Cartesian coordinate. A laser power source is mounted on the MBS. The ground UEs are uniformly distributed at $x \in [0, 500]$ and $y \in [0, 500]$. The DBS is first placed at $(\sum_{i=1}^I \frac{x_i}{I}, \sum_{i=1}^I \frac{y_i}{I})$ and gradually modifies its location in every time slot. The raspberry pi 4 mounted on the drone is responsible for executing computing tasks. The quad core CPU on the raspberry pi 4 can overclock to 2.3 GHz [34], [38]. So, the computational capacity of the DBS is set to 9.2 GHz. According to Federal Communications Commission, the specific absorption rate for mobile devices such as cell phones is 1.6 watts per kilogram (W/kg) [36]. The recent smartphones weight ranking report shows that the weight of the smartphone ranges from 112 g to 328 g [37]. So, the communication power of each UE is distributed within $[100, 500]$ mW. The task size of each UE is distributed within $[0.1, 0.5]$ Mb [32]. The deadline of every task is distributed within $[0.2, 1]$ s. The working current and voltage of the DBS are 23V and 5A, respectively. The original DBS service time without charging is 1800s. The original DBS service time is divided into 15 slots, considering the low mobility of the DBS. The other simulation parameters are illustrated in Table 1.

TABLE 1. Simulation Parameters

(a, b)	$(9.1, 0.16)$ [27]	(ξ^{Los}, ξ^{Nlos})	$(1, 20)$
$(\tau^{Los}, \tau^{Nlos})$	$(20, 20)$	(ξ_{im}, τ_{im})	$(131.1, 42.8)$
N_0	-174 dbm/Hz	(β_m, β_d)	$(20, 5)$ MHz
(C_m, C_d)	$(50, 9.2)$ GHz [33]	v	18 m/s
P_i	$[100, 500]$ mW	P_t	200 W
d_i	$[0.1, 0.5]$ Mb [32]	D [35]	1.5
θ	0.01	α	6.9db/km [28]
I^d	5A	V	23V
M	3.6kg [7]	η_m	0.85
η_e	0.95	D_i	$[0.2, 1]$ s [32]
F_0	1800 s	τ_0	120 s
I	40	(X_{max}, Y_{max})	$(500, 500)$ m

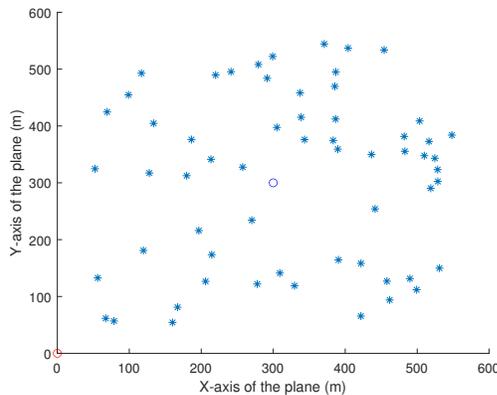


Fig. 2. Distribution of UEs.

Fig. 2 shows the distribution of UEs in the horizontal plane. The red circle is the location of the MBS and the blue circle is the location of the DBS. Fig. 3 demonstrates how the total

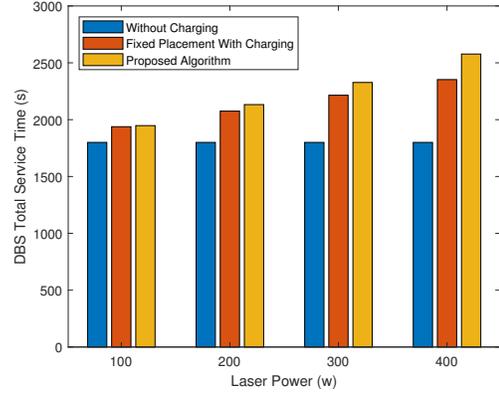


Fig. 3. Total service time with different laser source power.

service time of the DBS increases as the laser source power increases. The blue columns show the initial DBS service time without charging. The orange columns illustrate the service time with the fixed placement algorithm, in which the DBS is located in the middle of all ground UEs in every time slot. The khaki columns illustrate the DBS service time achieved by our proposed algorithm. The simulation results demonstrate that our algorithm is superior to the fixed placement algorithm because the proposed algorithm enables the DBS to harvest more energy. When the laser source power is 200w, 20% of the DBS service time is improved as compared to the service time without charging. As compared with the fixed placement algorithm, the service time is improved by more than 3%. When the laser source power is 400w [26], our proposed algorithm improves the service time by 16% as compared to the fixed placement algorithm and by 40% of the service time as compared to that without charging.

