An Efficient 3D Positioning Approach to Minimize Required UAVs for IoT Network Coverage

Zahra Rahimi, Mohammad Javad Sobouti, Reza Ghanbari, Seyed Amin Hosseini Seno, Amir Hossein Mohajerzadeh, Hamed Ahmadi, *Senior Member, IEEE*, and Halim Yanikomeroglu, *Fellow, IEEE*

***Abstract*—Using Unmanned Aerial Vehicles (UAVs) to cover users in wireless networks has increased in recent years. Deploying UAVs in appropriate positions is important to cover users and nodes properly. In this paper, we propose an efficient approach to determine the minimum number of required UAVs and their optimal positions. To this end, we use an iterative algorithm that updates the number of required UAVs at each iteration. To determine the optimal position for the UAVs, we present a mathematical model and solve it accurately after linearizing. One of the inputs of the mathematical model is a set of candidate points for UAV deployments in 2D space. The mathematical model selects a set of points among candidate points and determines the altitude of each UAV. To provide a suitable set of candidate points, we also propose a candidate point selection method: the MergeCells method. The simulation results show that the proposed approach performs better than the 3D P-median approach introduced in the literature. We also compare different candidate point selection approaches, and we show that the MergeCells method outperforms other methods in terms of the number of UAVs, user data rates, and simulation time.**

***Index Terms*—UAV, Positioning, Optimization, Internet of Things**

I. Introduction

The use of flying platforms such as UAVs is expanding rapidly in a wide range of wireless network applications. With their mobility, flexibility, and adaptability to different altitudes, UAVs are poised to become a key platform in wireless systems [1]. UAVs as aerial Base Stations (BSs) can be used to improve coverage, capacity, and reliability in wireless networks. They can also be used to enable large-scale wireless communications in next generation wireless networks. As an example, UAVs can complement existing cellular systems by providing additional capacity for hotspots [2], [3].

UAVs can also provide or increase network coverage in emergencies for public safety, where terrestrial BSs cannot be installed or may be costly, such as in mountainous areas, at sea, in emergency situations, or in case of natural disasters [4]. Compared to conventional terrestrial BSs, one advantage of using UAV BSs is their ability to provide on-the-fly communications and to establish Line-of-Sight (LoS) connections with ground users [5]. If UAVs are properly deployed and operated, they can provide reliable telecommunication solutions for a variety of real-world scenarios at a reasonable cost [6].

Since the proper deployment of UAVs increases the reliability of air-to-ground links and offers better coverage for users, their optimal placement has been studied in a number of works. In [7], for instance, we proposed an efficient 2D positioning algorithm for multiple UAVs covering many Internet of Things (IoT) nodes. We determined the minimum number of required UAVs by using a bisection algorithm. We proposed a mathematical model based on P-median to find the proper positions for the UAVs on a 2D plane where all UAVs were at the same altitude. We also considered an identical altitude for each UAV. As the mathematical model required candidate points to determine the proper position of the UAVs, we proposed a smart mesh approach, which yielded better results than a simple mesh or on-user approach. The literature in 2D positioning is further discussed in [7], [8].

Since UAVs operate in three dimensions, their altitude is also an important consideration. The authors in [9] presented an analytical model to find the optimal altitude of a UAV to maximize area coverage. They showed that, although increasing the altitude increases the LoS probability, the altitude increase also increased signal path loss. In [10] the outage probability at mmWave and sub-6 GHz frequency is investigated for different blockage environments and UAV altitudes. The authors first derived analytical approximate expressions for the outage probability. The analyzed the impact of antenna gain for two candidate frequencies on the fronthaul link. In [11] the optimal UAV altitude is derived to maximize the ground coverage and minimize the transmit power. Then, the problem of maximum coverage using two UAVs is investigated. The authors in [5] treated the UAV positioning problem by decoupling the vertical and horizontal dimensions. In so doing, they modeled the horizontal dimension problem as a circle placement problem. To find the proper altitude in the vertical dimension, they solved the enclosing circle problem. They also proposed an algorithm in [12] to maximize the coverage of users with different Quality of Service (QoS). In [13], the authors found the optimal 3D locations of UAV-BSs in various environments to maximize the number of users covered, while taking network revenue into account and proposing a computationally efficient numerical solution. The

Zahra Rahimi and Reza Ghanbari are with the Department of Applied Mathematics, Ferdowsi University of Mashhad, Mashhad, Iran, (e-mail: rahimi.zahra@mail.um.ac.ir; rghanbari@um.ac.ir).

MohammadJavad Sobouti, Amir Hossein Mohajerzadeh, and Seyed Amin Hosseini Seno are with the Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad,Iran, (e-mail: javad.sobouti@mail.um.ac.ir; {mohajerzadeh, hosseini}@um.ac.ir).

Hamed Ahmadi is with the Department of Electronic Engineering, University of York, UK, (email: hamed.ahmadi@ieee.org).

Halim Yanikomeroglu is with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada, (e-mail: halim@sce.carleton.ca).

The corresponding author is Reza Ghanbari (email: rghanbari@um.ac.ir).

authors also found the optimal 3D location of a UAV-BS to maximize the number of users whose Signal-to-Noise Ratio (SNR) requirement is met. In [14], the authors addressed the problem of finding the 3D location of one UAV and the bandwidth allocation for each user to maximize the profitability of the provided service. Since the problem was modeled as a Mixed Integer Non-Linear Programming problem, to overcome the complexity, a search algorithm was proposed. The authors then examined the algorithm's robustness after selecting the location of the UAV and its coverage area. In [6], the optimal 3D backhaul aware UAV positioning in both user-centric and network-centric approaches was discussed. In the proposed model of [6], both the total number of users and the sum rates were maximized, in each approach.

The aforementioned works considered only a single UAV in the placement problem, while in realistic scenarios there may be multiple UAVs. In [15], finding optimal cell boundaries and locations for multiple non-interfering UAVs were investigated. The objective of [15] was to minimize the total transmission power of UAVs. The jointly efficient 3D placement and mobility of the UAVs, device-UAV association, and uplink power control were discussed in [16]. The optimal UAV locations were determined in consideration of active IoT device locations and their maximum transmission power. The aim of the authors in [16] was to maximize the sum transmit power of IoT devices. In [17], the optimal location of UAVs in disaster situations and to improve public safety was established using a brute force search. The authors in [18] addressed the issue of deploying multiple UAVs and expanding UAV mapping to high demand traffic areas using a neural network-based cost function. In [19], 3D UAV-BS placement was investigated to maximize the number of covered users with different QoS requirements consuming the minimum energy. It modeled the problem as a multiple concentric circles placement problem with the objective of maximizing the numbers of covered users. To this end, it decoupled the UAV-BS deployment problem in the vertical and horizontal dimensions. The authors formulated a Mixed Integer Second Order Cone Problem (MISOCP) and proposed an improved Multi-Population Genetic Algorithm (MPGA) for the horizontal placement problem. The objective of [20] was to maximize the sum-rate of users while considering the constraints of LoS communications and fairness in serving a required data rate to users. The authors of [20] also proposed an algorithm to convex the proposed non-convex mathematical model.

