This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal

Efficient Deployment of Small Cell Base Stations Mounted on Unmanned Aerial Vehicles for the Internet of Things Infrastructure

Mohammad Javad Sobouti, Zahra Rahimi, Amir Hossein Mohajerzadeh, Seyed Amin Hosseini Seno, Reza Ghanbari, Johann M. Marquez-Barja, and Hamed Ahmadi

***Abstract*—In the Internet of Things networks deploying fixed infrastructure is not always the best and most economical solution. Advances in efficiency and durability of Unmanned Aerial Vehicles (UAV) made flying small cell base stations (BS) a promising approach by providing coverage and capacity in environments where using fixed infrastructure is not economically justified. A key challenge in covering an area with UAV-based small cell BSs is optimal positioning the UAVs to maximize the coverage and minimize the number of required UAVs. In this paper, we propose an optimization problem that helps to determine the number and position of the UAVs. Moreover, to have efficient results in a reasonable time, we propose complementary heuristic methods that effectively reduce the search space. The simulation results show that our proposed method performs better than genetic algorithms.**

***Index Terms*—Internet of Things, UAV, Placement, Optimization, Small Cells**

I. Introduction

With the development of the Internet of Things (IoT), it is estimated that the number of Internet-connected devices will increase by tens of billions over the next 5 to 10 years. IoT devices are used in data collecting and processing for data pattern recognition, analyzing and anticipating incidents, optimization, and eventually better and timely decision making. The massive data transfer in IoT requires effective coverage policies considering energy consumption and data rate [1].

Recently, UAVs have been substantially improved and used for various applications such as public safety and creating flying ad-hoc networks [2][3]. In IoT centric scenarios, UAVs serve as flying BSs, which provide a reliable and energy-efficient uplink for IoT [4]. Some works, including [5] used UAVs to collect data from IoT nodes. In contrast, others defined more duties for UAVs like traveling to the sites of sensor clusters, collecting data, and recharging the sensors in corresponding clusters [6]. Several works have studied UAV-based resource allocation in machine to machine (M2M) communications. In [7], to minimize the transmit power while satisfying the rate requirements of M2M devices, an optimal scheduling and resource allocation mechanism for cluster head UAV communications is proposed.

One of the main advantages of using UAVs as BSs lies in the fact that they do not require fixed infrastructure and can be deployed at any point. Moreover, with a higher altitude, for example, up to 300 meters according to DJI $S900$ characteristics [8], they communicate with the line of sight (LoS) and are less affected by channel disorders. Additionally, drones can change their positions depending on the circumstances to increase the quality of services and reduce the interference to enhance the percentage of covered users. It should be noted that with an increase in the coverage area and number of drones, in aggregate more data rate will be allocated to the users. However, the affordability and availability of drones, their path design, allocation of resources, and energy consumption are prompting some severe financial and practical challenges [8].

In this paper, we propose a mathematical model for optimum positioning of UAVs [1] as a kind of aerial BS to cover IoT nodes and collect data from them. The benefits of our model are twofold; our approach minimizes the number of UAVs, and also selects the best available positions from given candidate points to minimize the aggregate distance of IoT nodes form UAVs. Additionally, it considers covering at least a target percentage of users and provides their required data rate. We investigate the performance of our model using three different groups of candidate points and find the best strategy.

The rest of this paper is organized as follows, related works are discussed in Section II, then in Section III we introduce the mathematical model of the main problem. In Section IV, proposed problem formulation and problem-solving approaches are discussed. The implementation results and comparison between the three different modes of determining the candidate points are investigated in V and then Section VI presents our conclusions.

II. Related Works

In the literature, although UAV placement in 5G is mostly discussed, there are a few articles considering it in the IoT environment. In this work, we focus on this understudied but important area. Considering the decision on the UAVs altitudes in the articles, we can divide the related works into two groups,

MohammadJavad Sobouti, Amir Hossein Mohajerzadeh, and Seyed Amin Hosseini Seno are with the Department of Computer Engineering, Ferdowsi University Of Mashhad, Mashhad,Iran. Zahra Rahimi and Reza Ghanbari are with the Department of Applied Mathematics, Ferdowsi University of Mashhad, Mashhad, Iran. Johann Marquez-Barja is with University of Antwerp - imec, Belgium. Hamed Ahmadi is with the department of Electronic Engineering, University of York, UK.

The corresponding author is Amir Hossein Mohajerzadeh (email: mohajerzadeh@um.ac.ir)

[1]Although UAV and drone have small differences, these two terms have been commonly used interchangeably. Similarly, in the rest of this paper, we use the terms UAV and drone interchangeably.

two-dimensional (2D) and three-dimensional (3D) placement. Also, if we look at the number of UAVs considered in previous works, the literature is divided into two groups of single and multiple UAVs. In the following, we will review related works based on these categories.

In [9], initially, the optimal altitude of a drone small cell for maximum coverage and minimum required transmit power is derived. Then the problem of providing a maximum coverage using two UAVs is investigated in interference free and full interference scenarios. The authors found the optimal altitude of the UAVs and their position in both interference free and full interference scenarios. They found the optimal distance between two UAVs to minimize the interference. In [10] using brute force search, the optimum positions of UAVs are attained to deal with the disaster and improve the public safety. Additionally, [11] has suggested an active placement method for cache-enabled drones according to the message contents in order to make a better quality of experience (QoE). In this technique, a drone forecasts the contents based on a model, such as caching approach can diminish the latency of data packet transmission. Authors of [12] represent a method that finds the optimum 3D location of UAVs, which are equipped with small cell BSs with directional antennas using circle packing theory, so that the overall coverage of the region is maximized. In [13], the path and optimum position of several UAVs as aerial stations for collecting data in IoT has been investigated by exploiting the framework of optimal transport theory. In [14], the authors have considered complementing the capacity of current terrestrial macro BSs network by dynamically placing UAVs. They proposed two data field clustering algorithms in which existing terrestrial base stations do not satisfactorily address some fields. The authors of [15] proposed a polynomial-time algorithm for mobile BS placement where UAVs are placed sequentially along a spiral path to cover ground terminals (GTs) until all GTs are covered. In [16] a proactive drone-cell deployment framework has been recommended for overload reduction, which is initiated from the flash crowd in 5G. This approach assumes the cell placement as a clustering problem and considers users covered by each UAV as a cluster. Situating a UAV in the center of each cluster leads to the drone-cell to have the minimum total squared distance with all cluster members. Finally, a constraint bisecting k-means method for solving the drone placement problem has been proposed. Traffic models have been similarly studied for three social activities, including the stadium, parade, and gathering. In all the works mentioned above, UAVs are placed in a $2D$ plane with a fixed height. It should be noted that $2D$ placement of UAVs is as important and challenging as $3D$ placement, and it is especially more popular and efficient when dealing with a larger number of UAVs.

The goal of [17] is to maximize the coverage of a single UAV with base station 3D positioning and the bandwidth allocations to each user. The authors proposed a search algorithm to solve the problem and reduce the complexity, which is NP-hard due to being a Mixed Integer Non-Linear Programming problem. In [18], authors investigate a framework to provide a correlation between the supply and flash crowd traffic demands in 5G. They design a drone-cell assisted communication framework in 5G networks to boost communication coverage on flash crowd traffic. It explores the prediction and operation control schemes to identify the appropriate number and locations of drone-cells and employs the SDN technology to integrate/disintegrate drone-cells seamlessly.