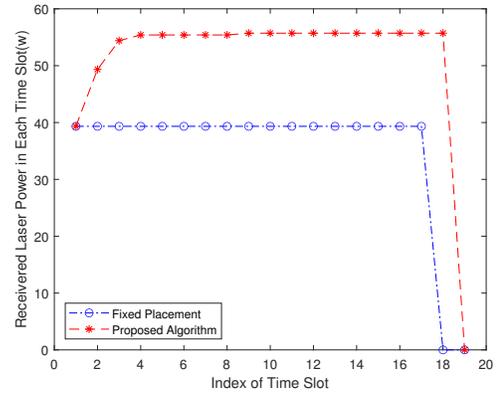


Fig. 4. Received power.

Fig. 4 illustrates the laser power received by the DBS. The laser source transmission power is 200w. The UE distribution is shown in Fig. 2. The blue line corresponds to the fixed placement. As the DBS is located in the middle of the ground UEs, the received power is almost the same in every time slot. At the 17th time slot, even with the laser charging energy, the DBS does not have sufficient battery energy to serve the next time slot. So, the DBS returns to the charging station. As a result, the received power at the 18th time slot becomes 0. The red curve corresponds to our proposed algorithm. The DBS

tries to find a location where it can both maintain the total task completion time calculated by Algorithm 2 and harvest the maximum energy. The result shows that the received power grows fast from time slot 1 to time slot 4 because the DBS is flying towards the charging station. From time slot 4 to time slot 18, the location of the DBS is almost fixed. The DBS is searching for a location where it can both extend the service time and maintain the task completion time obtained from Algorithm 2. Here, extending the DBS service time means the DBS has to move closer to the MBS, and maintaining the task completion time means the DBS may need to fly away to provide services to the UEs far from the MBS. In Algorithm 3 (lines 23-26), only when both conditions are fulfilled, will the DBS move. The two conditions conflict with each other. Since the DBS has to serve UEs and fulfill the deadlines of the tasks of UEs, the DBS cannot fly to the charging station as close as it wants. So, the DBS remains almost stationary from time slot 4 to 18, and the received power grows slowly and steadily. At time slot 18, the DBS does not have sufficient battery energy to provide services in the next time slot. Thus, the received laser power becomes 0 at time slot 19.

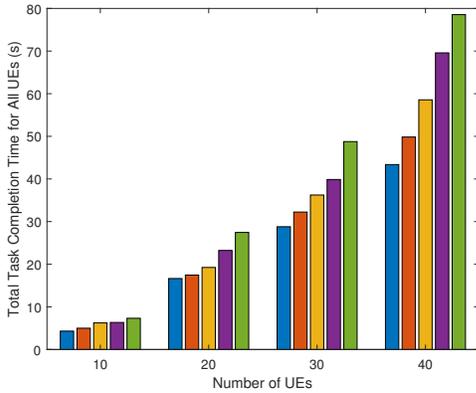


Fig. 5. Total task completion time for all UEs

Fig. 5 illustrates the time for completing tasks of all UEs for different numbers of UEs by implementing different algorithms. The UE distribution is shown in Fig. 2. The blue bar corresponds to our proposed algorithm and the orange bar to the greedy algorithm. The UEs try to offload their tasks to the base station which can provide better channel conditions. Since the DBS has a better channel condition most of the time, most tasks will be offloaded to the DBS until the DBS runs out of resources, and then the rest of the UEs must offload their tasks to the MBS. Although the DBS has better channel conditions, it has limited resources. As a result, more tasks have to be offloaded to the MBS, thus leading to a larger completion time as compared to our proposed algorithm. The khaki bar corresponds to the equal resource assignment algorithm. All the resources are equally allocated to all UEs and the UEs will try to offload their tasks to the DBS until the DBS runs out of resources. Although the equal distribution of resources seems very fair, this policy will greatly increase the delay of some tasks that require a large amount of resources. The purple bar corresponds to the fixed placement random association algorithm. The DBS is located

in the middle of the ground UEs and resources are equally shared among UEs. Tasks are equally likely offloaded to MBS and DBS. The green bar corresponds to the one without DBS assistance, in which all the tasks are offloaded to the MBS. As the result shows, when the number of UEs increases, the total task completion time increases. When there are 20 UEs, the fixed placement algorithm and the non-DBS assistance algorithm are not able to meet the task deadline requirement. When there are 30 UEs, only the proposed algorithm can meet the task deadline requirement. Furthermore, our algorithm is superior to the other algorithms, and more than 9% of the total task completion time is improved as compared to the greedy algorithm when there are 40 UEs.

