

Article

Joint Placement and Device Association of UAV Base Stations in IoT Networks

Ashfaq Ahmed ¹, Muhammad Awais ^{1,*} , Tallha Akram ¹ , Selman Kulac ² ,
Musaed Alhussein ³ and Khursheed Aurangzeb ^{3,*} 

¹ Department of Electrical & Computer Engineering, COMSATS University Islamabad, Wah Campus, Wah Cantt 47040, Pakistan; ashfaqahmed@ciitwah.edu.pk (A.A.); tallha@ciitwah.edu.pk (T.A.)

² Department of Electrical-Electronics Engineering, Faculty of Engineering, Duzce University, Konuralp, Duzce 81620, Turkey; selmankulac@duzce.edu.tr

³ Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia; musaed@ksu.edu.sa

* Correspondence: muhammadawais@ciitwah.edu.pk (M.A.); kaurangzeb@ksu.edu.sa (K.A.)

Received: 12 March 2019; Accepted: 27 April 2019; Published: 9 May 2019



Abstract: Drone base stations (DBSs) have received significant research interest in recent years. They provide a flexible and cost-effective solution to improve the coverage, connectivity, quality of service (QoS), and energy efficiency of large-area Internet of Things (IoT) networks. However, as DBSs are costly and power-limited devices, they require an efficient scheme for their deployment in practical networks. This work proposes a realistic mathematical model for the joint optimization problem of DBS placement and IoT users' assignment in a massive IoT network scenario. The optimization goal is to maximize the connectivity of IoT users by utilizing the minimum number of DBS, while satisfying practical network constraints. Such an optimization problem is NP-hard, and the optimal solution has a complexity exponential to the number of DBSs and IoT users in the network. Furthermore, this work also proposes a linearization scheme and a low-complexity heuristic to solve the problem in polynomial time. The simulations are performed for a number of network scenarios, and demonstrate that the proposed heuristic is numerically accurate and performs close to the optimal solution.

Keywords: resource management; Unmanned aerial vehicles (UAVs); Aerial base stations; Internet of Things (IoTs)

1. Introduction

The use of drone base stations (DBSs) has been realized as a promising addition to the conventional wireless networks. DBSs are adopted to achieve multiple objectives. On one hand, they help to increase the coverage of existing terrestrial networks, such as broadband and cellular networks [1]. On the other hand, they serve as assisting relays to improve the connectivity of ground wireless devices [2]. In contrast to the terrestrial BSs which are fixed, DBSs have the ability to fly and deliver network services to any hard-to-reach areas. In addition to the horizontal manoeuvring, they have the ability to adapt their altitude. This enables them to establish LOS links to ground users [3]. Moreover, they improve the capacity of networks by serving as mobile hotspots. Along with the open-air communication, small drones have also proven to be an effective solution for indoor communications [4].

Internet of Things (IoT) is one exciting application of DBSs. IoT refers to a set of small, uniquely identifiable devices connected to the Internet. Practical IoT networks are composed of a massive number of heterogenous devices, which include smart phones, sensors, electronic gadgets and wearables, network connectivity modules implanted on the vehicles, animals, household electrical

appliances, medical equipment, and many more. These devices perform tasks in a diverse range of applications, such as E-health [5], personal healthcare devices [6], radio sensor networks [7], the intelligent transport system (ITS) [8], smart cities [9], and industrial IoT [10]. The massive nature of IoT poses some major challenges, including spectrum scarcity, reliability, energy efficiency, high-speed uplinks, and ultra-low latency. In particular, the IoT devices are highly power constrained and lack the ability to communicate at longer distances. This becomes a critical issue when these devices are deployed in areas with poor or no coverage provided by existing terrestrial wireless networks.

In such an IoT scenario, the use of DBSs can be a promising solution. The DBSs can serve the role of moving data aggregators—in other words, they can fly close to IoT devices, collect their data, and transmit it to other devices which are out-of-range of the transmitting IoT devices. In addition, due to their aerial nature, DBSs can be deployed at high altitudes. This helps to mitigate the shadowing effects and increases the probability of LOS communication between the DBSs and ground IoT devices. Consequently, the battery-limited IoT devices will be able to communicate with much lower transmit power. In addition, DBSs also have the ability to change their location based on the activation pattern of IoT devices. This helps them to support an anticipated number of devices in massive IoT deployment.

1.1. Literature Review

Despite the elegant features, DBSs themselves are power-constrained machines. Using drones for IoT communications can be an expensive solution if they are not optimally positioned. The total cost of DBS deployment constitutes the flight time, the connection establishment time, disconnection time, and the cost of drones as a device. Therefore, it is of paramount importance to come up with an optimal drone placement to serve the maximum number of IoT devices. In the literature, the drone placement problem for different objectives is considered in a number of published works. In [11], the authors consider a 2D placement problem with an objective to maximize the network coverage while minimizing the transmit power. A heuristic is proposed to solve the optimization problem. The formulated 2D placement problem is further modified for efficient drones deployment in a 3D scenario. In [12], a backhaul-aware 3D drone placement is considered for a heterogeneous network. The placement algorithm maximizes the sum-rate and number of served users, considering the limiting factors, the peak data rate of wireless backhaul, and the bandwidth of DBS. In most of the research, the free-space path loss model has been considered for DBS deployment; however, in real scenarios, the IoT devices are scattered everywhere, that is, at indoor or outdoor locations. In [13], the wireless coverage for indoor users inside the high-rise buildings is considered. The presented model assumes a single drone. A hybrid path-loss model is proposed, which takes into account the path losses due to free-space, building penetration, and indoor losses. Particle swarm optimization (PSO) is generally used to minimize the total transmit power required to cover the indoor users. In [14], a solution was proposed to maximize the revenue of the cellular network using 3D placement of DBSs. The problem was formulated as a mixed-integer non-linear problem (MINLP) and solved using a heuristic. In [15], 3D deployment of DBS was considered to study energy-efficient communication in mobile ad hoc networks. The work proposes an optimal DBS placement that minimizes the UAV-recall-frequency (UAV-RF), while taking into account the on-board circuit power. The authors also studied the effect of scattering on the optimal hovering altitude of DBS. Furthermore, the work concludes that minimum UAV-RF is achieved when the transmit power is equal to the power consumed by the on-board circuit. Network life can be prolonged by limiting this on-board circuit power. In [16], a mathematical model is proposed for the optimal number of drones, along with their efficient placement in wireless cellular networks. The problem is solved using a PSO algorithm. In [17], a heuristic solution was proposed that minimizes the number of DBSs to provide wireless coverage to a group of distributed TBSs. In [18], the placement of DBSs was studied to maximize the coverage for users with different QoS requirements. In [19], a method was proposed for joint radio resource allocation, 3D placement, and user association of DBSs for IoT networks. As discussed earlier, drones can be used for the assistance of already deployed networks. In [20], the resource allocation problem for drone-assisted networks was investigated,

where the drones were utilized to harvest energy to the D2D pairs. Similarly, in [1], the drones were adopted to assist the TBSs for densification. In [21], the authors proposed a methodology for UAV trajectory design and radio resource assignment for a vehicle-to-everything scenario. The work proposes a joint scheduling strategy where a vehicle can be served by both the terrestrial BS (TBS) or DBS, and the one that provides maximum throughput is selected. The TBSs are selected on the basis of round robin scheduling, whereas the DBS are selected using both round robin and proportionality fair scheduling. A heuristic solution is proposed for UAV trajectory planning. DBSs can be used in a distributed and multi-layered architecture, such as the micro-cell/picocell, macrocell/microcell, or just macrocell [22]. The DBSs can also function in cooperation, using swarm formation. But as the DBSs have limited computational, storage, and communication capacities, they are unable to perform computationally intensive tasks. In order to extend the capacity of DBSs swarms, an DBSs-edge-cloud computational model [23] was adopted in order to support resource-intensive application, such as big data, artificial intelligence, and computer vision. An on-demand UAV placement was proposed in [24], where the users were arbitrarily distributed. The problem was modelled as a Knapsack-like problem, where a density-aware placement algorithm was proposed to maximize the user coverage subject to the constraint of the minimum required data rates per user.