In [21], the authors broke the 3D UAV placement optimization problem down into three sub-problems. First, they solved a 2D UAV positioning using dynamic K-means clustering. Then, using game theory, they found the most efficient altitude for UAVs under 2D positions. Finally, they solved the problem of associating users to UAVs. In [22], a macro BS and several DBSs were considered. First, an algorithm was proposed to find efficient 3D locations of DBSs, associate users to BS, and allocate bandwidth for access and DBS backhaul. Next, DBS locations were updated using a heuristic PSO algorithm for more efficiency. In [23], the authors proposed a 3D deployment scheme for minimizing the total number of UAVs to cover all users with different QoS. To do so, first, they found the relationship between the UAV altitude and the coverage. Then, they proposed an algorithm that considered both altitude and horizontal location. In [4], the minimum number of UAVs and their optimal 3D locations to cover users was calculated using a heuristic algorithm. The UAV thus would achieve its coverage range by changing its altitude according to the density of users, and in order to reduce interference with other antennas, reducing its altitude in denser areas. It serves a lower population density with a higher altitude.

The authors of [24] considered an uneven terrain for 3D UAV deployment. They formulated an optimal coverage model and optimal connectivity model which are NP-hard. To tackle this problem, they designed a meta-heuristic PSO algorithm to achieve a cost-effective solution. The results did not discuss the average covered data rate, but they compared the path loss and number of UAVs in different density scenarios. In [25] a framework for drone-BSs network planning and latency-minimal cell association for drone-UEs is proposed. On the network planning side, a method based on the truncated octahedron shapes is proposed to ensure full coverage in a given space with the minimum number of DBSs. Also, an optimal 3D cell association scheme is proposed for drone-UEs latency. To do so first, the spatial distribution of drone-UEs is estimated, then due to this distribution and the locations of DBSs, the 3D association for drone-UEs is derived considering latency minimization using optimal transport theory. Authors of [26] proposed a mathematical model for joint optimization of DBS placement and IoT users' assignment in an IoT network scenario. The objective of the optimization is to maximize the connectivity of the users by utilizing the minimum number of DBSs, satisfying network constraints such as path loss. The proposed optimization problem was NP-hard, and the optimal solution has an exponential complexity so the authors proposed a linearization scheme and a low complexity algorithm to solve the problem in polynomial time. Therefore, the results are close to the optimal solution.

To date, most of the literature that has modeled UAV positioning has considered only a fairly limited number of constraints. This has meant findings have had limited usefulness and generality. Moreover, existing studies have solved optimization models with heuristic or meta heuristic algorithms that have directly impacted the precision of the results, and this is time consuming.

In this paper, we consider a sporting event in a rural area. We aim to cover IoT sensors in the field and people who participate in or attend the event. We aim to serve their required data rate using a 5G cellular network. A potential type of UAV that can be considered in this scenario is the DJI S900, which the maximum altitude of its flight is 300 meters based on its characteristics [27]. In these scenarios, it is not only the coverage of users that is important; we also need to use as few UAVs as possible. To cover the users most efficiently, we must use the least number of UAVs possible and deploy them at the most effective positions and altitudes. We also consider orthogonal frequency reuse to avoid interference between UAVs in the network. To do this, we propose a mathematical model for the optimal positioning of UAVs as

aerial BSs to cover 5G users. Our proposed model minimizes the number of UAVs required to cover at least a target percentage of users while providing the required data rate. Our model also selects the most effective positions to minimize the aggregate path loss of users from UAVs. To find the optimal positions of UAVs, the model needs some candidate points from which to choose. As users gather in specific places during the event, we can group them into the clusters. The center of each cluster will be a candidate point for the mathematical model to deploy the UAVs. In what follows, we compare the performance of six different groups of candidate points to find the best one. In this paper, we consider coverage and data rate constraints together, unlike most previous works. The proposed mathematical model determines the UAV altitudes, and we solve the problem with an exact method. To sum up, the main contributions of this work are as follows:

- We provide a method to determine efficient values for Z, X, and Y coordinates (all three at the same time) using a proposed mathematical model.
- We propose a linearization method to arrive at an exact solution with proposed 3D positioning.
- We present two discrete and continuous methods to make decision on the efficient value of altitude regarding the application we are supposed to use.
- We introduce a novel clustering method to divide users in different cluster regarding the inherent characteristics of the UAV positioning problem.

The rest of this paper is organized as follows. In Section II the system model of the problem is presented. Section III focuses on the problem formulation. In Section IV, the linearization of the proposed mathematical model is discussed. Section V discusses how to find candidate points, and the proposed candidate point set method is discussed. In Section VI, the 3D expansion of P-median is introduced, and in Section VII the efficiency of the proposed method is compared with other methods in terms of the number of UAVs, users data rate and simulation time.

II. System Model

We consider a wireless system with a set of users who temporarily gathered in a free space environment to watch a sporting event. Deploying a ground BS for short-term scenarios is not affordable, so using a UAV BS solution would be the most economically viable option. Due to UAV backhaul limitations, a UAV can serve a limited number of users. Assuming the positions of the users are known, the main question we will answer in this paper is the following:
What is the minimum number of UAVs required and their most effective positions to cover a certain percentage of users?

The altitude of a UAV is one of the determining factors of its coverage range. On one hand, if a UAV flies at a higher altitude, it will have a larger coverage range, and we will require fewer UAVs to cover users. On the other hand, increasing the altitude of the UAVs increases the path loss and consequently decreases the QoS. In this work, considering a minimum and maximum allowed altitude for UAVs $(H_{\min}, H_{\max})$, we find the minimum number of required UAVs and their optimal

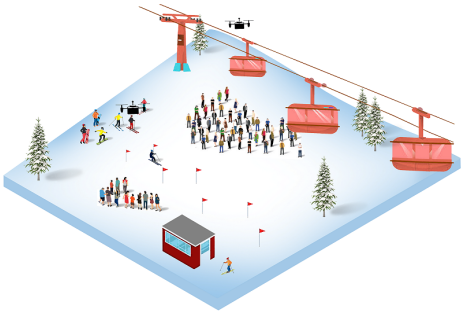


Fig. 1: A possible scenario of UAV positioning.

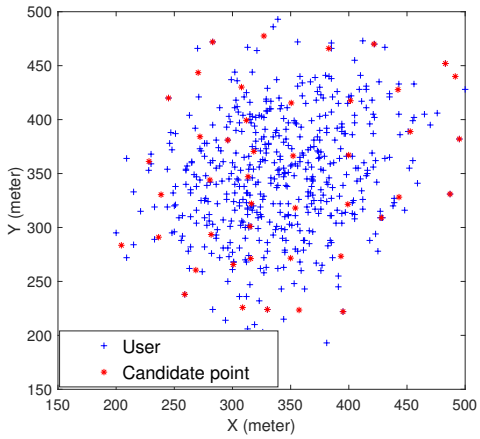
positions such that the path loss of each user does not exceed a certain bound. $H_{\min}$ can be defined as the lowest altitude of a safe flight for the UAV. Similarly, $H_{\max}$ is related to the flying capabilities of the UAV.

We find the minimum number of UAVs required in a bi-section algorithm. In each iteration of the algorithm, We find the location of a fixed number of UAVs while total path loss is minimized. If users are covered, we reduce the number of UAVs and otherwise, we increase it .