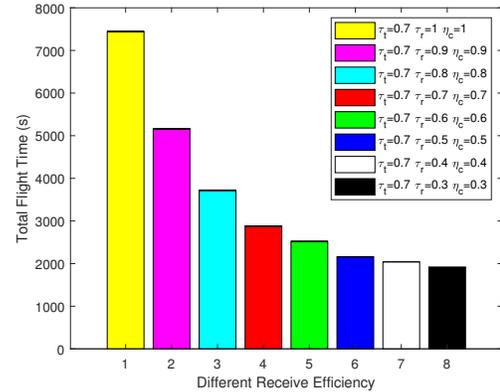


Fig. 6. DBS service time with different receiver efficiency.

Fig. 6 illustrates the total flight time with different receiver efficiencies. In Fig. 6, the y-axis represents the DBS service time and the x-axis corresponds to eight different receiver efficiencies. Here, τ_t is the transmission efficiency of the laser charging station. η_c is the converting efficiency of the charging circuit. τ_r is the energy harvesting efficiency of the receiver. The results show that as the receiver efficiency decreases, the total flight time decreases accordingly. The flight time illustrated by the last column (black) is the same as the flight time without charging. So, in order to extend the DBS service time when the laser power and transmission efficiencies are 200 w and 0.7, the energy harvesting efficiency and converting efficiency should be both at least 0.4.

VII. CONCLUSION

In this article, the joint user association bandwidth and computation resource assignment and laser charging (ACCRUAL) problem has been formulated. Both DBS and MBS are deployed to provide computing services to the ground UEs. A laser power source is mounted on the MBS to energize the DBS. Our objective is to jointly minimize the task completion time and maximize the DBS services time. We have decomposed the ACCRUAL problem into two sub-problems: the REAP problem and the DBS placement problem. An iterative algorithm has been proposed to solve the REAP problem to minimize the completion time with limited bandwidth and computing resources as well as the user association. A method to estimate the DBS service time has been developed. Implementing IA-REAP and CP algorithms has been

shown to extend the DBS service time. The simulation results demonstrate that when the laser power is 200w, our proposed algorithm can improve the DBS service time by 20% as compared to the service time without charging. When the laser power is 400w, our proposed algorithm can improve the DBS service time by 40% as compared to the service time without charging and 16% of service time is improved as compared to the fixed placement algorithm.