In [25], a joint optimization problem was solved for the placement and power allocation (PA) to improve the performance of the NOMA-UAV network. The objective was to maximize the sum rate of users. Along with IoT, the body sensor network (BSN) is also among one of the major applications of 5G networks. Both in IoT and BSN, the nodes are very quickly drained due to a number of performed operations, including registration, removal, and routing. Therefore, an energy-efficient solution is required to maintain a high transmission capacity. In [26], an efficient approach for device discovery in 5G-based IoT and BSNs using multiple UAVs was proposed, where significant achieved gains in energy consumption were claimed. An integrative IoT platform is presented in [27], where the drones are utilized to serve different IoT devices. [2] studies the role of drones as relays, both for amplify-and-forward (AF) and decode-and-forward (DF) protocols in the cooperative communication.

To the best of our knowledge, in most existing studies, one aspect of resource allocation is addressed, in that there are works which focus on DBS assignment alone, or works in which user association is optimized given a fixed deployment of DBSs. In contrast, this work proposes a mathematical model that jointly optimizes the DBS placement and resource allocation problem.

1.2. Contribution

The main contributions to this work are as follows.

- Firstly, a mathematical model is presented that jointly optimizes the resources used by IoT devices. The objective of the optimization problem is to minimize the ratio of the number of DBSs to the connected IoT devices. The proposed model is developed by adopting a realistic air-to-ground (A2G) path loss model;
- The formulation of the proposed model considers the practical network constraints, which include: (a) the DBS deployment budget, (b) the network's QoS requirements, and (c) the DBS battery-life prolongevity;
- A low-complexity heuristic is proposed to solve the optimization problem;
- The simulation results are carried out for a number of network scenarios and validate the accuracy of the proposed solution. The results' comparisons show that the proposed algorithm's performance is close to that of the optimal solution.

2. System Model and Problem Formulation

Figure 1 illustrates an example of a massive IoT network spread over a metropolitan city. In practical situations, hundreds of heterogeneous IoT devices are deployed over large geographical areas. These devices perform tasks in a wide variety of applications, including smart homes,

remote asset control in industries, connected vehicles, wearable gadgets, smart shops, smart airports, tele-medicine, environmental monitoring, and disaster management. Such massive IoT deployment results in enormous growth of internet traffic, and requires existing cellular networks to undergo major architectural changes.

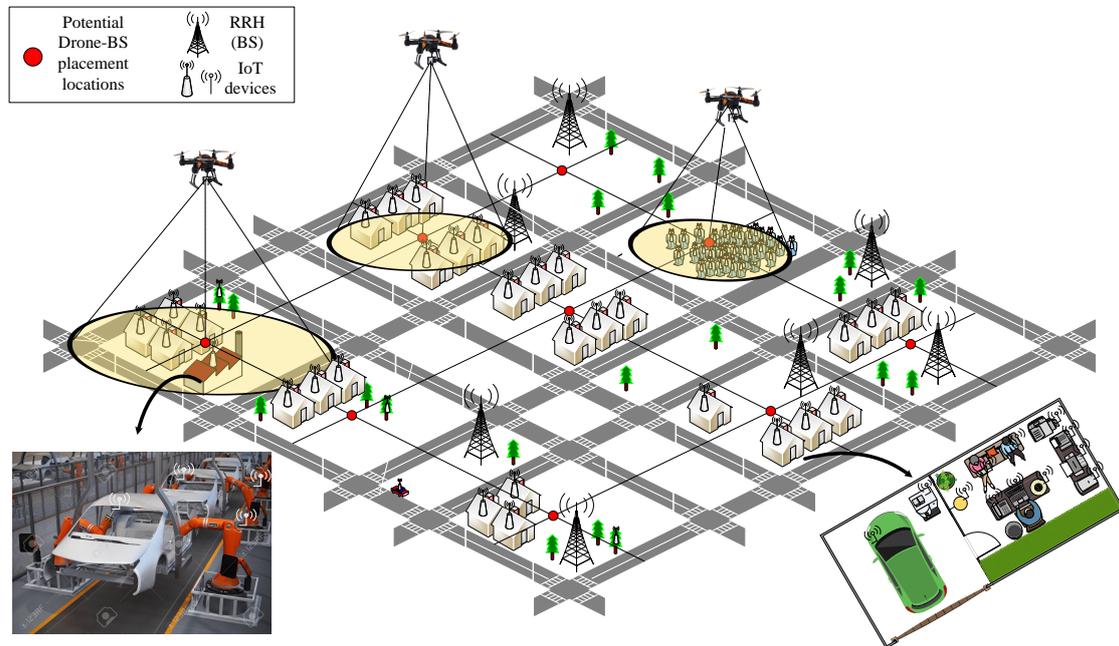


Figure 1. Proposed system model of an Internet of Things (IoT) network spread. It consists of N heterogeneous IoT (mobile and stationary) users spread over a metropolitan area with M potential locations for DBS placement.

In a general IoT network, the IoT user equipments (UEs) are able to communicate with each other through wireless resources provided by TBSs, equipped with remote radio heads (RRHs). The RRHs are then connected to each other through high-speed communication links. Due to their mobility, it may be that a massive number of IoT devices are active in a given geographical area for a limited time. In such a situation, the DBSs come handy and provide a cost-effective solution to complement the TBSs and improve the connectivity and energy efficiency of IoT networks. DBSs can also help to restore communications in areas where terrestrial infrastructure has been collapsed after a disastrous situation.

Consider an IoT network which serves a total of N IoT devices (In the following text, the word IoT devices is used interchangeably with IoT users or simply users) in a given geographical area. The area is divided into M potential candidate locations where DBS can be deployed. Due to their mobility, DBSs are able to move from one location to another in order to fulfil the connectivity requirements. Let's say the triplet $(\bar{x}_i, \bar{y}_i, h_i)$ denotes the location of DBS i in a 3D system, where \bar{x}_i, \bar{y}_i are its position coordinates in a X - Y coordinate system, and h_i is its height. This work assumes a constant height h for all DBSs deployed in the network. A DBS can potentially occupy any discrete location between $(0, 0, h)$ and $(\bar{x}_{max}, \bar{y}_{max}, h)$.

The DBS placement algorithm is aimed at determining the optimal number and position of DBSs which maximizes the connectivity for a given set of IoT users. An air-to-ground (A2G) path loss model is assumed in this work, and discussed as follows.

2.1. Air-to-Ground (A2G) Path Loss Model

As shown in Figure 1, the exact locations of the IoT devices and the approximate locations of the obstacles, such as high-rise buildings, are known, along with the environment (e.g., urban, rural, suburban). However, due to a lack of complete information, the probabilistic approach is adopted.