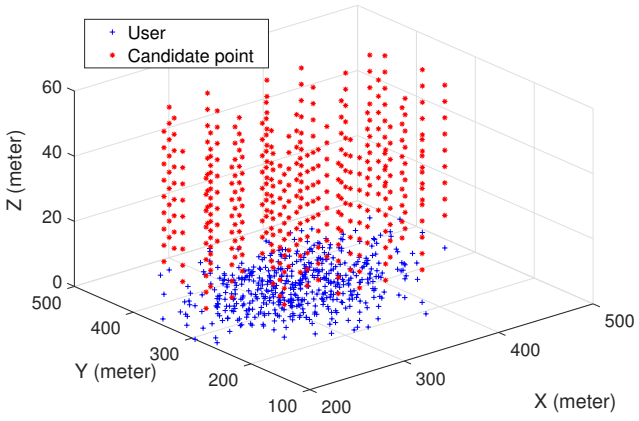
Beside path loss, we also considered data rate in the optimization model. However, the problem would become multi-objective if we considered data rate in the objective function. Therefore, we have included the data rate in the constraints to guarantee the services of the users' required data rate. We also considered that all users have equal bandwidth and transmission power. In the long-term we can omit the effect of fast fading components. In slow fading, we considered path loss as the most important attenuation component. It is worth noting that the main goal of the paper is to determine the most efficient number of UAVs and their 3D positions regarding the conditions of the problem. As we know the number of DBSs in each iteration, we find the best positions for DBSs in case of minimizing path loss (like [28]) and satisfying data rate and other constraints.

We have considered free space and Line-of-Sight as two main assumptions because of two main reasons as follows: there are other studies in this field such as [18], [20], [29] and [30] with similar assumptions. Moreover, we have considered outdoor events as main target application (see Figure 1) including winter sports, marathon or soccer games, etc. In such applications having considered Line-of-Sight and free space for communications is reasonable.

We determine the 3D location of UAVs by solving the mathematical model that we present in the next section. Our proposed mathematical model determines the optimal position of P UAVs by taking a set of candidate points $(\mathcal{I})$ to deploy UAVs. The candidate points in our proposed model are potential coordinates for the projection of UAV positions on the ground which are chosen among a continuous 2D space. In section V, we explain several methods that provide candidate points based on the position of users. The number of candidate points as one of the effective parameters in the sample size plays an important role in the number of decision variables, constraints, and consequently the solving time of a sample. In Section V, we also suggest a novel method to provide



(a) Candidate points in a 2D space.



(b) Candidate points in a 3D space.

Fig. 2: Candidate points.

candidate points intelligently based on the density and required data rate of users. Considering candidate points in a 2D space compared to a 3D space reduces the sample size. Also, in this case, decisions on the altitude of each UAV (h) are made by the proposed model, and the model finds the exact optimal altitude of UAVs in a continuous space.

III. Problem Formulation

Here, we assume that a set of points ($\mathcal{I}$) on the surface is given as the set of candidate points for the UAV's deployment. Our model selects P points among the set of candidate points to deploy UAVs, and it determines their proper altitudes to minimize the total path loss such that the UAVs will cover a specified percentage of users. Also, we formulate the problem such that the path loss of each user dose not exceed a certain bound. To achieve this objective, we need to know whether or not a candidate point is selected by the mathematical model. We define a binary decision variable m_i that is 1 if the model selects ith candidate point and takes the value 0 otherwise.

In our mathematical model, the candidate points are potential UAV shadows on the ground. So we consider the altitude of UAVs as decision variables that the model must find (h_i). If the candidate point i is selected by the model, the altitude of the UAV deployed at this point must be within the allowable altitude range ($[H_{\min}, H_{\max}]$). If the candidate point i is not selected by the model, h_i will be set to 0. To calculate the total path loss, we need to find out which user is served by which UAV. We define the binary variable x_{ij} that is 1 if the user j gets the service from the UAV deployed at candidate point i and becomes 0 otherwise. Also, the continuous variable k_{ij} indicates the path loss of user j if the user is served by the UAV i. Since path loss is directly related to the altitude of the UAV, we have to use an explicit path-loss formula in the mathematical model. The advantage of this formulation includes a significant reduction in the number of candidate points and the identification of exact optimal altitudes for UAVs. But the use of an explicit path-loss formula leads to a nonlinear model. We finally propose a linear model by adding some constraints. Known parameters and decision variables of our model are presented in Tables I and II, respectively.

We formulate the problem as follows, the objective function (1a) is defined to minimize the total path loss. Constraint (1b) states that each user can only get service from one UAV. Constraint (1c) states that user j can only get service from candidate point i, if point i is selected as one of the UAV deployment locations. Constraint (1d) guarantees at least α percent coverage of users. Constraint (1e) allows each UAV to serve as high a data rate as it can. Constraint (1f) states that we must select P point from the candidate points which is equal to available UAVs. Constraints (1g) and (1h) state

TABLE I: Using parameters.

Parameters	Description
f_c	Carrier frequency
C	Speed of light
$\mathcal{I}$	Set of candidate points
$\mathcal{J}$	Set of users
β	UAV data rate
U	Number of users
α	Minimum percentage of requested coverage
$H_{\min}$	Minimum allowed altitude
$H_{\max}$	Maximum allowed altitude
θ	UAV coverage angle
P	Number of UAVs to be deployed
PL_{max}	Maximum allowed path loss in the network
M	A big number
D_j	Data rate required for user j
d_{ij}	Distance between user j and candidate point i

TABLE II: Decision variables.

Decision variable	Description
x_{ij}	1, if user j is served by candidate point i, and 0, otherwise.
m_i	1, if candidate point i is selected for UAV deploying, and 0, otherwise.
h_i	The altitude of UAV is deployed at the candidate point i.
k_{ij}	The path loss between user j and candidate point i, if user j is served by candidate point i, and 0, otherwise.
t_{ij}	Auxiliary decision variable.

that if a candidate point is selected as a UAV position on the ground, this UAV must fly within the permissible range. Also, the UAV altitude at this point will be zero if and only if the candidate point i is not selected by the model. In constraint (1i), we want to prevent the assignment of users who are not within a UAV coverage area to that UAV.

$$\min \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} k_{ij} \tag{1a}$$

s.t

$$\sum_{i\in\mathcal{I}} x_{ij} \le 1, \quad \forall j \in \mathcal{J}, \tag{1b}$$

$$x_{ij} \le m_i, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{1c}$$

$$\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} x_{ij} \ge \alpha \times U, \tag{1d}$$

$$\sum_{j\in\mathcal{J}} D_j \times x_{ij} \le \beta \times m_i, \quad \forall i \in \mathcal{I}, \tag{1e}$$

$$\sum_{i\in\mathcal{I}} m_i = P, \tag{1f}$$

$$h_i \le H_{\max} \times m_i, \quad \forall i \in \mathcal{I}, \tag{1g}$$

$$h_i \ge H_{\min} \times m_i, \quad \forall i \in \mathcal{I}, \tag{1h}$$

$$\cot(\theta) \times x_{ij} \le \frac{h_i}{d_{ij}}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{1i}$$

$$(4\pi\frac{f_c}{C})^2(d_{ij}^2 + h_i^2) \le \mathsf{PL}_{\max} + (1 - x_{ij})M, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{1j}$$

$$k_{ij} \le M x_{ij}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{1k}$$

$$k_{ij} \ge (4\pi\frac{f_c}{C})^2(d_{ij}^2 + h_i^2)x_{ij}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{1l}$$

Lemma 1. *Consider a UAV at candidate point i and the altitude $h_i \in [H_{\min}, H_{\max}]$. The user j is within the UAV coverage area if and only if $\cot(\theta) < \frac{h_i}{d_{ij}}$ where θ is the UAV coverage angle.*

Proof. As we can see in Figure 3, $\cot(\theta) = \frac{h_i}{R_{\text{UAV}}}$ where R_{UAV} is the UAV coverage radius. So we have the following:

User j is within the UAV coverage area

$$\iff d_{ij} < R_{\text{UAV}}$$
$$\iff \frac{h_i}{R_{\text{UAV}}} < \frac{h_i}{d_{ij}}$$
$$\iff \cot(\theta) < \frac{h_i}{d_{ij}}.$$

□

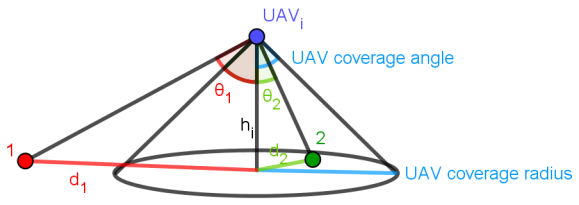


Fig. 3: UAV coverage.