The authors in [19] investigated the problem of user-demand-based UAV assignment over geographical areas subject to high traffic demands using a cost function based on the neural network. Additionally, [11] suggested an active placement method for cache-enabled drones according to the message contents to make a better quality of experience (QoE). In this technique, a drone forecasts the contents based on a model; such a cache can reduce the latency of data packet transmission. In [20], finding optimum cell boundaries and placement locations for multiple non-interfering UAVs has been studied. The purpose of this paper is to minimize total transmission power. [21] exhibits an analytical model for the discovery of an optimum height for UAVs to maximize the coverage area.

In [22], the minimum required number of drones and their optimum 3D position for covering users has been calculated. In this work, a drone acquires its coverage range by changing its altitude based on the density of users and with the purpose of reducing interference with the other small cells and also users. As the drone lowers and increases its height in the denser and less dense regions, respectively. [23] has provided a method of optimum 3D placement according to the backhaul in the two modes of user-centric and network-centric. This method examines drone robustness after selecting its location and coverage region. In [24], an algorithm is proposed that finds the optimum location of a UAV in two dimensions with the purpose of maximizing the number of users, while their consuming energy for the transmission is minimized. [24] has also obtained the optimum 3D location of UAVs for maximizing the number of covered users. Besides all the information above, the authors of [25] have presented an approach for finding the optimum 3D position of a drone-cell in which the number of served users is increased by satisfying signal-to-noise ratio (SNR) requirements.

[26] solved a joint $3D$ positioning and task offloading decision problems for UAV cloudlets in order to provision IoT services with strict latency requirements. It proposed an efcient meta-heuristic solution based on the motion of the ions optimization algorithm to solve the main mixed-integer problem. In [27] a UAV, location and user association problem from a load balancing perspective is investigated. Firstly, a clustering method to place UAVs is introduced. Then, a user association strategy is proposed where the optimization task is to minimize the maximum traffic demand of sub-regions with constraints on the capacity and the shape of sub-regions. A UAV positioning algorithm using the method of backtracking line search is proposed to refine the system's load balance. Finally, the altitude of each UAV is adjusted to decrease the power consumption of the system. By invoking user association and location algorithms, the results of UAV positions are near optimum. [28] presented an overview of optimization approaches to solve the location problem of UAV

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal

base stations. In addition to carefully reviewing the literature, [28] presented a general form of mathematical formulation for flying base stations location problem.

Most of the literature modeled the UAV positioning problem with few constraints and approximately solved it with heuristic or meta heuristic algorithms. Most of the works also consider few users to serve and few UAVs (mostly one or two UAV) to deploy. In this work, we scale up the problem by considering a large number of users, and both coverage and data rate constraints together. We also solve the problem with an exact method. Main contributions of this work are summarized as follows:

- Proposing a new approach to solve the optimal UAV positioning problem while considering coverage and data rate parameters,
- Finding candidate points for the proposed optimization model in the form of mesh refinement,
- Appropriate selection of UAV numbers with the aid of a *bi-section* technique, and
- Finding an upper-bound for the required number of UAV using greedy algorithm.

III. System Model

In this work we consider that each UAV can fly in a predefined altitude and due to its backhaul limitations it can serve a limited number of users/IoT sensor nodes[2]. To use fewer UAVs, it's better to deploy them in dense areas and minimize the distance of the UAV from its covered IoT nodes. Hence this is a bi-objective optimization problem which minimizes the number of UAVs and the distance between users and the UAV.

In this problem, we want to find the optimal location of UAVs which are covering users. There are two approaches to solve this NP-hard problem, metaheuristic algorithm, and mathematical programming. In mathematical programming, with small dimension cases, we can use methods like branch-and-bound or cutting plane approach to solve the problem optimally in efficient time. Considering a fixed altitude for UAVs, the solution space for locating UAVs is a continuous 2D space. In order to reduce the dimensions of the problem, we discretize the continuous space.

In this paper, we start with a fixed number of UAVs (P), model the problem of finding an optimal location for P UAVs, and solve it by a solver accurately in each iteration of the $bi-section$ algorithm. By doing so, we reduce a bi-objective optimization problem to a $bi - section$ and solve a single-objective optimization problem in each iteration. We discuss finding the optimal P later in IV

Finding P optimum position to place P UAVs in discrete space is an instance of P-median problem, which is a well-known problem in the field of location problems [29]. The P-median problem is locating P facilities to minimize the demand weighted average distance between demand nodes and the nearest of the selected facility. It also includes the capacitated and uncapacitated facility location problems. Since each UAV has a specific capacity, in this case, we model the problem as a capacitated P-median problem.

[2]In the rest of this paper we use users and IoT sensor nodes (nodes) interchangeably

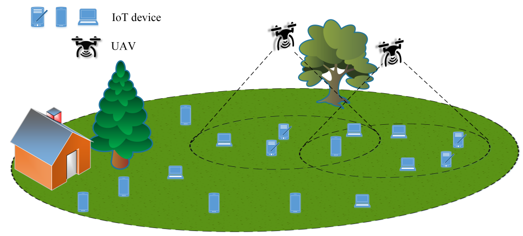


Fig. 1: A possible scenario of users covering

TABLE I: Using parameters in the mathematical model

Parameters	Description
DR	The sum data rate covered by each small cell
DRU_j	Required data rate of node j
I	Candidate points set for deploying UAVs
J	Users set
d_{ij}	The distance of UAV i from node j
α	Minimum percentage of requested coverage
R	Coverage radius of each small cell
mb_i	Mobile Base station i
P	The number of UAVs that should be deployed
D	The number of candidate points for UAV deployment
U	The total number of users

In the P-median problem, the position of candidate points to locating facilities is known. We consider the points that we obtain from the discretization of two-dimensional space as the candidate points. P-median is an NP-hard problem. Although by applying a discrete setting on the two-dimensional space still face an NP-hard problem, which trying to solve it more efficiently by an intelligent discretization of the two-dimensional space. To prevent the effect of position noises and measurement errors, we assume a normal distribution noise for users' positions error. Therefore, in general, the distance between UAVs and users will not be affected. We also assume that DBSs use dynamic channel allocation or dynamic frequency selection methods for interference avoidance. In Fig.1, we present a considered system in which the users covered by two UAVs.

IV. Problem Formulation and Proposed Method

To model the placement problem as an optimization problem, our objective is to find the optimum position of P UAVs in a way that the total distance of users from their covering UAV, such that at least α percent of users are covered, is minimized. In this problem, we assume that node coordinates, candidate points, the data rate of each small cell, and the data rate required by each node are known as denoted in Table I.