The probability of line-of-sight (LoS) and non-line-of-sight (NLoS) between DBSs placed at location m and the IoT device at n are given as [28]:

$$P_{LoS}^{mn} = \frac{1}{1 + a \exp(-b[\theta_{mn} - a])}, \forall m, n \quad (1)$$

$$P_{NLoS}^{mn} = 1 - P_{LoS}^{mn}, \forall m, n \quad (2)$$

where a and b are the constants, and their values depend on the carrier frequency f_c and type of environment, such as rural, urban, or dense urban [28]. θ_{mn} is the elevation angle between the DBS m and the IoT device n , and is equal to $\tan^{-1} \frac{h}{r_{mn}}$, where h is the altitude of DBS and r_{mn} is the horizontal distance between DBS m , and the IoT device n is given as:

$$r_{mn} = \sqrt{(\bar{x}_n - \bar{x}_m)^2 + (\bar{y}_n - \bar{y}_m)^2 + h^2}, \forall m, n \quad (3)$$

The expressions for path loss for LOS and NLoS communication are given as $L_{LoS, mn} = Y + \sigma_{LoS}$ and $L_{NLoS, mn} = Y + \sigma_{NLoS}$, respectively. Here, $Y = 10\alpha \left(\frac{4\pi f_c r_{mn}}{c} \right)$ [28], where c is the speed of light, α is the path-loss exponent, and σ_{LoS} and σ_{NLoS} are average additional losses which occur in addition to the free-space propagation loss. The values of σ_{LoS} and σ_{NLoS} depend on the environment. Finally, the average path loss can be represented as a function of DBS height and coverage circle, as:

$$L(x_m, y_m, h_m) = P_{LoS}^{mn} L_{LoS}^{mn} + P_{NLoS}^{mn} L_{NLoS}^{mn}, \forall m, n \quad (4)$$

and the average channel gain can be computed as:

$$g_{mn} = L(x_m, y_m, h_m)^{-1}, \forall m, n \quad (5)$$

Using the channel gain computed above, the signal-to-noise ratio (SINR) for a downlink channel between a DBS m and IoT user n can be computed as:

$$SINR_{mn} = \frac{P_m g_{mn}}{\sum_{u \in \mathcal{I}_{int}} P_u g_{un} + \sigma^2} \quad (6)$$

where P_m is the transmit power of DBS m , \mathcal{I}_{int} is the set of interfering DBSs, and σ is the background noise.

2.2. Variable Definition

The main notations and variables used in this work are listed in Table 1. This work considers an IoT network composed of N IoT devices and M candidate locations for DBS placement. Let's say \mathbf{x} is a DBS placement vector of size M , given as:

$$\mathbf{x} = (x_1, x_2, \dots, x_M) \quad (7)$$

where an entry x_i in \mathbf{x} is a binary "1" when a DBS is placed at location i , and "0" otherwise. Similarly, let's say \mathbf{Y} is the user-DBS connectivity matrix of size $N \times M$. An entry y_{nm} of \mathbf{Y} is a binary "1" when a user n is connected to a DBS placed at potential location m , and "0" otherwise.

2.3. System Constraints

The DBS deployment problem modelled in this work considers the following practical constraints.

2.3.1. Minimum Users Connectivity Constraint

To achieve the QoS requirements of the IoT network, a certain minimum percentage β of total IoT devices must be served. Mathematically,

$$\sum_{n=1}^N \sum_{m=1}^M y_{mn} \geq \beta N \quad (8)$$

where $\beta \in [0, 1]$.

2.3.2. User Connectivity with Deployed DBS Constraint

According to this constraint, an IoT user must be connected to a potential location where DBS has actually been deployed. Mathematically,

$$y_{mn} \leq x_m, \forall n \in \{1, \dots, N\}, m \in \{1, \dots, M\}, \quad (9)$$

Equation (9) means that if a DBS is placed at a certain location, m (i.e., $x_m = '1'$ in \mathbf{x}), then a user may or may not be connected to it (i.e., entries in the m -th row of \mathbf{Y} can be "0" or "1"). However, if a DBS is not placed at location m (i.e., $x_m = '0'$), then no user can be connected to this location (i.e., all entries in row m of \mathbf{Y} must be equal to "0").

2.3.3. Single DBS Connectivity for a User Constraint

A user/device can be connected to one DBS at most. Mathematically, this constraint can be formulated as:

$$\sum_{m=1}^M y_{mn} \leq 1, \forall n \quad (10)$$

since rows of \mathbf{Y} correspond to the DBSs and its columns correspond to users, $\sum_{m=1}^M y_{mn}$ indicates the number of DBSs to which the user n is simultaneously connected. According to (10), this sum must be less than or equal to "1" for each IoT user in the network.

2.3.4. Maximum and Minimum Capacity of DBS Constraint

Each DBS in the network is a resource-limited device, and is capable of serving a certain maximum number γ^{max} of users. Mathematically:

$$\sum_{n=1}^N y_{mn} \leq \gamma^{max} x_m, \forall m \in \{1, \dots, M\} \quad (11)$$

According to Equation (11), if a DBS is assigned at location m (i.e., $x_m = '1'$), then the total number of users connected to it (i.e., $\sum_{n=1}^N y_{mn}$) must be upper-bounded by γ^{max} . Similarly, a DBS must be connected to a certain minimum number γ^{min} of users to justify its deployment cost. Mathematically:

$$\sum_{n=1}^N y_{mn} \geq \gamma^{min} x_m, \forall m \in \{1, \dots, M\} \quad (12)$$

2.3.5. DBS Deployment Budget Constraint

The DBS deployment incurs a cost, which includes the costs of flight time and connection establishment. The total deployment cost of all DBSs assigned must not exceed a certain available budget. Mathematically, this constraint can be modeled as:

$$\sum_{m=1}^M c_m x_m \leq C^{max} \quad (13)$$

where C^{max} is the maximum available budget for DBS deployment, and c_m is the deployment cost of a single DBS placed at location m .

2.3.6. Maximum User-DBS Distance Constraint

A user can only be connected to a DBS if its distance to the DBS location lies within a certain maximum value d^{max} . Mathematically,

$$y_{mn} \cdot d_{mn} \leq d^{max} \quad (14)$$

where d_{mn} denotes the distance between an IoT device n and the DBS placed at location m .

2.4. An Optimization Model for Drone-BS Deployment for Maximal Coverage

Given the network scenario and practical constraints mentioned above, this work considers the following mathematical model for DBS placement.

$$\min_{\substack{x_m \in \{0,1\}, \\ y_{mn} \in \{0,1\}, \\ \forall n,m}} \frac{\overbrace{\sum_{m=1}^M x_m}^{\text{No. of DBSs}}}{\underbrace{\sum_{n=1}^N \sum_{m=1}^M y_{mn}}_{\text{No. of connected IoT devices}}} \quad (15)$$

subject to:

Constraints (8)–(14)

The optimization problem of (15) is an integer fractional optimization model. The numerator of the objective function denotes the total number of DBSs deployed, whereas the denominator denotes the total number of IoT users served by the network. Therefore, the optimization goal is to obtain maximum network connectivity with the minimum number of DBSs, while satisfying the practical network constraints.

2.5. Linearization of Optimization Problem

This work linearizes the fractional optimization problem of (15) using the method proposed in [29]. Let us consider two variables $t = \{\sum_{n=1}^N \sum_{m=1}^M y_{mn}\}^{-1}$ and $\alpha = \max[1, \min(\sum_{n=1}^N \sum_{m=1}^M y_{mn})^{-1}]$, where $y_{mn} \in \{0, 1\}$ and $\sum_{n=1}^N \sum_{m=1}^M y_{mn} > 0$. Therefore, $\alpha = \max[1, 1] = 1$. Now the optimization problem (15) can be rewritten as:

$$\min_{\substack{x_m \in \{0,1\}, \\ y_{mn} \in \{0,1\}, \\ 0 \leq t \leq 1 \\ \forall n,m}} \alpha \sum_{m=1}^M x_m t \quad (16)$$

subject to:

Constraints (8)–(14)

$$\alpha \sum_{n=1}^N \sum_{m=1}^M y_{mn} t = 1$$

The objective function in (16) can be expanded as,

$$\min \alpha [x_1 t + x_2 t + \dots + x_M t] \quad (17)$$

and constraint $\alpha \sum_{n=1}^N \sum_{m=1}^M y_{mn}t = 1$ in (16) can be expanded as,

$$\min \alpha [y_{11}t + y_{12}t + \dots + y_{1N}t + y_{21}t + \dots + y_{2N}t + \dots + y_{M1}t + \dots + y_{MN}t] = 1 \quad (18)$$