In constraint (1i), if $\cot(\theta) \ge \frac{h_i}{d_{ij}}$, x_{ij} must take the value 0 because user j is not within the UAV coverage area. Assuming a UAV at candidate point i and the altitude h_i, in the next constraint, we want the user j not to be assigned to the UAV i if the path loss exceeds $\mathsf{PL}_{\max}$. We rewrite this conditional statement with problem symbols as

$$(4\pi\frac{f_c}{C})^2(d_{ij}^2 + h_i^2) \ge \mathsf{PL}_{\max} \rightarrow x_{ij} = 0. \tag{2}$$

Constraint (1j) as a valid constraint in the optimization models satisfies the above requirements. Constraint (1k) sets the value of k_{ij} to 0 if the user j is not assigned to UAV i. Constraint (1l) states that if user j is served by the UAV whose shadow is on candidate point i, the value of k_{ij} must be at least equal to the path loss of this user from the UAV. Since the problem is to minimize the sum of k_{ij}s, the value of k_{ij} will not be greater than the right-hand side of the inequality.

Constraints (1j) and (1l) are two nonlinear constraints in terms of decision variables in the proposed model. Below, we linearize these two constraints to achieve a linear model. There are exact methods like Branch and Bound (B & B) in theoretical optimization and powerful tools in terms of implementation like CPLEX solver for mixed binary linear optimization problems. We will use the CPLEX solver to solve our model, which exploits Branch and Bound algorithms to solve mixed-integer linear optimization problems.

IV. Linearization of Nonlinear Constraints

Consider $F_{ij}(h_i) = (4\pi\frac{f_c}{C})^2(d_{ij}^2 + h_i^2)$ is the path-loss function in terms of UAV altitude. The conditional statement (2) can be rewritten as follows:

$$x_{ij} = \begin{cases} 0, & \text{if } F_{ij}(h_i) \ge \mathsf{PL}_{\max}, \\ 0 \quad \text{or} \quad 1, & \text{otherwise.} \end{cases} \tag{3}$$

We obtain a linear conditional statement in terms of h_i by replacing $F_{ij}(h_i)$ in (3) with its linear approximation achieved from Taylor expansion around some h_0.

$$F_{ij}(h_i) = F_{ij}(h_i - h_0 + h_0) \approx F_{ij}(h_0) + F_{ij}^{'}(h_0)(h_i - h_0)$$
$$= (4\pi\frac{f_c}{C})^2(d_{ij}^2 + h_0^2) + (4\pi\frac{f_c}{C})^2 \times 2 \times h_0 \times (h_i - h_0).$$

Now we have

$$x_{ij} = \begin{cases} 0, & \text{if } (4\pi\frac{f_c}{C})^2(d_{ij}^2 + h_0^2) + \\ & \quad 2h_0(4\pi\frac{f_c}{C})^2 \times (h_i - h_0) \ge \mathsf{PL}_{\max}, \\ 0 \quad \text{or} \quad 1, & \text{otherwise.} \end{cases} \tag{4}$$

By simplifying the conditional expression we have

$$x_{ij} = \begin{cases} 0, & \text{if } h_i \geq \frac{\mathsf{PL}_{\max} - (4\pi\frac{f_c}{C})^2(d_{ij}^2 - h_0^2)}{2h_0(4\pi\frac{f_c}{C})^2}, \\ 0 \quad \text{or} \quad 1, & \text{otherwise.} \end{cases}$$

By defining $a_{ij} = \frac{\mathsf{PL}_{\max} - (4\pi\frac{f_c}{C})^2(d_{ij}^2 - h_0^2)}{(4\pi\frac{f_c}{C})^2 \times 2 \times h_0}$, the conditional expression will be simplified as follows

$$x_{ij} = \begin{cases} 0, & \text{if } h_i \geq a_{ij}, \\ 0 \quad \text{or} \quad 1, & \text{otherwise.} \end{cases} \tag{5}$$

To form (5) as a valid constraint in mathematical programming, we represent the following expression:

$$x_{ij} \leq \frac{M - h_i}{M - a_{ij} + \frac{1}{2}}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{6}$$

where M is a big number. Since all components in the definition of a_{ij} are known, constraint (6) is linear in terms of decision variables. Also, if $h_i \geq a_{ij}$, the right-hand side of the inequality in (6) is less than 1, and since x_{ij} is a binary decision variable, it will take 0 value, and if $h_i < a_{ij}$, $\frac{M - h_i}{M - a_{ij} + \frac{1}{2}}$ will take a value greater than 1 then x_{ij} can be 0 or 1. So we replace the constraint (1j) with (6).
As mentioned above, we used the first-order Taylor expansion of F_{ij} around h_0 to linearize the constraints. In the following theorem, we find the best h_0 so that the total approximate error is minimized.

Theorem 1. *Suppose the linear approximation of $F_{ij}(h)$ defined in Section IV at h_0 is $\bar{F}_{ij}(h)$. Also assume that the domain of $F_{ij}(h)$ is $[H_{\min}, H_{\max}]$. $h_0 = \frac{H_{\min} + H_{\max}}{2}$ minimizes the total approximation error.*

Proof. The Taylor expansion of $F_{ij}(h)$ around arbitrary point h_0 is as follows:

$$F_{ij}(h) = F_{ij}(h_0) + F_{ij}^{'}(h_0)(h - h_0) + F_{ij}^{''}(h_0)\frac{(h - h_0)^2}{2} + F_{ij}^{'''}(h_0)\frac{(h - h_0)^3}{6} + \dots$$

As $\forall h, F_{ij}^{'''}(h)$ and higher order derivatives of F_{ij} are equal to zero, we have

$$F_{ij}(h) = F_{ij}(h_0) + F_{ij}^{'}(h_0)(h - h_0) + F_{ij}^{''}(h_0)\frac{(h - h_0)^2}{2}.$$

Since $\bar{F}_{ij}(h)$ is the linear approximation of $F_{ij}(h)$, we have

$$F_{ij}(h) = \bar{F}_{ij}(h) + F_{ij}^{''}(h_0)\frac{(h - h_0)^2}{2}.$$

Therefore, the approximation error in h is

$$F_{ij}(h) - \bar{F}_{ij}(h) = F_{ij}^{''}(h_0)\frac{(h - h_0)^2}{2},$$

where $F_{ij}^{''}(h_0) = 2(4\pi\frac{f_c}{C})^2$. By defining $A = (4\pi\frac{f_c}{C})^2$ the total approximation error is equal to

$$E(h_0) = \int_{H_{\min}}^{H_{\max}} A(h - h_0)^2 = \frac{A(h - h_0)^3}{3}\Big|_{H_{\min}}^{H_{\max}} = \left[\frac{A(H_{\max} - h_0)^3}{3}\right] - \left[\frac{A(H_{\min} - h_0)^3}{3}\right] = \frac{A}{3}(H_{\max}^3 - H_{\min}^3 - 3h_0(H_{\max}^2 - H_{\min}^2) + 3h_0^2(H_{\max} - H_{\min})).$$

The minimizer of $E(h_0)$ satisfies $E^{'}(h_0) = 0$. So we have

$$-3(H_{\max}^2 - H_{\min}^2) + 6h_0(H_{\max} - H_{\min}) = 0 \implies h_0 = \frac{H_{\min} + H_{\max}}{2}.$$