The objective function is defined to minimize the total distance of users from UAVs to deploy UAVs in optimum places. To achieve this, we need to find out which nodes should be served by which UAV; in other words, map each IoT node to a UAV. This is done by x_{ij}, which is equal to 1 if node j served by UAV i and 0 otherwise. As mentioned, we discretized the search area; thus, the UAVs will be positioned on a set of finite candidate points. In the optimization problem,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal

we denoted the candidate points with mb_i, which becomes 1 if i_{th} candidate point is selected for UAV deploying.

$$\min_{x} \sum_{i \in I} \sum_{j \in J} d_{ij} x_{ij} \tag{1a}$$

$$\text{s.t.} \quad x_{ij} \leq mb_i, \qquad \forall i \in I, \forall j \in J \tag{1b}$$

$$\sum_{i=1}^{D} x_{ij} \leq 1, \qquad \forall j \in J \tag{1c}$$

$$x_{ij} = 0, \qquad \forall i \in I, j \in J, d_{i,j} > R \tag{1d}$$

$$\sum_{j=1}^{U} DRU_j x_{ij} \leq DR, \qquad \forall i \in I \tag{1e}$$

$$\sum_{i=1}^{D} mb_i = P \tag{1f}$$

$$\sum_{j=1}^{U} \sum_{i=1}^{D} x_{ij} \geq \alpha U. \tag{1g}$$

In our formulation, (1b) states that node j can only use the candidate point i's service if that point is selected for the UAV deployment. It is clear that if candidate point i is not selected for UAV deployment, it will not be available to provide service to any node. (1c) states that each node can get the service from only one UAV. Since x_{ij} is binary variable, constraint (1c) utmost allows one of them to get 1. (1d) does not let the nodes that are out of the coverage range of one small cell to get the service from it. (1e) refers to the limited data rate of each small cell. The total data rate of nodes that receive the service from i-th UAV should not be more than the data rate of the small cell itself. (1f) lets to deploy only P UAVs. Section 5 discusses how to specify a P value that is appropriate to the cost as well as the quality of the user's service. (1g) states that the ratio of the covered nodes to the whole nodes is greater than α. This constraint guarantees that at least α percent of users will get the service. Also (1e) and (1g) guarantee that each UAV serves with a maximum data rate capacity. (1d) and (1g) are not in original P-median model. Since these constraints are like covering problem constraints, our model will be a combination of two NP-hard problems, covering and P-median. With $\alpha = 1$ in (1g) proposed model will merge to a P-median model. Therefore, finding the optimum position is also an NP-hard problem.

Knowing that we cannot reach the optimum solution for an NP-hard problem in polynomial time, to have an operational solution, heuristic and meta-heuristic methods are used. Also, reducing the problem to a smaller one, and finding the exact solution is another approach that we considered. Since the number of candidate points affects the problem complexity, we try to reduce our NP-hard problem to a smaller one by intelligently defining the set of candidate points and solve the problem exactly.

After modeling the problem, we should answer the following questions:
1) What is the best set of candidate points for deploying UAVs? As seen in describing model parameters in Table I, in this kind of modeling, the candidate points set (I) must be given to the model for deploying UAVs.
2) What is the optimal appropriate value for the number of UAVs (P)?

A. Finding a set of candidate points

In our model, we need to provide a set of candidate points. With this type of modeling and considering candidate score for a finite number of points on the page, the main optimization problem, which is the selection of P points from an uncountable number of points on the surface is converted to an integer problem to select P points (equal to the number of UAVs) from a large but finite number of candidate points.

We propose three strategies for selecting candidate points. In the first strategy, the position of the nodes is presented as candidate points. In the second strategy, we find candidate points by a simple mesh, and in the third, we find out candidate points based on the node positions with a smart meshing.

1) On Users: Nodes positions are selected as a set of candidate points for the deployment of the small cell. This strategy has been considered due to excluding the chance of being nominated from points where the user is not there. When there are many nodes, the number of candidate points increases relatively, and achieving the solution of P-median will take time.

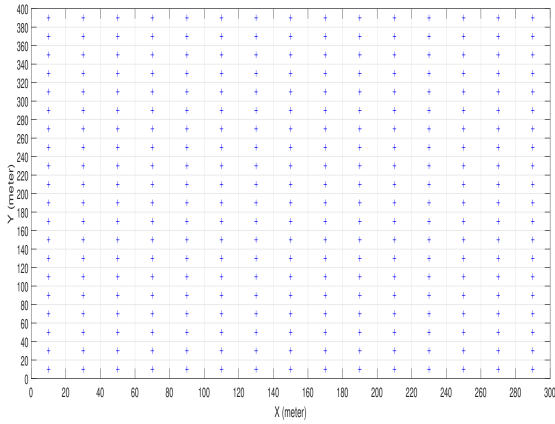
However, if a part of the environment is empty of the user, there will be no points for nominating on that coordinate. Besides, if more than one node is located near a coordinate, this point of the surface will be announced several times as a candidate and produce some identical form of constraints. To prevent this problem, it can be considered distinctive points of node positions as candidate points.

2) Simple mesh: In this approach, we mesh the surface and give the intersection points of horizontal and vertical lines as candidate points to the problem. Length and width of mesh cells do not depend on the node positions and select in such a way that if the nodes are located at the maximum distance (Fig.2a), cover all of them. It means that if users are in the worst state, each cell of simple mesh contains one user.

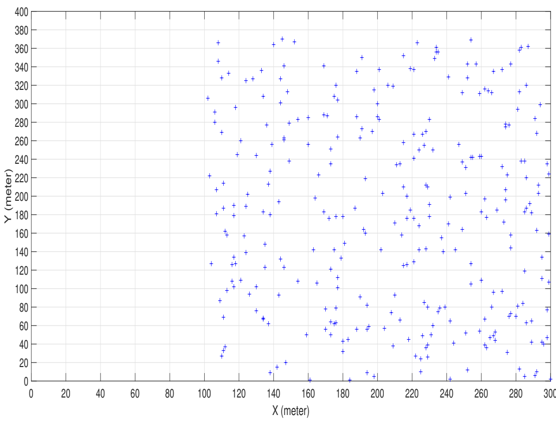
The length and width of the mesh are specified here, and as shown in Fig.2b for the other nodes placement in the surface, this mesh is still used. Although this approach has a simple calculation, high-density and low-density of the points do not affect meshing.

3) Smart mesh (Mesh refinement): this type of meshing completely depends on the node positions. In this case, when the density of each cell exceeds a distinct value (for example, one node in every 10 m^2), we have to refine it. This means that we divide that cell into four smaller ones and add created intersections to the candidate points set (Algorithm 1). The density parameter of this meshing gets two different values. For the big size cells, in order to prevent quick stop and having more numbers of candidate points, we consider a smaller density parameter than the parameter of smaller cells. This is because of nodes' density and distribution in bigger cells are less than smaller ones. Considering single parameter in the Algorithm 1, candidate points can not cover all nodes if the parameter is too high. Otherwise, the algorithm will not stop,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal



(a) Scenario 1



(b) Scenario 2

Fig. 2: Simple meshing with different user distributions

and dividing into smaller regions will continue forever. Mesh refinement provides more candidate points for denser regions to deploy UAVs. In Fig.3, the mesh refinement on a rectangular region, where the nodes are heterogeneously distributed, is presented. At first, the whole environment is considered as a cell; this cell marks by 1 (Fig.3a). The cell's density is compared with the density parameter; if it is dense, we divide the cell into four smaller cells (Fig.3b). This procedure will be done for all cells until no dense cell remains (Fig.3c-3d).

Although this kind of finding the candidate points needs a little more time than the preceding two methods, considering the node density during the meshing is one of its strengths.