Let's say that $z_1 = x_1t, z_2 = x_2t, \dots, z_M = x_Mt$ and $w_{11} = y_{11}t, w_{12} = y_{12}t, \dots, w_{MN} = y_{MN}t$. As shown in [29], the multiplication of a binary variable a with a continuous variable b , that is, $c = ab$, can be represented as linear inequalities, such as:

$$c = ab \Rightarrow \begin{cases} \text{(i)} & b - c \leq 1 - a \\ \text{(ii)} & c \leq b \\ \text{(iii)} & c \leq a \\ \text{(iv)} & c \geq 0 \end{cases}, \quad (19)$$

In (17) and (18), there are M and $M + N$ multiplications of the continuous variable t with binary variables x_m and y_{mn} , respectively. By using (19), these multiplications can be converted into linear inequalities. Therefore, the optimization problem (16) can be restated as:

$$\min_{\substack{x_m, y_{mn} \in \{0,1\}, \\ z_m, w_{mn} \geq 0, \\ 0 \leq t \leq 1 \\ \forall n, m}} \alpha \sum_{m=1}^M z_m$$

subject to:

Constraints (8)–(14)

$$\alpha \sum_{n=1}^N \sum_{m=1}^M w_{mn} = 1 \quad (20)$$

$$t - z_m \leq 1 - x_m, \quad \forall m$$

$$z_m \leq t, \quad \forall m$$

$$z_m \leq x_m, \quad \forall m$$

$$t - w_{mn} \leq 1 - y_{mn}, \quad \forall m, n$$

$$w_{mn} \leq t, \quad \forall m, n$$

$$w_{mn} \leq y_{mn}, \quad \forall m, n$$

3. Proposed Solution

The linear programming optimization problem of (20) is solvable optimally by using a Branch and Bound (B&B) algorithm. However, the worst case complexity is exponential to the product of the sizes of the binary variables \mathbf{x} and \mathbf{Y} . This work proposes a low-complexity greedy heuristic for joint DBS placement and the user association problem. Algorithm 1 shows the main computation steps of the proposed solution, whose main symbols and notations are listed in Table 1. The algorithm receives as inputs, the values of $N, M, \gamma^{max}, \gamma^{min}$, and d^{max} . For simulation purposes, the network area is represented as a grid with discrete potential DBS locations, and where the users are evenly distributed. A matrix \mathbf{d} of dimensions $N \times M$ is obtained, whose entry d_{nm} denotes the distance of an IoT user n from DBS location m . During the initialization phase, the important variables are initialized, which include the DBS placement vector \mathbf{x} , user-DBS association matrix \mathbf{Y} , a vector \mathbf{c} of size M which denotes the number of users served by each DBS, and a vector \mathbf{t} of size N which denotes the number of DBSs connected to each user. A binary matrix \mathbf{W} of dimensions $N \times M$ is also computed, which shows all possible connections between users and DBSs. An entry $w_{n,m}$ of \mathbf{W} is equal to "1" only if the distance

between the user n and location m is less than the maximum distance D^{max} , and is “0” otherwise. The execution of the proposed algorithm is divided into five phases, which are discussed below.

Algorithm 1 Heuristic for efficient drone placement

```

1: Inputs:  $N, M, \gamma^{max}, \gamma^{min}, \beta, \mathbf{R}, d^{max}$ 
2: Initialization:  $\forall n \in \{1, \dots, N\}, m \in \{1, \dots, M\}$ 
3:  $\mathbf{Y}_{nm} \leftarrow 0, \mathbf{x}_m \leftarrow 0, \mathbf{c}_m \leftarrow 0, \mathbf{t}_n \leftarrow 0$ 
4:  $\mathbf{W}_{nm} \leftarrow \begin{cases} 1, & \text{if } R_{nm} < d^{max} \\ 0, & \text{otherwise} \end{cases}$ 
5: Step I: Allocation to users with single connection option
6:  $\mathbf{t}_n \leftarrow \sum_{m=1}^M \mathbf{W}_{n,m'}, \forall m' \in \{1, \dots, M\}$ 
7: for  $i \in \{1, \dots, N\}$  do
8:   if  $\mathbf{t}_i == 1$  then
9:      $\mathbf{k} \leftarrow \text{find}(\mathbf{W}_{i,:} == 1)$ 
10:     $\mathbf{Y}_{i,k} \leftarrow 1, \mathbf{x}_k \leftarrow 1, \mathbf{c}_k \leftarrow \mathbf{c}_k + 1$ 
11:   end if
12: end for
13: Step II: Allocation of users with connections options to multiple drones
14: for  $i \in \{1, \dots, N\}$  do
15:   if  $\mathbf{t}_i > 1$  then
16:      $flag \leftarrow 0$ 
17:      $\mathbf{k} \leftarrow \text{find}(\mathbf{W}_{i,:} == 1), \mathbf{j} \leftarrow \text{find}(\mathbf{x} == 1)$ 
18:      $[\mathbf{I}, \mathbf{v}] \leftarrow \text{sort}(\mathbf{c})$ 
19:     for  $j = 1 : \text{len}(\mathbf{I})$  do
20:        $\mathbf{a}_j \leftarrow \begin{cases} 1, & \text{if } \mathbf{I}_j \in \mathbf{k} \\ 0, & \text{otherwise} \end{cases}$ 
21:     end for
22:     if  $\sum_{j=1}^{\text{len}(\mathbf{a})} \mathbf{a}_j > 0$  then
23:        $\mathbf{I}' \leftarrow \text{find}(\mathbf{a} == 1)$ 
24:       for  $q \leftarrow 1 : \text{len}(\mathbf{I}')$  do
25:          $\mathbf{s} \leftarrow \mathbf{I}'_q$ 
26:         if  $\mathbf{c}_{\mathbf{I}_s} < \gamma^{max}$  then
27:            $\mathbf{Y}_{i,\mathbf{I}_s} \leftarrow 1, \mathbf{x}_{\mathbf{I}_s} \leftarrow 1, \mathbf{c}_{\mathbf{I}_s} \leftarrow \mathbf{c}_{\mathbf{I}_s} + 1$ 
28:            $flag \leftarrow 1$ 
29:           break
30:         end if
31:       end for
32:     if  $flag == 0$  then
33:        $\mathbf{k} \leftarrow \mathbf{k} \setminus \mathbf{I}$ 
34:       if  $\mathbf{k} == \emptyset$  then
35:          $\mathbf{W}_{(i,:)} \leftarrow 0$ 
36:          $flag \leftarrow 1$ 
37:       end if
38:     end if
39:   end if
40:   if  $flag == 0$  then
41:      $\mathbf{Y}_{i,\mathbf{k}_1} \leftarrow 1, \mathbf{x}_{\mathbf{k}_1} \leftarrow 1, \mathbf{c}_{\mathbf{k}_1} \leftarrow \mathbf{c}_{\mathbf{k}_1} + 1$ 
42:   end if
43: end if
44: end for
45: Step III: Check for under utilized drones
46:  $\mathbf{v} \leftarrow \text{find}((\mathbf{c} < \gamma^{min}) \cdot \mathbf{x} == 1)$ 
47:  $\mathbf{Y}_{:,v} \leftarrow 0, \mathbf{x}_v \leftarrow 0, \mathbf{c}_v \leftarrow 0$ 
48:  $\mathbf{p} \leftarrow \sum_{i=1}^N \mathbf{Y}_{i,:}$ 
49:  $\mathbf{u}' \leftarrow \text{find}(\mathbf{p} == 0)$ 
50:  $b \leftarrow \text{round}(\beta \cdot N)$ 
51:  $t \leftarrow \sum_{i=1}^N \sum_{j=1}^M \mathbf{Y}_{i,j}$ 
52: for  $i \leftarrow 1 : \text{len}(\mathbf{u}')$  do
53:   if  $t < b$  then
54:      $flag \leftarrow 0$ 
55:      $[\mathbf{I}, \mathbf{v}] \leftarrow \text{sort}(\mathbf{c})$ 
56:     for  $j \leftarrow 1 : M$  do
57:       if  $flag == 0$  then
58:         if  $\mathbf{c}_j < \gamma^{max}$  and  $\mathbf{W}_{\mathbf{u}',j} == 1$  then
59:            $\mathbf{Y}_{\mathbf{u}',j} \leftarrow 1, \mathbf{x}_j \leftarrow 1, \mathbf{c}_j \leftarrow \mathbf{c}_j + 1$ 
60:            $flag \leftarrow 1$ 
61:            $t \leftarrow \sum_{ii=1}^N \sum_{jj=1}^M \mathbf{Y}_{ii,jj}$ 
62:         end if
63:       end if
64:     end for
65:   end if
66: end for
67: Step IV: Connect all unconnected users
68:  $t \leftarrow \sum_{i=1}^N \sum_{j=1}^M \mathbf{Y}_{i,j}$ 
69:  $\mathbf{p} \leftarrow \sum_{i=1}^N \mathbf{Y}_{i,:}$ 
70:  $\mathbf{u}' \leftarrow \text{find}(\mathbf{p} == 0)$ 
71: for  $i \leftarrow 1 : \text{len}(\mathbf{u}')$  do
72:   if  $t < b$  then
73:      $\mathbf{r} \leftarrow \mathbf{R}_{(i,:)}$ 
74:      $flag \leftarrow 0$ 
75:      $\mathbf{k} \leftarrow \mathbf{W}_{\mathbf{u}',i}$ 
76:     for  $j \leftarrow 1 : \text{len}(\mathbf{k})$  do
77:       if  $flag == 0$  then
78:          $[\mathbf{l}^v, \mathbf{l}^i] \leftarrow \min(\mathbf{r})$ 
79:          $\mathbf{r}_j \leftarrow \inf$ 
80:          $[\mathbf{l}^v, \mathbf{l}^i] \leftarrow \min(\mathbf{r})$ 
81:         if  $\mathbf{W}_{\mathbf{u}',\mathbf{l}^i} > 0$  then
82:            $\mathbf{Y}_{\mathbf{u}',\mathbf{l}^i} \leftarrow 1, \mathbf{x}_{\mathbf{l}^i} \leftarrow 1, \mathbf{c}_{\mathbf{l}^i} \leftarrow \mathbf{c}_{\mathbf{l}^i} + 1$ 
83:         end if
84:       end if
85:     end for
86:   end if
87: end for
88: Step V: Remove all under-utilized drones
89:  $\mathbf{d}^u \leftarrow \text{find}((\mathbf{c} < \gamma^{min}) \cdot \mathbf{x} == 1)$ 
90:  $\mathbf{Y}_{:,d^u} \leftarrow 0, \mathbf{x}_{d^u} \leftarrow 0, \mathbf{c}_{d^u} \leftarrow 0$ 
91:  $t \leftarrow \sum_{i=1}^N \sum_{j=1}^M \mathbf{Y}_{i,j}$ 
92: Outputs:  $\mathbf{Y}, \mathbf{x}$ 