□

Constraint (1l) also contains the multiplication of x_{ij} and h_i^2, which is a nonlinear decision expression. To linearize this constraint, we replace the path-loss formula with its linear approximation:

$$k_{ij} \geq [(4\pi\frac{f_c}{C})^2 d_{ij}^2 + h_0^2 + (4\pi\frac{f_c}{C})^2 \times 2 \times h_0 \times (h_i - h_0)]x_{ij}.$$

By doing so, the nonlinear part is reduced to the multiplication of x_{ij} and h_i. We propose a linear constraint by introducing the decision variable $t_{ij} = x_{ij}h_i$ and placing it in the above constraint:

$$k_{ij} \geq [(4\pi\frac{f_c}{C})^2 d_{ij}^2 - h_0^2]x_{ij} + (4\pi\frac{f_c}{C})^2 \times 2 \times h_0 \times t_{ij}. \tag{7}$$

But now there are a few things to consider:

- t_{ij} must be zero if x_{ij} or h_i is equal to zero. Constraints (8) and (9) satisfy this requirement:

$$t_{ij} \leq h_i, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{8}$$

$$t_{ij} \leq H_{\max} \times x_{ij}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{9}$$

- t_{ij} must be equal to h_i if x_{ij} becomes 1. Constraints (8) and (10) satisfy this requirement:

$$t_{ij} \geq h_i - (1 - x_{ij})H_{\max}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{10}$$

Note that if $x_{ij} = 1$, constraint (9) is a redundant constraint.

By replacing constraint (1l) with inequality (7) and adding equations (8), (9), and (10) to the mathematical model as

constraints, we achieve a linear mathematical model. The aggregated proposed linear model is as follows:

$$\min \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} k_{ij} \tag{11a}$$

s.t

$$\sum_{i\in\mathcal{I}} x_{ij} \le 1, \quad \forall j \in \mathcal{J}, \tag{11b}$$

$$x_{ij} \le m_i, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11c}$$

$$\sum_{i\in I\mathcal{I}}\sum_{j\in\mathcal{J}} x_{ij} \ge \alpha \times U, \tag{11d}$$

$$\sum_{j\in\mathcal{J}} D_j \times x_{ij} \le \beta \times m_i, \quad \forall i \in \mathcal{I}, \tag{11e}$$

$$\sum_{i\in\mathcal{I}} m_i = P, \tag{11f}$$

$$h_i \le H_{\max} \times m_i, \quad \forall i \in \mathcal{I}, \tag{11g}$$

$$h_i \ge H_{\min} \times m_i, \quad \forall i \in \mathcal{I}, \tag{11h}$$

$$\cot(\theta) \times x_{ij} \le \frac{h_i}{d_{ij}}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11i}$$

$$x_{ij} \le \frac{M - h_i}{M - a_{ij} + \frac{1}{2}}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11j}$$

$$k_{ij} \le M \times x_{ij}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11k}$$

$$k_{ij} \ge [(4\pi\frac{f_c}{C})^2 d_{ij}^2 - h_0^2]x_{ij} + (4\pi\frac{f_c}{C})^2 \times 2 \times h_0 \times t_{ij}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11l}$$

$$t_{ij} \le h_i, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11m}$$

$$t_{ij} \le H_{\max} \times x_{ij}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, \tag{11n}$$

$$t_{ij} \ge h_i - (1 - x_{ij})H_{\max}, \quad \forall i \in \mathcal{I}, j \in \mathcal{J}. \tag{11o}$$

After modeling the problem, we need to answer the following questions:
1) What is the best set of candidate points for the model? As seen in describing the model parameters in Table I, the candidate points set ($\mathcal{I}$) must be given to the model.
2) What is the optimal appropriate value for the number of UAV (P)?

V. Finding a Set of Candidate Points

Our proposed model requires a set of candidate points. For this, instead of considering all the points of the space as potential positions for UAV deployment, we provide a finite number of points as the set of candidate points. By doing so, the selection of P points among uncountable points is converted to a mixed-integer optimization that seeks to select P points from a large but finite number of candidate points. Since the presented model is a mixed-integer optimization and consequently an NP-hard problem, the solving time increases exponentially by increasing the instance dimension. The number of candidate points as a problem input has a meaningful effect on the number of constraints and variables in the model and exact solving time. We try to reduce the problem size by intelligently providing candidate points and then obtaining the exact solution of the reduced problem quickly by CPLEX solver. Here we suggest six methods to make discrete 2D space and introduce candidate points. In the next Section, we will generalize these methods in 3D space and provide candidate points for the model presented in [1] to solve the problem of UAV deployment in 3D space.

A. On user strategy

This method considers each user as a candidate for a UAV shadow. Although the candidate points introduced by this method provide good coverage for users, the high number of candidate points is the weakness of this method in crowded scenarios.

B. Clustering methods

Clustering is a machine learning technique that involves the grouping of data points. Given a set of data points, we can use a clustering algorithm to associate each data point with a specific group. In theory, data points that are in the same group should have similar features, while data points in different groups should have dissimilar features. Here, we cluster users by considering the coordinates of each user as its feature. So users who are close to each other are clustered in the same group. After clustering the users, we offer the centroids as the set of candidate points. By doing so, we expect that after solving the mathematical model with the candidate points provided, the nearby users will get service from the same UAV.

Most clustering methods require some parameters to cluster the data. For example, in K-means and K-medoids, parameter K is the number of clusters that must be specified [31]. Some clustering methods, such as DBSCAN (Density-Based Spatial Clustering of Applications with Noise) and Mean-shift, group the data based on density and do not need the number of clusters. Instead, in the DBSCAN method, the minimum number of neighbors and a distance threshold must be predetermined [32]. In the Mean-shift method, output depends on window size that must be specified before clustering [33]. Finding the proper parameters for each method in different modes of users distributions is challenging and time-consuming.

There is a little sensitivity to outliers in density-based methods in the sense that the outliers have little effect on clustering. While in K-means and K-medoids methods, an outlier can significantly move the cluster center. Also, due to the grouping of data based on density in these methods, there is no predefined shape for clusters. Although this feature is mentioned as one of the strengths of density-based methods, a strip-shaped cluster, for example, would not be helpful in the present application.

We compare four well-known clustering methods as strategies to provide candidate points: K-means, K-medoids, DBSCAN, and Mean-shift.

C. MergeCells

Here, we propose the MergeCells method that does not need to know the number of clusters and provides groups of users in the shape of a square. The average coordinate of users within each square will be considered as a candidate point. Since we

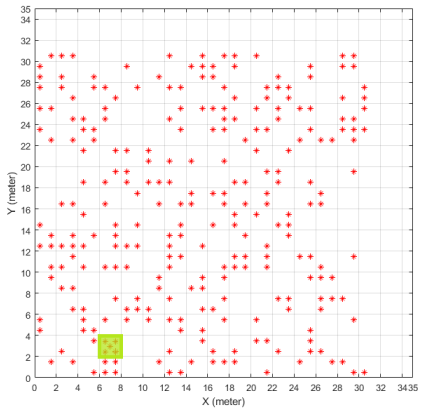


Fig. 4: Finding the densest cell.