B. Finding optimal P

Practically, we need a suitable P before solving our mathematical model. The proposal of this paper for finding the optimal P is following as: Firstly, we need to find P_{max}. In fact, P_{max} is a value that guarantees the problem based on this value for P is feasible. Then, the space of $[1, P_{max}]$ is explored by binary search. It means actually, we solve the problem's mathematical model at most $\log_2^{P_{max}}$ times.

Algorithm 1 Mesh Refining

$k \longleftarrow 1$;
for (each node i)
  node label(i) $\longleftarrow 1$;
end for
while ($k \leq$ number of cells)
  if ($cell_k$ is big)
    density parameter = a;
  else
    density parameter = b;
  end if
  while ($\frac{\text{total number of nodes with label } k}{\text{area space of } cell_k} \geq$ density parameter)
    $t \longleftarrow$ total number of cells;
    divide $cell_k$ into four equal cells;
    mark new cells with $k, t+1, t+2, t+3$;
    update the node labels according to its cell mark;
  end while
  $k \longleftarrow k+1$;
end while

The most straightforward choice for P_{max} is the number of nodes. However, in order to increase the efficiency and reduce the search space, we propose a greedy algorithm to find a suitable P_{max}, which its steps is described in Algorithm 2. In this greedy algorithm, first, we arrange the nodes in the order of the number of neighboring nodes in descending order, then we put a DBS on the first dense node. As long as the small cell's data rate allows, nodes that are in the range of the DBS are assigned to the small cell. Then we reorder the remaining nodes and continue the process until all nodes are covered. We can also reduce search space to $[P_{min}, P_{max}]$ which P_{min} is obtained from equation (2).

$$P_{min} = \frac{\alpha * NumberofUsers * Meandatarate}{UAV_{datarate}} \tag{2}$$

Lemma 1. *P_{min} in equation (2) is a lower bound for the number of UAVs required to cover α percent of users with specified data rate.*

Proof. Suppose UAVs have no limit on the coverage radius. The number of needed DBSs, in this case, will not be more than the number of required DBSs when they have a limited coverage radius. Without affecting the generality, suppose that

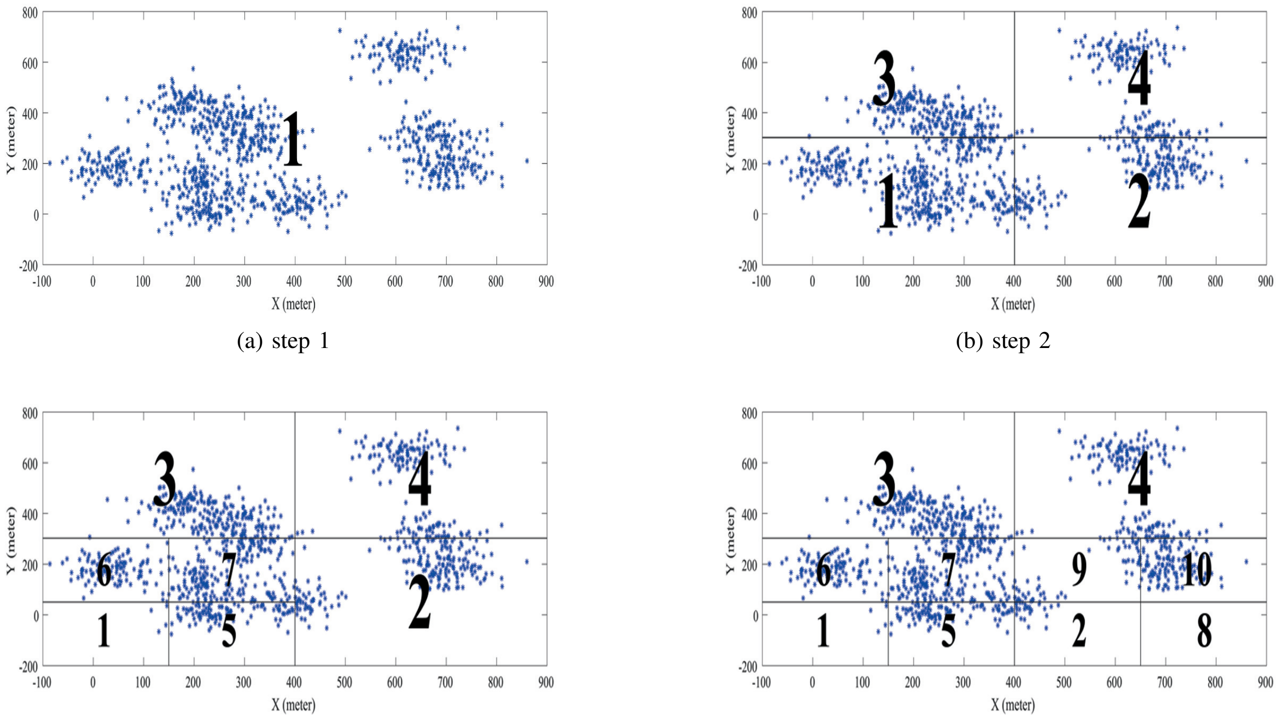
$$DRU_1 \leq DRU_2 \leq \cdots \leq DRU_N$$

where DRU_j is required data rate for user j and N is the number of users. By covering users with less DRU, it can reach α percent of users' coverage with fewer UAVs. So the minimum number of UAVs to cover α percent of users is equal to

$$\frac{\sum_{j=1}^{\alpha N} DRU_j}{UAV_{datarate}}$$

But since we do not have exact DRU values when calculating the lower bound, we use the average of distribution which is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal



(a) step 1 (b) step 2 (c) step 3 (d) step 4

Fig. 3: Mesh Refinement steps

estimated for the requested data rate instead. So we have:

$$P_{min} = \frac{\sum_{j=1}^{\alpha N} Meandatarate}{UAV_{datarate}} = \frac{\alpha * N * Meandatarate}{UAV_{datarate}}$$

□

Algorithm 2 A greedy algorithm for finding P_{max}

Mark all nodes as uncovered
for (each node i)
  $A(i) \longleftarrow$ the number of nodes in R-radius of node i;
end for
while (all nodes are not covered)
  $i_0 \longleftarrow$ arg max A;
  put a UAV at point i_0;
  while (data rate of the small cell has not reached)
    assign its around nodes to the small cell;
    mark assigned node as covered;
    $A(i_0) \longleftarrow 0$;
  end while
end while

After choosing one of the candidate points methodologies, we find P_{max} using Algorithm 2. Then we calculate P_{min}, then we solve the problem of locating $p = \frac{P_{min}+P_{max}}{2}$ UAVs by solving the mathematical model using Cplex[3] [30]. If the mathematical model has a feasible solution and α percent of the nodes are covered, we update $P_{max} = p$ and otherwise update $P_{min} = p$, then resolve the problem of locating $p = \frac{P_{min}+P_{max}}{2}$ UAVs. We continue this process while P_{max} is larger than P_{min}. The solving procedure of the proposed optimization model is represented in figure 4.

[3]A solver software that is commercially available.

C. Metaheuristic approach

In the next section, we compare the proposed mathematical model with the genetic algorithm (GA). In fact, in each iteration of the $bi-section$ algorithm, instead of solving the mathematical model of locating p UAVs, the problem is solved by the genetic algorithm presented in [31]. If the solution is feasible, the $bi-section$ algorithm decreases P. Otherwise, P increased, and the genetic algorithm reruns for the updated P. In the following, we discuss our benchmarking GA briefly. In GA, each feasible solution is represented as a chromosome that has a certain number of genes. In our approach, each chromosome has P gens, where P is the number of UAVs, and the value of each gene denotes the index of a candidate point.