```

Table 1. List of notations and variables of this work.

Notation	Explanation
N	Total no. of IoT users
M	Total no. of potential locations for DBS placement
\mathbf{x}	Binary indicator vector of size M for DBS placement
\mathbf{Y}	Binary matrix of size $N \times M$, which indicates achieved associations between users and DBSs
β	Constant which indicates the minimum percentage of users that must be serviced by the network
$\gamma^{\min}(\gamma^{\max})$	Minimum (Maximum) number of users that each DBS must (could) serve
c_m	Cost of DBS deployment at location m
C^{\max}	Maximum deployment cost of all DBSs
d_{nm}	Distance between user n and DBS location m
d^{\max}	Maximum coverage distance of a single DBS
\mathbf{W}	A matrix of dimensions $N \times M$ which indicates potential DBS associations for each user
\mathbf{R}	A matrix of dimensions $N \times M$ which contains the distance between IoT users and DBS locations
\mathbf{t}	A column vector of size N indicating the possible number of DBS connections for each user
\mathbf{c}	A row vector of size M which indicates the total number of connected users for each DBS
$find(x)$	Function which returns the indexes where its argument x is true
$sort(c)$	Sorts the vector c in descending order and returns values and indexes
∇	Vector of size M which stores the indexes of under-utilized DBSs
\mathbf{u}'	Vector which stores the indexes of unconnected users
$len(\mathbf{x})$	Returns the size of vector \mathbf{x}
$min(\mathbf{r})$	Returns the minimum value and its indexes from the vector \mathbf{r}
:	Operator indicating complete row or column in a matrix

3.1. Phase I: Allocation of Users within the Coverage Area of a Single DBS

In the first phase, those users are assigned which are located within the coverage area of a single DBS location. Step 8 obtains the total number of DBS connections possible for each user $n \in \{1, \dots, N\}$. This is done by adding all entries at row n of \mathbf{W} and storing the result at the corresponding row in vector \mathbf{t} . In the next step, for the loop runs for N iterations, if the possible number of DBS connections of user i (i.e., t_i) is equal to "1" during each iteration, then the corresponding index of DBS is obtained from \mathbf{W} (i.e., column index) and stored in k . The k -th entry in \mathbf{x} is set to "1" to mark a DBS placement at potential location k . Similarly, the entry $y_{i,k}$ is set to "1" in \mathbf{Y} to mark an association between user i and DBS location k . Finally, the count value of connected users for DBS k is updated in \mathbf{c} .

3.2. Phase II: Allocation of Users within the Coverage Area of Multiple DBS Locations

During this phase, those users are assigned which are located in the coverage area of multiple DBS locations. In Step 17, for loop runs for N users (iterations), for each user i , the number of its potential DBS connections (i.e., t_i) is obtained. If this value is greater than "1", then Step 20 obtains the indexes of all potential DBSs for user i and stores them in vector \mathbf{k} . In addition, the indexes of those DBSs already deployed are obtained from \mathbf{x} and stored in vector \mathbf{j} . The function $find(\cdot)$ returns the indexes where the argument is true. In Step 21, the non-zero entries of \mathbf{c} are sorted in descending order and stored in vector λ along with their indexes in vector \mathbf{I} . In Steps 22–24, a binary vector \mathbf{a} is obtained which has non-zero entries for the indexes of all DBSs which have already been deployed, and can also be assigned to user i . The indexes of these DBSs are obtained in Step 26. Among these potential DBSs for user i , the DBS connected to maximum number of users is selected, provided it satisfies the maximum capacity γ^{\max} criteria. Such user-to-DBS assignment is done in Steps 26–34, along with the update of variables \mathbf{x} , \mathbf{Y} , and \mathbf{c} . A temporary *flag* variable is set whenever a user is assigned. If all the existing DBSs are fully up to their maximum capacity, then the flag is still 0 at the end of Step 34. Steps 35–42 exclude the indexes of deployed DBSs from the potential connections for user i . If the result is a null vector ϕ , it means that the user i cannot be connected to any DBS other than existing DBSs which are already full. Such a user is not assigned by the network as shown in Step 38, which places a "0" at all DBS indexes in \mathbf{W} for user i . In the third case, a user may have potential connection to one or multiple locations where no DBS has already been deployed. In such a case, a new DBS is

placed at the location with the smallest available index, and user i is associated to it. This is done in Steps 43–47 by updating the corresponding entries of \mathbf{x} , \mathbf{Y} , and \mathbf{c} .