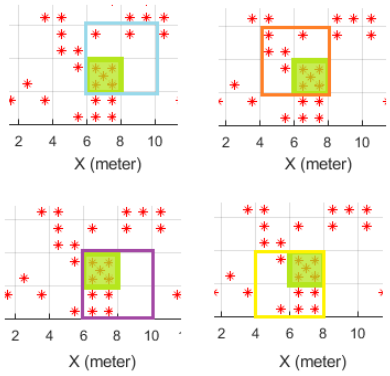


Fig. 5: States for cell expansion in the first step.

will deploy at most one UAV at each candidate point, in this method, the clusters will be at most as large as the coverage area of a UAV. In the MergeCells method, we attempt to group users in such a way that the number of candidate points is not too many. At first, we mesh the surface with small cells and consider the average coordinates of the users within each cell as candidate points. Then, in an iterative process, we try to reduce the number of candidate points by merging the cells as much as possible. In the following, we will explain this method with the help of an example. After meshing the space with small cells, we count the number of users within each cell and select the densest one. If the data rate required by the users of this cell is equal or in excess of the UAV data rate, we would place the number of candidate points needed into the cell using a uniform distribution and mark it as a non-expandable cell. Otherwise, we consider four states for cell expansion. In each case, the cell enlarges in the direction of one corner. Four states of expansion at the first iteration for the selected cell in Figure 4 is as Figure 5.

For each state, the number of users within the expanded cell is counted. If the number of users exceeds the average number of users an UAV can cover, the expansion will be labeled as an infeasible state. Among the feasible states, we select the densest and expand the initial cell to it (for this example the yellow cell). Then, the expansion will be continued for the selected state. The expansion states of the yellow cell are shown in Figure 6.

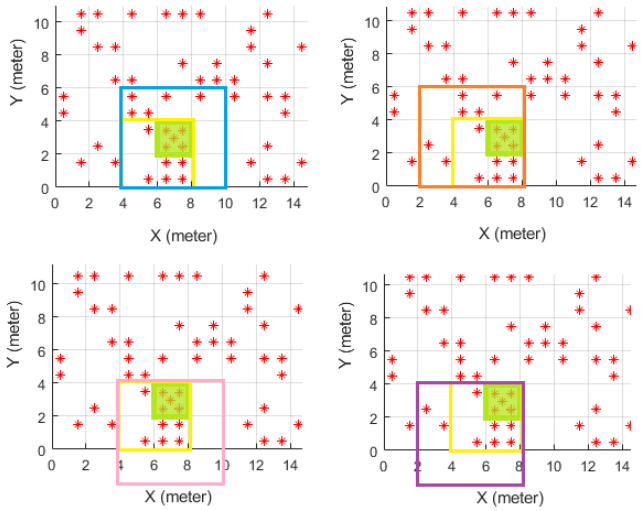


Fig. 6: States for cell expansion in step 2.

We keep expanding the cell until either the cell has no feasible expansion or the side length of the cell exceeds a predetermined parameter R. In such a situation, we mark the cell as non-expandable, stop its expansion and replace the candidate points within this cell with the average coordinates of its users. Parameter R is the side length of the largest enclosed square within the coverage area of a UAV. Using this parameter, we prevent the cell from becoming too large and crossing the boundaries of a UAV's coverage area. Since the coverage area of a UAV depends on its altitude, and this algorithm must be run before solving the model and determining the altitude of the UAVs, we consider some values for R, and in Section VII, we compare the numerical results of each value.

After stopping the expansion of one cell, the next cell will be selected among the expandable cells. We will continue the process as long as no expandable cells remain. Since it is possible in reality to have an overlap in UAV coverage range, we also allow cells to overlap in this meshing. To avoid the unreasonable cessation of cell expansion, users who are in the coverage range of two cells are not included in the count of covered users in the second cell. The details of this method are described in Algorithm 1.

The finer the initial mesh in this method, the higher the accuracy of selecting candidate points. But in practice, if the accuracy of the candidate points is higher than the precision of UAV's controllability, deployment in the obtained points is difficult to achieve.

Selecting the densest expandable cell in each iteration of the "while" loop and selecting the densest state for cell expansion indicates that Algorithm 1 is greedy. Although we cannot guarantee that our greedy algorithm will find the best candidate points, the numerical results show that it performs better than the other mentioned methods.

Algorithm 1 MergeCells

1- for all $j \in$ Users
2- Mark j as uncovered;
3- end for
4- Mesh the space with small cells;
5- $C \longleftarrow$ The set of cells;
6- for all $c \in C$
7- Mark c as an expandable cell;
8- $V_c \longleftarrow$ The number of users within cell c;
9- end for
10- $CandidatePoints \longleftarrow \{\}$;
11- while (There is an expandable cell)
12- $CurrentCell \longleftarrow argmax_c\{V_c | c$ is expandable$\}$;
13- $J \longleftarrow$ The set of uncovered users within $CurrentCell$;
14- if($|J| >$ average number of users an UAV can cover)
15- $K \longleftarrow \frac{|J| \times \text{Mean data rate of users}}{\text{UAV data rate}}$;
16- $RC \longleftarrow \{r_1, r_2, \ldots, r_K | r_i \quad i = 1, 2, \ldots, K$ is random point
17- within $CurrentCell\}$;
18- $CandidatePoints \longleftarrow CandidatePoints \cup RC$;
19- for all $j \in J$
20- mark j as covered;
21- end for
22- else
23- while(The side of $CurrentCell < R$)
24- Form the cell expansion states E_1, E_2, E_3, E_4;
25- for $m = 1, 2, 3, 4$
26- $J_m \longleftarrow$ The set of users within E_m;
27- $DRE_m \longleftarrow |J_m| \times$ Mean data rate of users;
28- if ($DRE_m >$UAV bandwidth)
29- mark E_m as an infeasible expansion;
30- end if
31- end for
32- $index \longleftarrow argmax_{m \in 1,\ldots,4}\{|J_m| \quad |E_m$ is a feasible expansion$\}$;
33- $NewCell = E_{index}$;
34- if($NewCell == \varnothing$)
35- Mark $CurrentCell$ as a non-expandable cell;
36- for all c within $CurrentCell$
37- Mark c as a non-expandable cell;
38- end for
39- $CandidatePoint \longleftarrow CandidatePoint \cup \{$The average
40- coordinates of the users within $CurrentCell\}$;
41- else
42- $CurrentCell \longleftarrow NewCell$;
43- for all $j \in J_{index}$
44- mark j as covered;
45- end for
46- end if
47- end while
48- end if
49- end while

Theorem 2. *The complexity of the MergeCells algorithm for an $m \times n$ rectangle with U users is of the order $O(U) + O((R-1) \times m \times n)$.*

Proof. In this algorithm, the rectangle is divided into $N = m \times n$ small cells. The complexity of marking and assigning users to small cells is of the order $O(U)$. In the second part of the algorithm, the cell expansion is performed for each expandable cell. Therefore cell expansion occurs $O(N)$ times. Adding sufficient candidate points for dense cells is of order $O(1)$. Non-condensed cells expand as long as they do not violate the data rate limit or UAV coverage radius. In each iteration of cell expansion, one unit is added to the side of the cell. Since the UAV coverage radius is considered during the expansion, the cell is enlarged at most $R-1$ times. So the complexity of the MergeCells algorithm is equal to $O(U) + O((R-1) \times N) = O(U) + O((R-1) \times m \times n)$. □

VI. 3D P-MEDIAN

To compare the model presented in this paper with our previous work [7], we need to generalize the method presented in [7] to 3D positioning. For this purpose, we must provide the set of candidate points for the model presented in [7] from 3D space. We use a 3D expansion of the candidate points obtained from one of the methods mentioned in Section V as the set of candidate points.