The procedure of finding how good (fit) each solution (chromosome) is called fitness evaluation. In order to find the fitness of a chromosome, it must first determine which nodes are assigned to each gene (selected candidate point) in a chromosome.

In the genetic algorithm, presented in [31] fitness procedure tries to assign each node to the nearest selected candidate point. Since each drone has a fixed bandwidth, some nodes will have to be assigned to second, third, and other nearest candidate points. Suppose there is an assignment conflict,

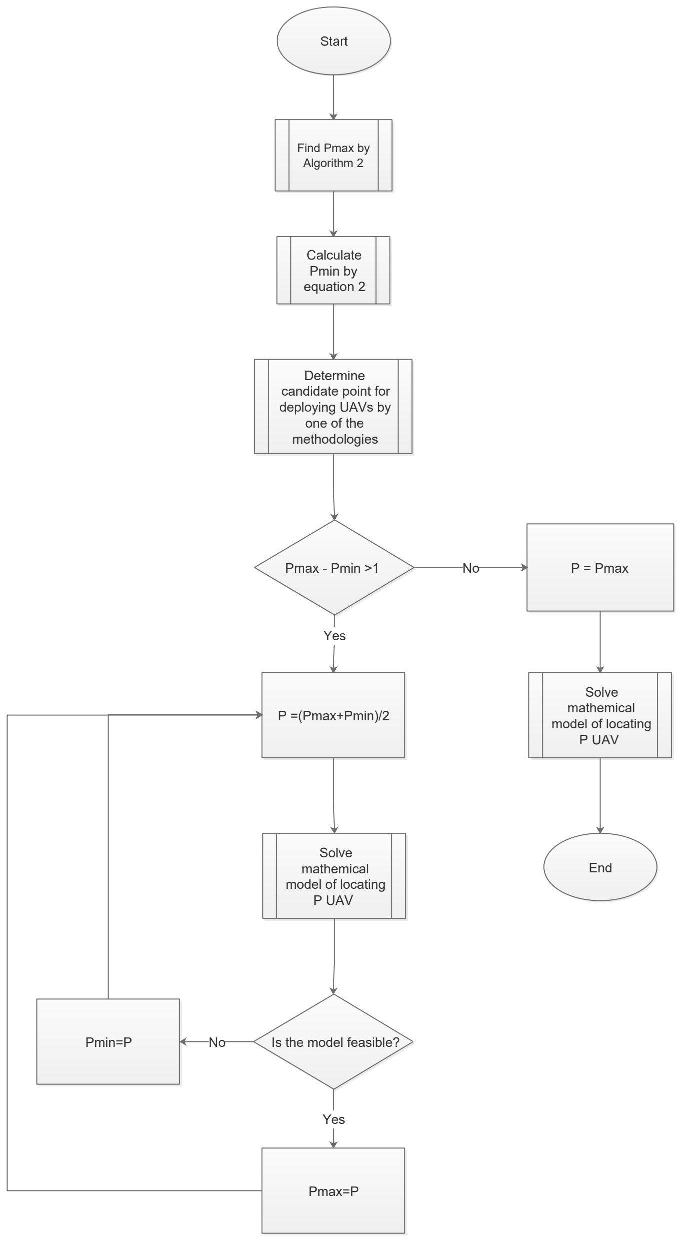


Fig. 4: Solving UAV placement problem

for example, a candidate point can serve to only one more node, but this candidate point is the nearest for two nodes. The procedure prefers to assign the node that would be most intolerant if it assigned to the second nearest candidate point. The difference between the two distances determines the intolerance of a node, distance between the node, and the nearest candidate point and distance between the node and the second nearest candidate point. Once the assignment procedure complete, the fitness of a chromosome can be computed. In our case for a chromosome, it is likely that less than 90% of the nodes are within the allowed range of selected candidate points. In this case, by adding a large number to the fitness, we try not to select the chromosome with this condition, as it will denote an infeasible solution.

After determining the fitness of each chromosome in the initial population, the best chromosomes are selected to generate a new generation. We can use two types of selection schemes. Proportionate-base selection picks out individuals base upon their fitness relative to the fitness of other individuals. Ordinal-base selection select individuals upon their rank in population. In this paper, we use proportionate-base selection. Finally, a new generation is created by applying the crossover operator on the selected chromosomes. Crossover is a recombination operator that proceeds in three steps:

- Select two chromosomes for mating at random.
- Select a cross site at random along a chromosome length.
- Swap position values between two chromosomes.

After repeating the whole process for a predetermined number of generations, the best chromosome in the last generation will be select as the best solution.

TABLE II: Test parameters for evaluating the problem model

Parameters	Description
$Region$	500×500
$numberofnodes$	200, 300, 400, 500
α	90%
J	Users set
d_{ij}	The distance of UAV i from node j
α	Minimum percentage of requested coverage
R	$40, 60m$
$UAV_{datarate}$	$20Mbps$
$GeneticPopulation$	1000
$GeneticSteps$	40

V. Numerical results

A. Test System

In this section, we look at the implementation results of the optimization placement model for UAVs using the three presented approaches and GA. For simulation, we consider a 500×500 meters area with scenarios including 500, 400, 300, and 200 nodes with four different distributions from dense to scattered using Poisson Point Process. In this optimization problem, we want to achieve the minimum number of UAVs for covering at least 90% of users ($\alpha = 0.9$) according to the quality constraints. Additionally, the data rate of each UAV assumed 20 megabit per second, which is the limitation of the sum of covered users' uplink, and their flying altitude is 40 and 60 meters. Here we considered elevation angle of 45 degrees to the coverage radius would be 40 and 60 meters, respectively, which is related to the DJI drone specifications and power consumption. P_{max} obtained through implementing Algorithm 2 on the data of nodes while the minimum number of possible UAVs to cover users' data rate has been achieved according to equation (2). These parameter values for each scenario are shown in table II.

Based on implementing the *bi-section* algorithm in order to find the optimal P, we get the lowest necessary UAVs with at most $\log_2^{(P_{max}-P_{min})}$ times execution of the optimization model. Also, the population of GA is 1000, and the number of steps is equal to 40. Moreover, candidate points given to the GA are reached from the simple meshing of the surface using the smallest size of mesh refinement cell. All of these parameters are shown in table II. For time consumption we ran algorithms and Cplex studio IDE, for solving proposed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal

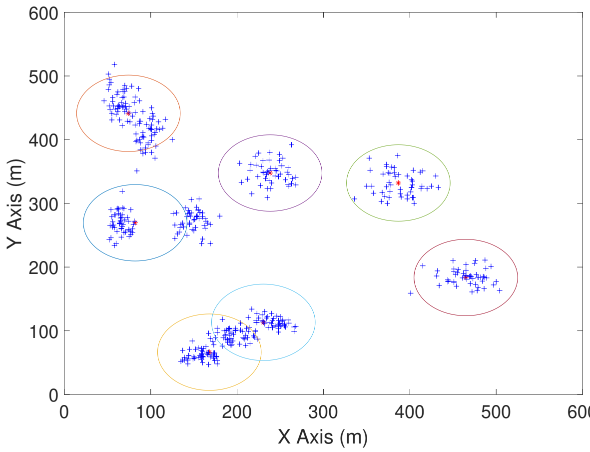


Fig. 5: UAV position using smart mesh method

mathematical model, on a system with $12GB$ RAM and $2.4GHz$ Core-i5 CPU.