3.3. Phase III: Check for Under-Utilized Drones

In Step 50, the indexes are stored in ∇ for all DBSs that remain under-utilized after the association steps of phase I and II. These DBSs are removed in Step 51 by disassociating their connected users and setting the relevant entries of \mathbf{Y} , \mathbf{x} , and \mathbf{c} to “0”. In Steps 52–53, the indexes of unconnected users are obtained and stored in vector \mathbf{u}' . Step 54 computes b , that is, the minimum number of users that must be serviced, whereas Step 55 computes t , that is, the number of users that have been serviced so far. If the number of served users is less than the minimum users to be served, it is executed until both these numbers become equal. During each iteration of the loop, that is, in Steps 56–70, one of the unconnected users is selected from the list \mathbf{u}' and assigned to an existing DBS with the highest utilization, with the condition that the DBS is not overcrowded and belongs to the set of DBSs that have potential connections for the user. The user-DBS association is marked by updating the relevant entries of \mathbf{Y} , \mathbf{x} , and \mathbf{c} .

3.4. Step IV: Association of All Unconnected Users until Now

In this phase, the users which remain unconnected in the previous phases are associated to the DBSs. Their indexes are obtained in \mathbf{u}' in Step 75. The for loop of Steps 76–92 performs user-DBS association on the basis of distance. For each user $i \in \mathbf{u}'$, its distances from all the DBS locations are obtained from matrix \mathbf{R} and stored in \mathbf{r} in Step 78. In Step 79, the list of potential DBS connections for the users is obtained and stored in \mathbf{k} . In Steps 82–86, among all potential DBSs for the user, the DBS with the minimum distance is selected. The user association to this DBS is marked in Step 87 by updating the relevant matrices.

3.5. Step V: Remove All Under-Utilized Drones

Finally, the under-utilized drones are removed by disassociating the users connected to them, and the total number of connected users is computed. The outputs of the algorithm are the user-DBS association matrix \mathbf{Y} and the DBS placement vector \mathbf{x} .

3.6. Complexity

The main advantage of the proposed method is its low computational complexity, where the complexity is calculated in terms of flops. The complexity of the initialization phase (Steps 2–4) is $5MN$. The complexity of Steps 6–12 is $2MN + 4N$. The worst case complexity of Steps 14–44 is $M^2N + 11MN + 13N$, and Steps 46–66 have a worst case complexity equal to $2M^2 + M^2N + 9M$, where Steps 68–87 have a complexity $6MN + 9N$ and Steps 89–91 have a complexity equal to $5M + MN$ flops. The worst case complexity of the proposed heuristic overall is $O(M^2N)$, whereas the complexity of the optimal algorithm is 2^{NM^2} . This clearly demonstrates that the proposed solution is fairly applicable for practical, large-area IoT networks.

4. Results and Discussion

This section presents the simulation results of proposed solution. As discussed earlier, the network area is visualized as a grid with discrete points which represent the potential DBS locations, and the users are evenly distributed on the grid. The simulations are carried out for a variety of realistic network scenarios with grid sizes of 3×3 , 4×4 , and 5×5 DBS locations, with the total number of users N ranging from 50 up to 200 and $\beta = 0.2$ and 0.8 . The optimization goal is to minimize the utility function of (15), which is represented as a ratio between the number of deployed DBSs and the number of connected users. The performance of the proposed heuristic is compared with the optimal algorithm. As discussed earlier, β is the minimum fraction of users that must be serviced. This means

that for a larger β , more drones need to apparently be deployed. Similarly, if the grid size is smaller, it would be difficult to service a larger β percentage of users. These facts have been proven through a number of simulations, and the results are discussed below.

Before going into the simulation analysis, let's try to analyze the objective function. The objective function is the ratio of the total number of deployed drones to the total number of connected users. The objective is to minimize this ratio. From the basic mathematical theory, it can be easily observed that this ratio (objective function) could be minimized if:

- The numerator is decreased, keeping the denominator fixed;
- The denominator is increased, keeping the numerator fixed;
- The numerator is decreased, and at the same time, the denominator is increased.

In this objective function, the numerator represents the total number of deployed drones, whereas the denominator represents the total number of connected users. Therefore, from the analysis above, either a smaller number of drones have to be deployed, keeping a fixed number of connected users, or more users need to be connected with a smaller number of drones. The third option is to decrease the number of drones, and increase the number of connected users at the same time.

In Figure 2, the performance results are reported for different grid sizes and $\beta = 0.2$. Figure 2a plots the value of objective function (15) for different grid sizes and the number of users. A number of trends can be observed from the plot. The value of objective function achieved by optimal and heuristic algorithms increases with grid size, keeping other parameters constant. This is due to the fact that greater grid size means a greater number of potential locations. This increases the probability of greater DBS deployment. Moreover, for a given grid size, the value of the utility function and the performance gap between the two algorithms decrease with an increase in the number of users. This trend is due to the fact that as more users are placed within the network of a given grid size, the degree of freedom of their association with DBSs increases, resulting in better overall network utilization. For all grid sizes and the number of users, the utility function value obtained by the proposed heuristic is fairly close to the optimal solution.

In Figure 2b, the performance results are presented in terms of the number of connected users. For all grid sizes and values of N , both algorithms satisfy the minimum QoS requirements represented as black bars in the plots. The plots further reveal that, for a given grid size and N , the proposed heuristic results in greater number of connected users as compared to the optimal one. Moreover, the user connectivity achieved by the proposed heuristic decreases with an increase in grid size, whereas this quantity remains almost the same for an optimal algorithm. This trend is due to the fact that as grid size increases for a given value of N , the users initially get connected to their nearest DBS location, resulting in some drones being under-utilized. These DBSs are removed in the later phases of the algorithm to satisfy constraint C4. As a result, some of the users are connected to the existing DBSs, while others remain unassigned. In Figure 2c, the performance results are plotted in terms of the number of DBSs assigned. The proposed heuristic places more DBSs, as compared to the optimal algorithm. However, it also serves a greater number of users. Overall, the combined result, that is, the ratio of quantities of Figure 2b,c of the proposed heuristic, is comparable to the optimal solution.

Figure 3 demonstrates the simulation results for $\beta = 0.8$ and various grid sizes. The value of utility functions for two algorithms are plotted in Figure 3a, which shows that both algorithms perform close to each other. A similar trend is exhibited by the plots of Figure 3b,c, where the proposed heuristic not only satisfies the minimum user connectivity requirements for all cases, but also achieves a number of connected users and DBSs that is fairly close to the optimal algorithm.