1) Obtain a set of points on the 2D space from one of the methods mentioned in the previous Section.
2) For each (x, y) in this set, add the following points to the set of candidate points:

$$\{(x, y, z) | z \in \mathbb{Z}, H_{\min} \leq z \leq H_{\max}\},$$

where $H_{\min}$ and $H_{\max}$ are the minimum and the maximum allowable UAV altitudes, respectively. Note that by doing so, the altitude of the UAVs will be selected from a discrete space. So the 3D placement of the UAVs using the model presented in this paper is more accurate than the model presented in [7] because we consider the altitudes of UAVs as continuous decision variables.

As mentioned at the beginning of Section II, we search for the smallest P in a bi-section algorithm. This algorithm requires an upper bound and a lower bound for P. The upper

Algorithm 2 Solving UAV placement problem

1- Choose one of the methodologies to determine candidate points for
2- deploying UAVs
3- Find $P_{\max}$
4- Calculate $P_{\min}$
5- while ($P_{\max} - P_{\min} \geq 1$)
6- $p \leftarrow \lfloor \frac{P_{\max} + P_{\min}}{2} \rfloor$
7- Solve mathematical model using Cplex with $P = p$
8- if (mathematical model is feasible)
9- $P_{\max} \leftarrow p$;
10- else
11- $P_{\min} \leftarrow p$;
12- end if
13- end while

and lower bounds are calculated in the manner described in [7]. We solve the main problem of finding the minimum number of UAVs as well as their optimal position and altitude of each UAV to cover α percent of users by using Algorithm 2.

VII. Numerical Results

In this Section, we first introduce the test system and simulation parameters. Then, we discuss and compare the results for selecting proposed 3D optimization placement models for UAVs using the six approaches presented on the basis of candidate points.

A. Test system

In our simulations, we consider only one ownership for the network provider and the centralized decision making. We also consider a 500×500 meter area with scenarios including 300, 500, and 700 users in three different distributions from dense to scattered using Poisson Point Process. The λ parameter is interpreted as the average number of points per unit and it is also called the mean density or mean rate [34][35]. In this paper, we used the "poissrnd" Matlab function to generate random numbers from the Poisson distribution with the mean parameter λ. This function gets another argument which is the number of instances that we want to create. Hence, we created, for example 700 users with λ parameter in Poisson distribution with separate x and y values. Therefore, the unit of λ is users per km. The λ of the dense and median scenarios are equal to 20, and $2,000$, respectively. The distribution of the scattered scenario is uniform distribution. In these optimization problems, the goal is to cover at least α percent of users according to the quality constraints. We consider three different α values, specifically 70, 80, and 90%. Additionally, the average data rate required by each user is 2 Mbps, and the distribution of the user data rate is a uniform distribution. The backhaul data rate of each UAV is assumed to be 300 Mbps, which is the limitation of the sum of the uplink for covered users, and their flying altitude is between 10 to 50 meters. We also consider an elevation angle of 45 degrees, so the coverage radius would be the same as the altitude. These assumptions are related to the DJI UAV specifications and power consumption. These parameter values for each scenario are shown in Table III.

Based on the implementation of the bi-section algorithm to find the optimal P, we get the lowest necessary UAVs with at most $\log_2^{(\mathrm{P_{max}}-\mathrm{P_{min}})}$ times execution of the optimization model. For time consumption, we ran algorithms and CPLEX

TABLE III: Test parameters for evaluating the problem model.

Parameters	Description
$Region$	500×500 m
U	300, 400, 500, 600 , 700
α	70%, 80%, 85%, 90%, 95%
β	300 Mbps
D_{avg}	2 Mbps
$H_{\min}$	10 m
$H_{\max}$	50 m

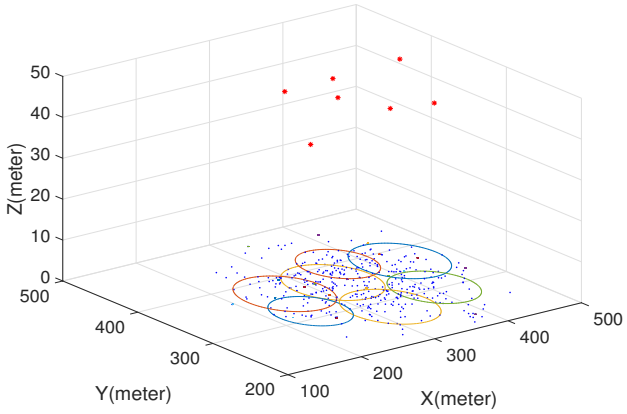


Fig. 7: Illustration of UAV positions.

Studio IDE, for solving the proposed mathematical model on a system with 12 GB RAM and 2.4 GHz Core-i5 CPU.

B. Results

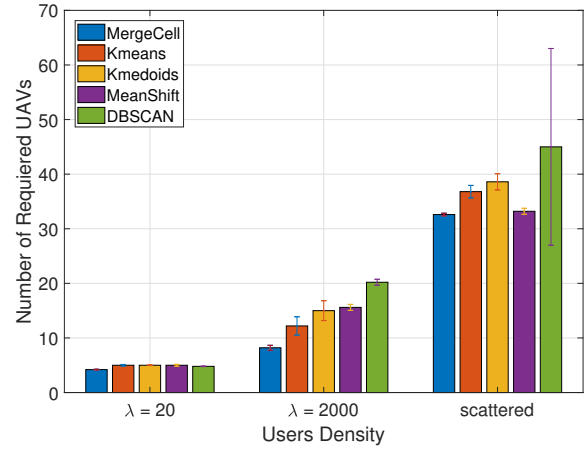
In the following, we compare these six approaches of candidate points' selection as inputs of the optimization problem with each other in each scenario. Figure 7 shows a snapshot of how UAVs are deployed after solving the optimization problem that derives candidate points with the proposed MergeCells method.

To have informative and generalizable results in the rest of our simulations we present results that are the average of about 50 runs for each scenario and method. We consider three different parameters for each clustering method, including the MergeCells method, to find the most suitable set of candidate points in each method. Parameters for the MergeCells method are R = 10, R = 50 and R = 100. Besides, we assume that the precision of controllability of UAVs is 1 m. So the dimension of small cells is 1 m $\times$ 1 m. For the K-means and K-medoids methods, we assume parameters based on $\mathrm{P_{max}}$. Parameters for the Mean shift clustering method are bw = 2, bw = 2.5 and bw = 3.25. For the DBSCAN clustering method, we assume $\epsilon = 50$ and $\mu = 5$ as it needs.

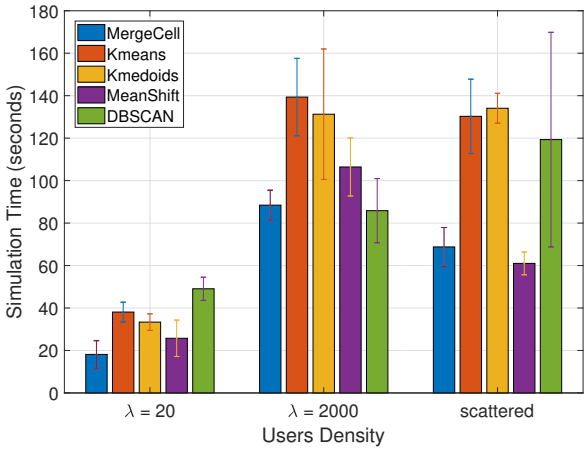
Figure 8 compares the results of the proposed optimization model solving the scenario of covering 90% of 700 users in three different densities. Figure 8a shows that in each density we need fewer UAVs using the MergeCells method compared with other methods. The same figure also shows that the candidate points found with K-means and K-medoids are not suitable for dense scenarios. Moreover, the candidate points found with the DBSCAN clustering method is not suitable for scattered scenarios. In addition to previous results, the on-user method could not manage to have a solution because the number of candidate points exceeded the required capacity and time frame to find an optimal solution.