REFERENCES

- [1] T. Kan, Y. Chiang and H. Wei, "Task offloading and resource allocation in mobile-edge computing system," *WOCC*, 2018, pp. 1-4.
- [2] Knud Lasse Lueth "State of the IoT 2020" <https://iot-analytics.com/state-of-the-iot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/>
- [3] S. O'Dea "Global unique mobile subscribers from 2010 to 2025" <https://www.statista.com/statistics/740154/worldwide-unique-mobile-subscribers-by-region/>
- [4] S. Zhang and N. Ansari, "3D Drone Base Station Placement and Resource Allocation With FSO-Based Backhaul in Hotspots," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3322-3329, March 2020.
- [5] S. Zhang and N. Ansari, "Latency Aware 3D Placement and User Association in Drone-assisted Heterogeneous Networks with FSO-Based Backhaul," *IEEE Trans. Veh. Technol.*, doi: 10.1109/TVT.2021.3112483. <https://baofengtech.com/usermanual/X-Series-User-Guide.pdf>.
- [6] <https://www.dji.com/matrice600-pro>.
- [7] "Communications Status Report for Areas Impacted by Hurricane Maria," *Federal Communications Commission*, Jan. 2018. <https://docs.fcc.gov/public/attachments/DOC-348974A1.pdf>.
- [8] <https://arstechnica.com/information-technology/2017/11/att-drone-brings-lte-access-to-hurricane-damaged-puerto-rico/>.
- [9] Majumdar, A. "Free-space laser communication performance in the atmospheric channel," *J Optic Comm Rep* 2, 345-396 (2005).
- [10] N. Ansari, Q. Fan, X. Sun, and L. Zhang, "SoarNet," *IEEE Trans. Wireless Commun.*, vol. 26, no. 6, pp. 37-43, Dec. 2019.
- [11] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569-572, Dec. 2014.
- [12] L. Yang et al., "Multi-UAV-Enabled Load-Balance Mobile-Edge Computing for IoT Networks," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6898-6908, Aug. 2020.
- [13] L. Zhang and N. Ansari, "Latency-Aware IoT Service Provisioning in UAV-Aided Mobile-Edge Computing Networks," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10573-10580, Oct. 2020.
- [14] T. M. Mostafa, A. Muharam and R. Hattori, "Wireless battery charging system for drones via capacitive power transfer," *2017 IEEE PELS Workshop on Emerging Technologies: Wireless Power Transfer (WoW)*, Chongqing, 2017, pp. 1-6.
- [15] Q. Wu, Y. Zeng and R. Zhang, "Joint Trajectory and Communication Design for UAV-Enabled Multiple Access," *2017 GLOBECOM*, Singapore, 2017, pp. 1-6.
- [16] J. Ouyang, Y. Che, J. Xu and K. Wu, "Throughput Maximization for Laser-Powered UAV Wireless Communication Systems," *2018 ICC Workshops*, Kansas City, MO, 2018, pp. 1-6.
- [17] M. Jiang, Y. Li, Q. Zhang and J. Qin, "Joint Position and Time Allocation Optimization of UAV Enabled Time Allocation Optimization Networks," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3806-3816, May 2019.
- [18] Y. Sun et al., "Optimal 3D-Trajectory Design and Resource Allocation for Solar-Powered UAV Communication Systems," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4281-4298, June 2019.
- [19] M. Alzenad, A. El-Keyi and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station for Maximum Coverage of Users With Different QoS Requirements," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 38-41, Feb. 2018.
- [20] Q. Fan and N. Ansari, "Towards Traffic Load Balancing in Drone-Assisted Communications for IoT," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3633-3640, April 2019.
- [21] 3GPP TR 36.828 version 11.0.0, release 11, 3GPP Tech. Rep., 2012. URL <http://www.qtc.jp/3GPP/Specs/36828-b00.pdf>.
- [22] J. Chiasson and B. Vairamohan, "Estimating the state of charge of a battery," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 3, pp. 465-470, May 2005.
- [23] W. Zhang, L. Wang, and Li. Wang, "An improved adaptive estimator for state-of-charge estimation of lithium-ion batteries," *Journal of Power Sources*, vol.402, pp.422-433, Oct. 2018.
- [24] Y. Qian, F. Wang, J. Li, L. Shi, K. Cai and F. Shu, "User Association and Path Planning for UAV-Aided Mobile Edge Computing With Energy Restriction," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1312-1315, Oct. 2019.
- [25] U.S. Naval Research Laboratory Laser demonstration [online]. Available: <https://phys.org/news/2019-10-transmit-energy-laser-historic-power-beaming.html>.
- [26] L. Zhang, Q. Fan, and N. Ansari "3-D Drone-Base-Station Placement With In-Band Full-Duplex Communications," *IEEE Commun. Lett.*, vol. 22, no. 9, Sep. 2018.
- [27] M. Alzenad, "FSO-Based Vertical Backhaul/Fronthaul Framework for 5G+ Wireless Networks," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 218-224, Jan. 2018.
- [28] M. Hifi and Lei Wu, "An equivalent Model for Exactly Solving the Multiple-choice Multidimensional Knapsack Problem," *Int. J. Comb. Optim. Probl. Informatics*, vol. 3, pp. 43-58, Oct. 2012.
- [29] M. Moradi, et al., "SkyCore: Moving Core to the Edge for Untethered and Reliable UAV-based LTE networks," *MobiCom '18*, New Delhi, India, Oct. 29 - Nov. 2, 2018, pp. 35-49.
- [30] U. Fattore, et al., "UPFlight: An enabler for Avionic MEC in a drone-extended 5G mobile network," *VTC2020-Spring*, May 25-July 31, 2020, online
- [31] L. Zhang, et al., "Optimizing the operation cost for UAV-aided mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 6085-6093, June 2021.
- [32] J. Zhang, et al., "Joint resource allocation for latency-sensitive services over mobile edgecomputing networks with caching," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4283-4294, Jun.2019.
- [33] Raspberry Pi 4 Tech Specs [Online]. Available: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>
- [34] W. Liu, L. Zhang and N. Ansari, "Laser Charging Enabled DBS Placement for Downlink Communications," *IEEE Trans. Netw. Sci. Eng.*, doi: 10.1109/TNSE.2021.3118328.
- [35] Cell Phones and Specific Absorption Rate [Online]. Available: <https://www.fcc.gov/general/cell-phones-and-specific-absorption-rate>
- [36] Smartphones Weight Ranking [Online]. Available: <https://www.techrankup.com/en/smartphones-weight-ranking/>
- [37] How to overclock Raspberry Pi 4 [Online]. Available: <https://www.tomshardware.com/how-to/raspberry-pi-4-23-ghz-overclock>