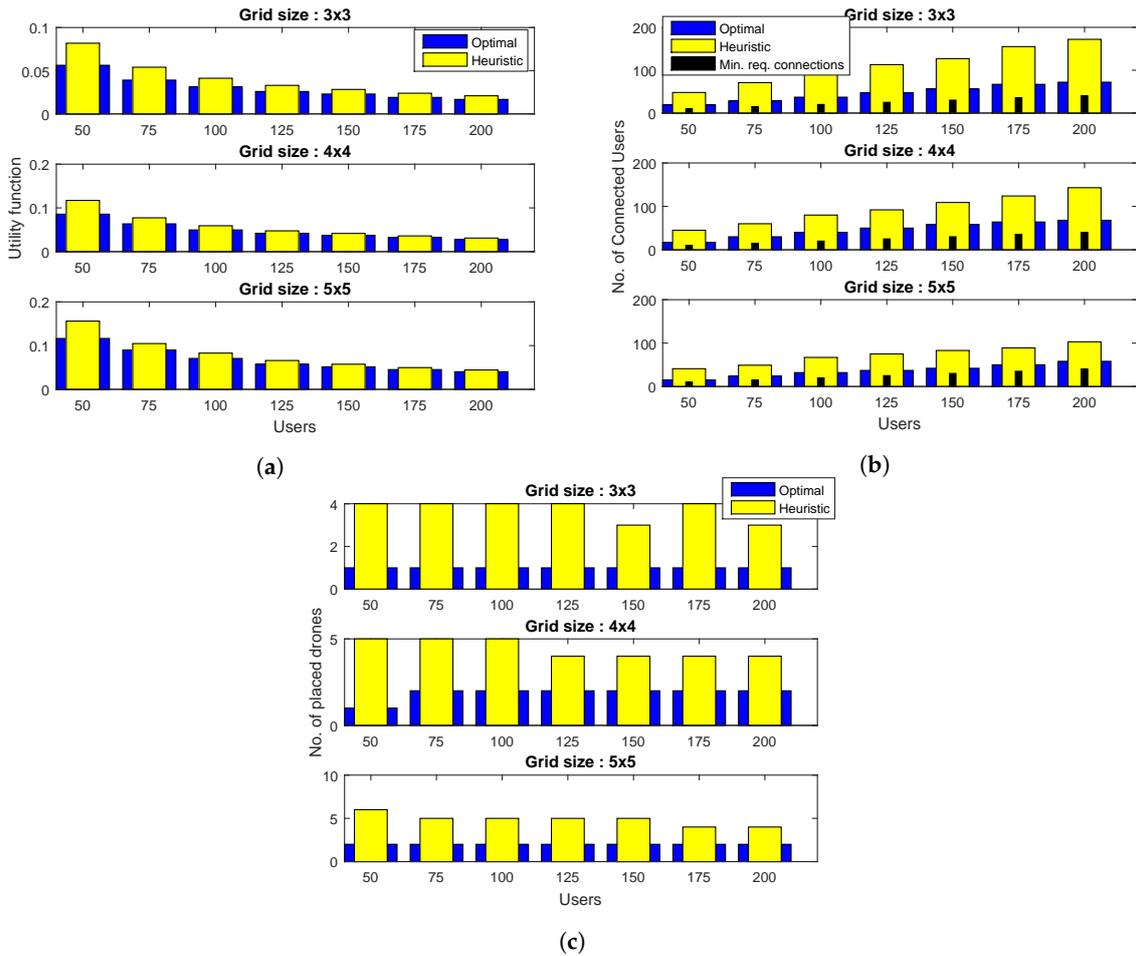


Figure 2. Comparison of proposed heuristic with the optimal algorithm for $\beta = 0.2$ and different grid sizes, in terms of: (a) The total number of connected users, (b) total number of placed drones, and (c) the utility value, that is, the ratio of the total number of placed drones to the total number of connected users.

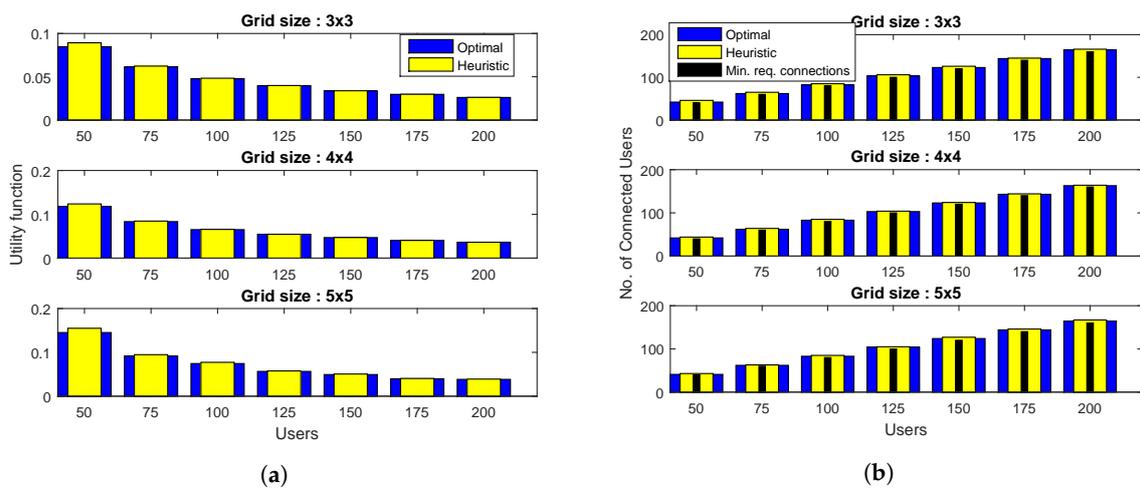
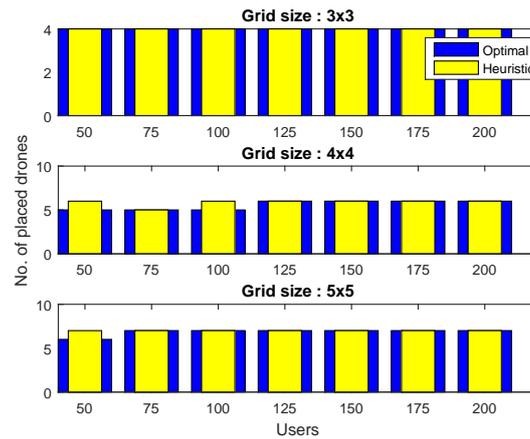


Figure 3. Cont.



(c)

Figure 3. Comparison of proposed heuristic with the optimal algorithm for $\beta = 0.8$ and different grid sizes in terms of: (a) The total number of connected users, (b) total number of placed drones, and (c) the utility value, that is, the ratio of the total number of placed drones to the total number of connected users.

Finally, in Figure 4, the confidence interval (CI) is plotted as a function of β for various values of N . CI is defined as the squared difference between the utility function value of the optimal and proposed heuristic. A very small value of CI is achieved for all cases of simulation, which validate the numerical accuracy of proposed scheme.

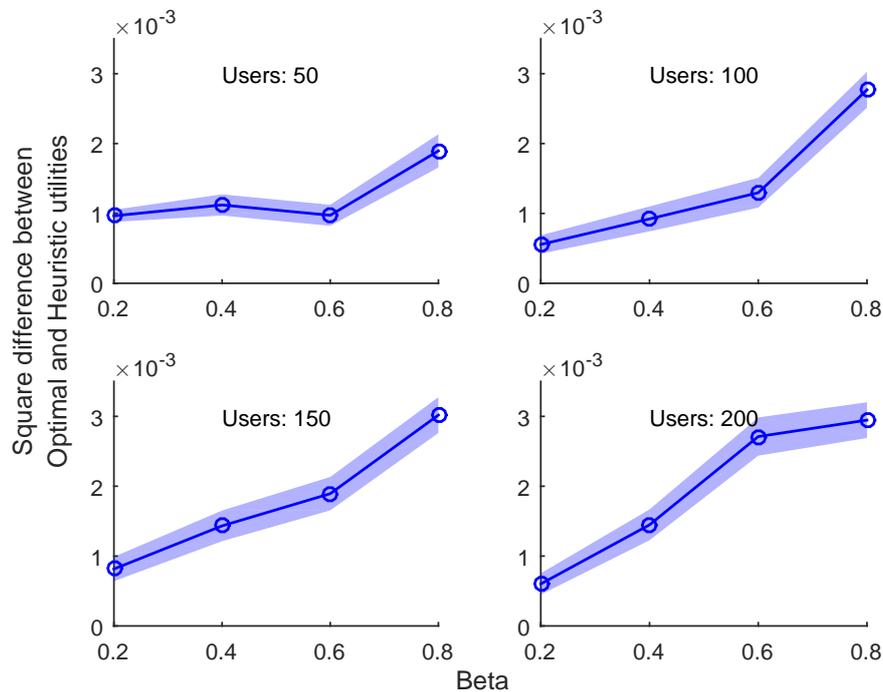


Figure 4. Confidence intervals for grid size = 3×3 .

5. Conclusions

In this paper, DBSs have been investigated as a promising solution to fulfil the aggressive connectivity and QoS requirements of large-scale IoT networks. However, DBSs being power-constrained devices themselves, a careful scheme is required for their deployment and the user association to them. This aspect was addressed in this work, by first providing a realistic mathematical model for the joint optimization problem of DBS assignment and user association in a massive, large-scale IoT network.