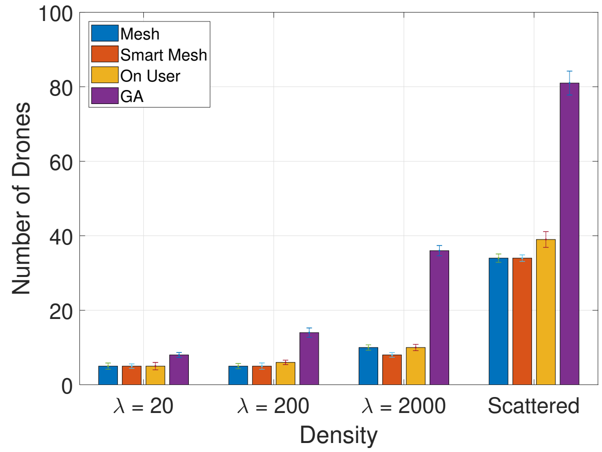
B. Results

In the following, we compare these three approaches of candidate points' selection as inputs of the optimization problem with each other and with the results of the GA in each scenario. Fig.5 illustrates how UAVs are deployed via solving the optimization problem that derives candidate points with the proposed refined mesh method.

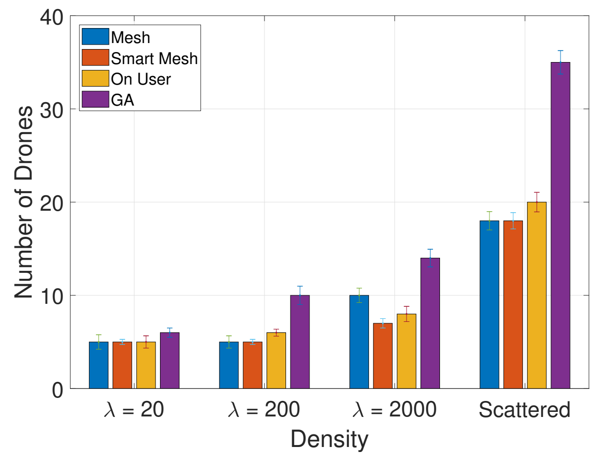
Figure 6 compares the essential UAV numbers for covering 500 users. All suggested methods should cover more than 90% of nodes due to existing constraint (1g). Although one of the goals is to use the lowest number of UAVs, all three methods of finding candidate points cover at least 90% of nodes. These results show that even though the mesh refinement method covers the same number of nodes, it needs fewer UAVs to cover them. In many scenarios, the number of required UAVs is equal to the least possible number of UAVs covers $\alpha\%$ of users, and it is clear that this number is the optimum solution of the problem. In others, because of nodes distribution, it is not possible to cover all nodes with the number of UAVs obtained from equation (2).

Figures 8a-8c present number of drones required with 40 meter altitude to cover 200, 300 and 400 nodes with two different distributions PPP with $\lambda = 2000$ and scattered. As shown in these figures, our proposed method needs fewer UAVs to cover nodes comparing with GA. Also figures 8d-8f shows results of covering nodes with 60 meter altitude drones. In this scenario, GA needs more UAVs too. The difference in the outcome of the different methods of selecting the candidate points in our proposed method is also less enough to be neglected. As seen, the refined mesh is more successful than other methods due to finding high-density points with less number of UAVs.

The standard deviation of the number of drones with different altitudes is shown in figures 6 and 8, which illustrate that in both altitudes, the results of smart mesh for number of DBSs have less deviation from the average. Therefore it is the most reliable method for determining candidate points to solve the optimization problem. To compare with the lower bound,



(a) 40m altitude



(b) 60m altitude

Fig. 6: mean drone number required for 500 user with a) 40m and b) 60m altitude of UAVs

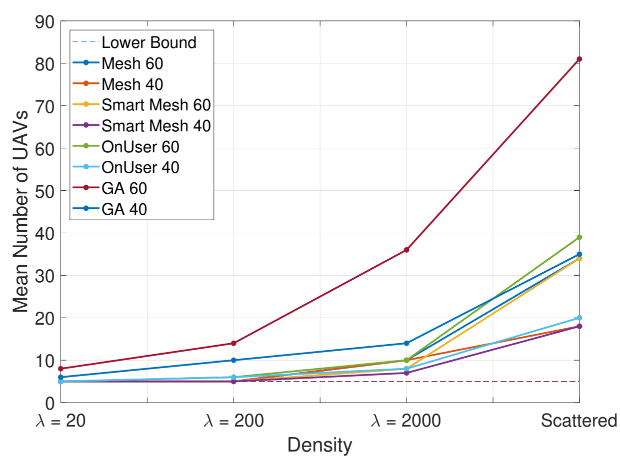
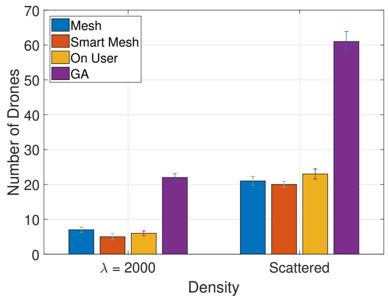
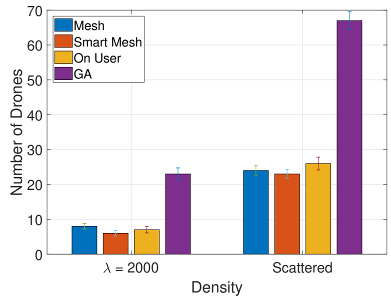


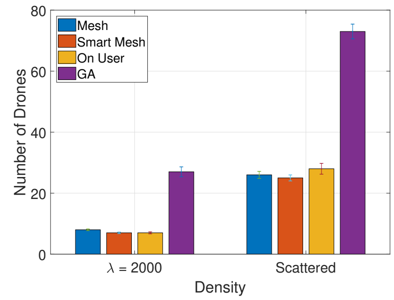
Fig. 7: Comparison between lower bound of P and the optimization result