The mathematical model was developed by keeping in mind an air-to-ground path-loss model and practical network constraints. The objective function minimizes the ratio between the number of DBSs utilized to the number of users connected. Such an integer fractional problem suffers from prohibitive complexity when solved optimally for practical-sized networks. A linearization of the problem was proposed, along with a low-complexity heuristic algorithm for its solution in linear time. A comparison of the results of the proposed solution with the optimal one confirmed the accuracy and validity of our approach.

Author Contributions: Conceptualization, A.A. and M.A. (Muhammad Awais); Investigation, M.A. (Muhammad Awais); Methodology, A.A. and T.A.; Software, A.A., M.A. (Muhammad Awais) and T.A.; Validation, M.A. (Musaed Alhussein) and K.A.; Writing—original draft, A.A. and M.A. (Muhammad Awais); Writing—review & editing, T.A., Selman Kulac, M.A. (Musaed Alhussein) and K.A.

Acknowledgments: The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group NO. (RG-1438-034).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lagum, F.; Bor-Yaliniz, I.; Yanikomeroglu, H. Strategic Densification with UAV-BSs in Cellular Networks. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 384–387. [[CrossRef](#)]
2. Yin, S.; Tan, J.; Li, L. UAV-assisted Cooperative Communications with Wireless Information and Power Transfer. *arXiv* **2017**, arXiv:1710.00174.
3. Al-Hourani, A.; Kandeepan, S.; Lardner, S. Optimal LAP Altitude for Maximum Coverage. *IEEE Wirel. Commun. Lett.* **2014**, *3*, 569–572. [[CrossRef](#)]
4. Pereira, A.A.; Espada, J.P.; Crespo, R.G.; Aguilar, S.R. Platform for controlling and getting data from network connected drones in indoor environments. *Future Gener. Comput. Syst.* **2019**, *92*, 656–662. [[CrossRef](#)]
5. Sharma, S.; Chen, K.; Sheth, A. Toward practical privacy-preserving analytics for IoT and cloud-based healthcare systems. *IEEE Internet Comput.* **2018**, *22*, 42–51. [[CrossRef](#)]
6. Woo, M.W.; Lee, J.; Park, K. A reliable IoT system for Personal Healthcare Devices. *Future Gener. Comput. Syst.* **2018**, *78*, 626–640. [[CrossRef](#)]
7. Ahmad, A.; Ahmad, S.; Rehmani, M.H.; Hassan, N.U. A Survey on Radio Resource Allocation in Cognitive Radio Sensor Networks. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 888–917. [[CrossRef](#)]
8. Yan, Y.; Xu, C. A Development Analysis of China's Intelligent Transportation System. In Proceedings of the 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, Beijing, China, 20–23 August 2013; pp. 1072–1076.
9. Sanchez-Gomez, J.; Sanchez-Iborra, R.; Skarmeta, A. Transmission Technologies Comparison for IoT Communications in Smart-Cities. In Proceedings of the GLOBECOM 2017—2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6.
10. Fu, J.; Liu, Y.; Chao, H.; Bhargava, B.K.; Zhang, Z. Secure Data Storage and Searching for Industrial IoT by Integrating Fog Computing and Cloud Computing. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4519–4528. [[CrossRef](#)]
11. Alzenad, M.; El-Keyi, A.; Lagum, F.; Yanikomeroglu, H. 3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage. *IEEE Wirel. Commun. Lett.* **2017**, *6*, 434–437. [[CrossRef](#)]
12. Kalantari, E.; Shakir, M.Z.; Yanikomeroglu, H.; Yongacoglu, A. Backhaul-aware robust 3D drone placement in 5G+ wireless networks. In Proceedings of the 2017 IEEE International Conference on IEEE Communications Workshops (ICC Workshops), Paris, France, 21–25 May 2017; pp. 109–114.
13. Shakhathreh, H.; Khreishah, A.; Alsarhan, A.; Khalil, I.; Sawalmeh, A.; Othman, N.S. Efficient 3D placement of a UAV using particle swarm optimization. In Proceedings of the 2017 8th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 4–6 April 2017; pp. 258–263.

14. Bor-Yaliniz, R.I.; El-Keyi, A.; Yanikomeroglu, H. Efficient 3-D placement of an aerial base station in next generation cellular networks. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016; pp. 1–5.
15. Lu, J.; Wan, S.; Chen, X.; Fan, P. Energy-Efficient 3D UAV-BS Placement versus Mobile Users' Density and Circuit Power. In Proceedings of the 2017 IEEE Globecom Workshops (GC Wkshps), Singapore, 4–8 December 2017; pp. 1–6.
16. Kalantari, E.; Yanikomeroglu, H.; Yongacoglu, A. On the Number and 3D Placement of Drone Base Stations in Wireless Cellular Networks. In Proceedings of the 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall), Montreal, QC, Canada, 18–21 September 2016; pp. 1–6.
17. Lyu, J.; Zeng, Y.; Zhang, R.; Lim, T.J. Placement Optimization of UAV-Mounted Mobile Base Stations. *IEEE Commun. Lett.* **2017**, *21*, 604–607. [[CrossRef](#)]
18. Alzenad, M.; El-Keyi, A.; Yanikomeroglu, H. 3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 38–41. [[CrossRef](#)]
19. Azizi, A.; Mokari, N.; Javan, M.R. Joint Radio Resource Allocation, 3D Placement and User Association of Aerial Base Stations in IoT Networks. *arXiv* **2017**, arXiv:1710.05315.
20. Wang, H.; Wang, J.; Ding, G.; Wang, L.; Tsiftsis, T.A.; Sharma, P.K. Resource allocation for energy harvesting-powered D2D communication underlaying UAV-assisted networks. *IEEE Trans. Green Commun. Netw.* **2018**, *2*, 14–24. [[CrossRef](#)]
21. Mignardi, S.; Buratti, C.; Bazzi, A.; Verdone, R. Trajectories and Resource Management of Flying Base Stations for C-V2X. *Sensors* **2019**, *19*, 811. [[CrossRef](#)] [[PubMed](#)]
22. Huo, Y.; Dong, X.; Lu, T.; Xu, W.; Yuen, M. Distributed and multi-layer UAV network for the next-generation wireless communication. *arXiv* **2018**, arXiv:1805.01534.
23. Chen, W.; Liu, B.; Huang, H.; Guo, S.; Zheng, Z. When UAV Swarm Meets Edge-Cloud Computing: The QoS Perspective. *IEEE Netw.* **2019**, *33*, 36–43. [[CrossRef](#)]
24. Lai, C.C.; Chen, C.T.; Wang, L.C. On-Demand Density-Aware UAV Base Station 3D Placement for Arbitrarily Distributed Users with Guaranteed Data Rates. *IEEE Wirel. Commun. Lett.* **2019**. [[CrossRef](#)]
25. Liu, X.; Wang, J.; Zhao, N.; Chen, Y.; Zhang, S.; Ding, Z.; Yu, F.R. Placement and Power Allocation for NOMA-UAV Networks. *IEEE Wirel. Commun. Lett.* **2019**. [[CrossRef](#)]
26. Sharma, V.; Song, F.; You, I.; Atiquzzaman, M. Energy efficient device discovery for reliable communication in 5G-based IoT and BSNs using unmanned aerial vehicles. *J. Netw. Comput. Appl.* **2017**, *97*, 79–95. [[CrossRef](#)]
27. Motlagh, N.H.; Bagaa, M.; Taleb, T. UAV Selection for a UAV-Based Integrative IoT Platform. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–6.
28. Al-Hourani, A.; Kandeepan, S.; Jamalipour, A. Modeling air-to-ground path loss for low altitude platforms in urban environments. In Proceedings of the 2014 IEEE Global Communications Conference, Austin, TX, USA, 8–12 December 2014; pp. 2898–2904.
29. Li, H.L. A global approach for general 0–1 fractional programming. *Eur. J. Oper. Res.* **1994**, *73*, 590–596. [[CrossRef](#)]