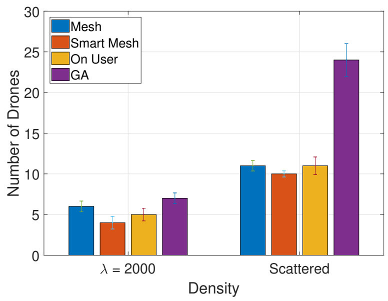
(a) 200 nodes and 40m UAV altitude



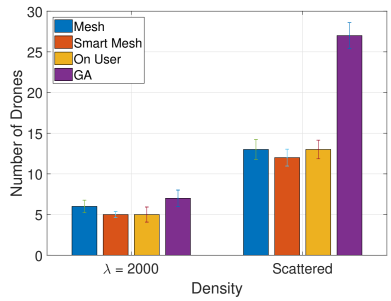
(b) 300 nodes and 40m UAV altitude



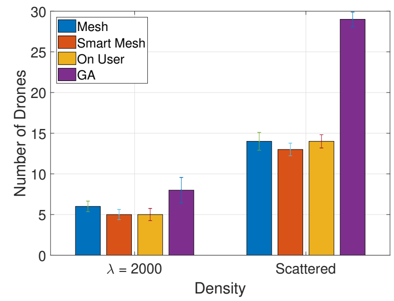
(c) 400 nodes and 40m UAV altitude



(d) 200 nodes and 60m UAV altitude

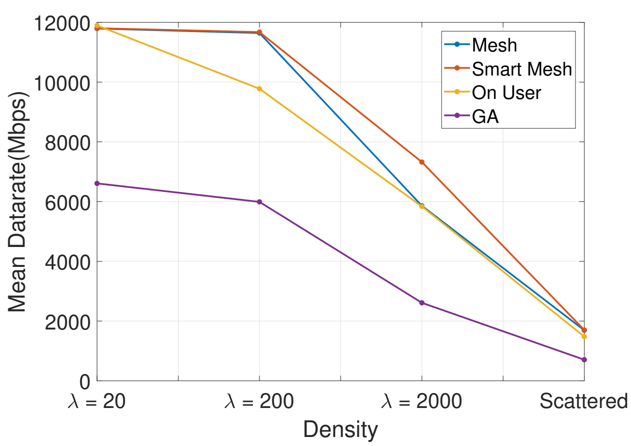


(e) 300 nodes and 60m UAV altitude

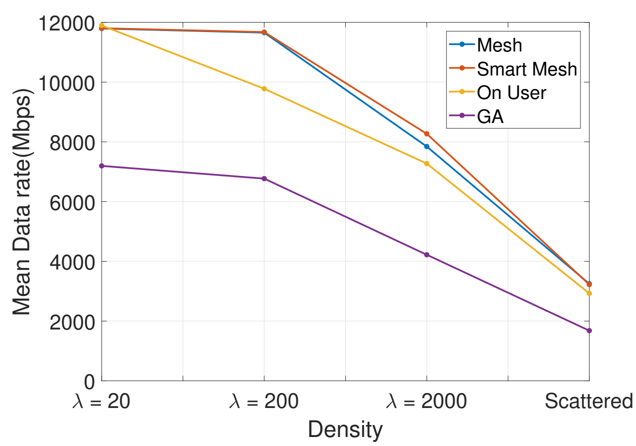


(f) 400 nodes and 60m UAV altitude

Fig. 8: Mean drone number required in other different scenarios



(a) Mean data rate served to 500 users by each small cell with 40m altitude



(b) Mean data rate served to 500 users by each small cell with 60m altitude

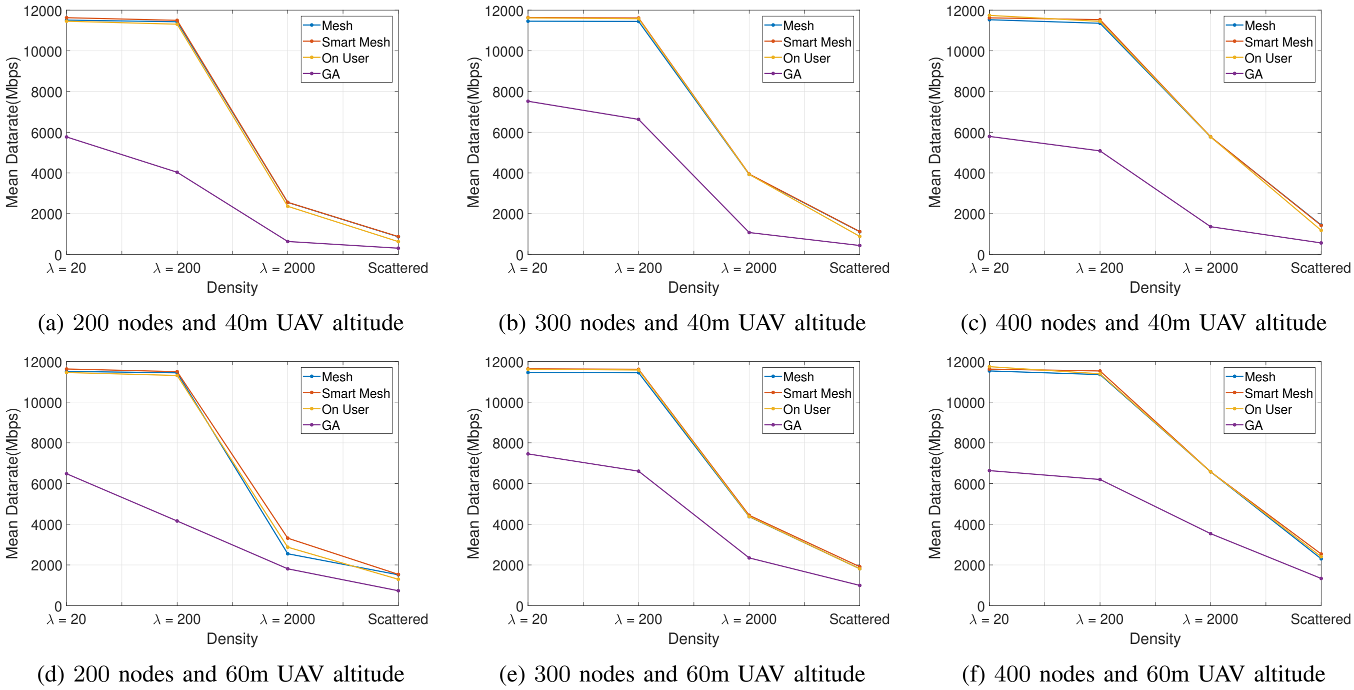
Fig. 9: Mean data rate served to 500 users by each small cell with a) 40m and b) 60m altitude

which is calculated from equation (2), figure 7 shows that in users' dense distributions, which are $\lambda = 20$ and $\lambda = 200$ the results of the optimization model is equal to the lower bound. In other distributions because of users' positions and their distance from each other more UAVs needed to cover them. However, the smart mesh method requires the least possible UAVs to cover users.

Figure 9 shows a comparison of mentioned methods established upon data rate that they covered in different distributions for 500 users. Similar to Fig. 8, figures 10a-10c and 10d-10f represent mean data rate served by each small cell to 200, 300 and 400 nodes with 40 and 60 meter altitude drones, respectively. According to this comparison, in overall, the refined mesh method has the best operation because of having more average data rate in each small cell, consequently, using UAVs more efficient and also covering more data rate using less UAVs. In a condition that the candidate points have coordinates exactly like the nodes, a better operation than simple mesh is expected, because high-density locations have more chance for deploying UAVs.

The comparison of these methods in terms of execution time shows that the refined mesh method obtained the optimal answer much faster than the others on average. The average execution time for running each epoch of our method using

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2973320, IEEE Sensors Journal



(a) 200 nodes and 40m UAV altitude (b) 300 nodes and 40m UAV altitude (c) 400 nodes and 40m UAV altitude

(d) 200 nodes and 60m UAV altitude (e) 300 nodes and 60m UAV altitude (f) 400 nodes and 60m UAV altitude

Fig. 10: Mean data rate served by each small cell in other different scenarios

mesh, smart mesh, and on user candidate point strategy is 165689.44, 14790.16, 16230.25 milliseconds(ms) respectively. Meanwhile, each epoch of GA took 87434.94 ms on average.

To compare memory usage, one of the most effective factors in the amount of memory usage for each of the mentioned strategy is the number of branches and cuts in $BranchandCut$ algorithm in the CPLEX solver. The number of decision variables in the mathematical model has a high impact on the number of branches. Since the number of decision variables in each strategy has a direct relation to the number of candidate-points, the method which produces fewer candidate points is expected to occupy less memory. Because of the smart offering of candidate-points in the smart mesh method, this method provides the lowest number of candidate-points. On user strategy and simple mesh rank second and third in terms of the number of candidate-points, respectively. The average amount of memory usage in mesh, smart mesh, on user and GA are 268.3, 158.6, 206.4 and 351.7 Megabytes(MB), respectively.

We ran all methods for four types of node distribution, from dense to scattered and compared the results. These results show that our proposed method performs significantly better than GA. We should mention that the results of our proposed method are exact, but GA cannot reach the exact answer. In our proposed method, different strategies to determine candidate points in some cases have different results. But the smart mesh method overally has the best solution comparing other methods. It is due to in different distribution scenarios the smart mesh method has the best candidate points in terms of candidate point number, density, and position. In terms of number, other methods always have a constant number of candidate points, and in terms of density and position they do not decide the density and position of candidate points due to users density, but smart mesh method determines number, position, and density of candidate points considering these terms. This decision leads to have better and more reliable results in less execution time.

To solve the optimization problem, the smart mesh method is reached to the ideal answer of UAV positions in less time compared to the others. In addition to using less number of UAVs to cover nodes, it also supports more average data rate. The lower execution time affects in a demanded higher number of UAVs. Moreover, attaining the appropriate number of UAVs needs more execution time.

VI. Conclusion

In this paper, to determine the positions of UAVs in the desired environment, a mathematical optimization model based on P-median has been proposed. Considering that solving this problem is not possible at a reasonable time, to get the P value, which is the number of UAVs, the surface discreted and the *bi-section* method has been used. P-median needs candidate points to select P points from them. Three methods have been suggested for candidate points, including exploratory on user, simple meshing, and mesh refinement (smart meshing). After implementing simulation and solving the optimization problem, according to the acquired results, it is generally evident that the refined mesh works better and reach the optimum solution based on less number of UAVs and more covered data rate.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies,

protocols, and applications," *IEEE communications surveys & tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (fanets): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.

[3] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint." *IEEE Communications Surveys and Tutorials*, vol. 18, no. 4, pp. 2624–2661, 2016.

[4] T. Park, N. Abuzainab, and W. Saad, "Learning how to communicate in the internet of things: Finite resources and heterogeneity," *IEEE Access*, vol. 4, pp. 7063–7073, 2016.

[5] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient internet of things communications," *IEEE Transactions on Wireless Communications*, vol. 16, no. 11, pp. 7574–7589, 2017.

[6] Y. Pang, Y. Zhang, Y. Gu, M. Pan, Z. Han, and P. Li, "Efficient data collection for wireless rechargeable sensor clusters in harsh terrains using UAVs," in *Global Communications Conference (GLOBECOM), 2014 IEEE*. IEEE, 2014, pp. 234–239.

[7] M. N. Soorki, M. Mozaffari, W. Saad, M. H. Manshaei, and H. Saidi, "Resource allocation for machine-to-machine communications with unmanned aerial vehicles," in *Globecom Workshops (GC Wkshps), 2016 IEEE*. IEEE, 2016, pp. 1–6.

[8] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016.

[9] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Drone small cells in the clouds: Design, deployment and performance analysis," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.

[10] A. Merwaday and I. Guvenc, "Uav assisted heterogeneous networks for public safety communications," in *Wireless Communications and Networking Conference Workshops (WCNCW), 2015 IEEE*. IEEE, 2015, pp. 329–334.

[11] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1046–1061, 2017.

[12] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage." *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016.

[13] ——, "Mobile internet of things: Can uavs provide an energy-efficient mobile architecture?" in *2016 IEEE global communications conference (GLOBECOM)*. IEEE, 2016, pp. 1–6.

[14] S. Iellamo, J. J. Lehtomaki, and Z. Khan, "Placement of 5g drone base stations by data field clustering," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*. IEEE, 2017, pp. 1–5.

[15] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, "Placement optimization of uav-mounted mobile base stations," *IEEE Communications Letters*, vol. 21, no. 3, pp. 604–607, 2016.

[16] P. Yang, X. Cao, C. Yin, Z. Xiao, X. Xi, and D. Wu, "Proactive drone-cell deployment: Overload relief for a cellular network under flash crowd traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 10, pp. 2877–2892, 2017.

[17] C. T. Cicek, T. Kutlu, H. Gultekin, B. Tavli, and H. Yanikomeroglu, "Backhaul-aware placement of a uav-bs with bandwidth allocation for user-centric operation and profit maximization," *arXiv preprint arXiv:1810.12395*, 2018.

[18] P. Yang, X. Cao, C. Yin, Z. Xiao, X. Xi, and D. Wu, "Proactive drone-cell deployment: Overload relief for a cellular network under flash crowd traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 10, pp. 2877–2892, 2017.

[19] V. Sharma, M. Bennis, and R. Kumar, "UAV-assisted heterogeneous networks for capacity enhancement," *IEEE Communications Letters*, vol. 20, no. 6, pp. 1207–1210, 2016.

[20] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Optimal transport theory for power-efficient deployment of unmanned aerial vehicles," in *Communications (ICC), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–6.

[21] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal lap altitude for maximum coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.

[22] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3d placement of drone base stations in wireless cellular networks," in *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*. IEEE, 2016, pp. 1–6.

[23] E. Kalantari, M. Z. Shakir, H. Yanikomeroglu, and A. Yongacoglu, "Backhaul-aware robust 3d drone placement in 5G+ wireless networks," in *Communications Workshops (ICC Workshops), 2017 IEEE International Conference on*. IEEE, 2017, pp. 109–114.

[24] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-d placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, 2017.

[25] R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," in *Communications (ICC), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–5.

[26] R. Islambouli and S. Sharafeddine, "Optimized 3D Deployment of UAV-Mounted Cloudlets to Support Latency-Sensitive Services in IoT," *IEEE Access*, vol. PP, p. 1, 2019.

[27] Y. Sun, T. Wang, and S. Wang, "Location Optimization and User Association for Unmanned Aerial Vehicles Assisted Mobile Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10 056–10 065, 2019.

[28] C. T. Cicek, H. Gultekin, B. Tavli, and H. Yanikomeroglu, "UAV Base Station Location Optimization for Next Generation Wireless Networks: Overview and Future Research Directions," *2019 1st International Conference on Unmanned Vehicle Systems-Oman, UVS 2019*, pp. 5–7, 2019.

[29] P. B. Mirchandani and R. L. Francis, *Discrete location theory*, 1990.

[30] I. ILOG, "IBM ILOG CPLEX V12. 1 User's Manual for CPLEX," *IBM, Armonk, NY*, 2009.

[31] M. Mangla, R. D. Akhare, S. S. Ambarkar, and N. Mumbai, "Implementing genetic algorithm to solve facility location problem," 2015.