Figure 8b compares the running time of the proposed approach using methods that determines candidate points. In dense scenarios, the proposed MergeCells method takes less time to find the optimum solution. In scenarios with $\lambda = 2,000$ although DBSCAN has a better time, the MergeCells method,

which is in the second position, has a better result in terms of the number of UAVs that is the main objective of this problem. In scattered scenarios, K-means and K-medoids methods are approximately the same in terms of simulation time, but the MergeCells method has a better solution in terms of the number of UAVs. As K-means and K-medoids have no solution in dense scenarios, no users are covered and the data rate is zero. This happens in scattered scenarios for DBSCAN clustering. As the optimization problem tries to cover at least 90% of users, the sum data rates for covered users are nearly the same.



(a) Number of required UAVs.



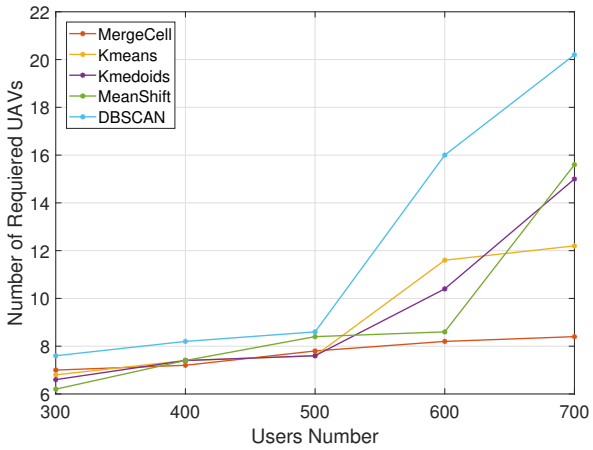
(b) Simulation time.

Fig. 8: The results of covering 90% of 700 users scenario.

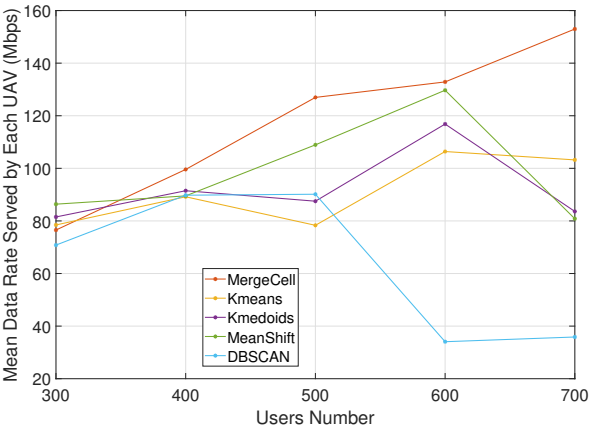
For a better overview of the results, we compare different methods in different scenarios. It is worth noting that the overall results of different scenarios in terms of the number of users are like figure 8a.

In Figure 9 the scenario of covering 90% of users with $\lambda = 2,000$ Poison density in terms of the number of required UAVs and average data rates served by UAVs is compared.

Figure 9a shows that, overall, the MergeCells method requires fewer UAVs compared to other methods. Despite in the scenario with 300 users the MergeCells method is not the best, it is as efficient as other methods with the difference less



(a) Number of required UAVs.



(b) Average date rate served by each UAV.

Fig. 9: Comparing results of methods in the scenario of covering 90% of users with $\lambda = 2,000$ density.

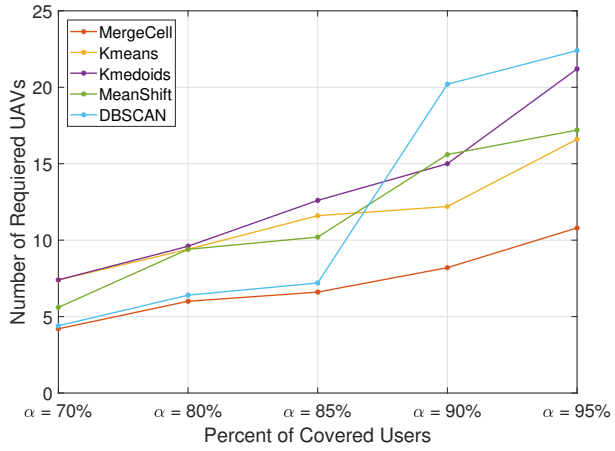
than one UAV. The on-user method is not a proper choice, especially in the scenario with 700 users.

Figure 9b illustrates the average data rate served by each UAV. In scenarios with 500 and 700 users, the MergeCells method has the best results, but in the scenarios with 300 users, the Mean-shift, K-means, and K-medoids methods have better results. However, the difference between the MergeCells and Mean-shift result, which is the best method in this scenario, is less than 10 Mbps. It is also worth noting that Mean-shift has the fourth rank in the scenario with 700 users.

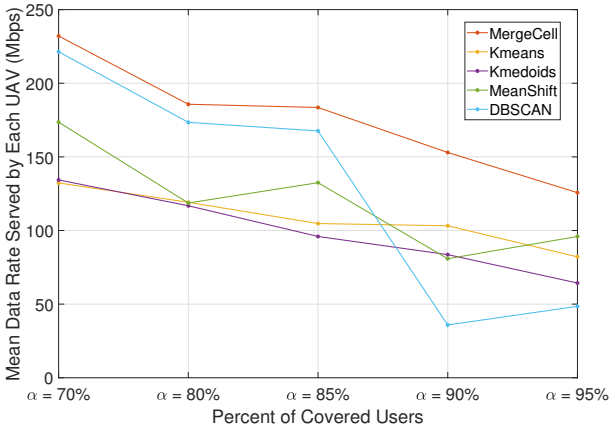
Figure 10 compares the scenario of covering 700 users $\lambda = 2,000$ density in different percentages of coverage. In Figure 10a this comparison is shown in terms of the number of required UAVs. The MergeCells method has the best results in all different alphas, whereas the on-user method has no results and K-medoids is the worst choice in $\alpha = 70\%$ and $\alpha = 80\%$ and the DBSCAN method is the worst in $\alpha = 90\%$ scenarios.

Figure 10b compares the results in term of mean data rate served by each UAV. In all scenarios, the MergeCells method has the best results. Because the difference between covering 70% and 90% of users is not more than 140 users total, the sum data rate served is equal to 280 Mbps on average, which

is less than one UAV backhaul. By the way, more than one UAV is needed if we aim to cover 10% more users. Therefore, the average data rate served by each UAV decreases while covering a greater percentage of users.



(a) Number of required UAVs.

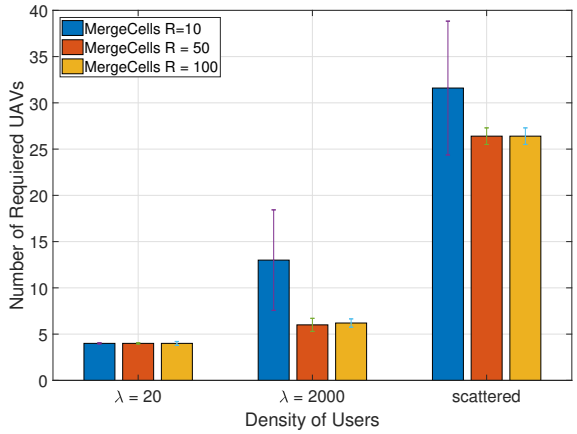


(b) Mean date rate served by each UAV.

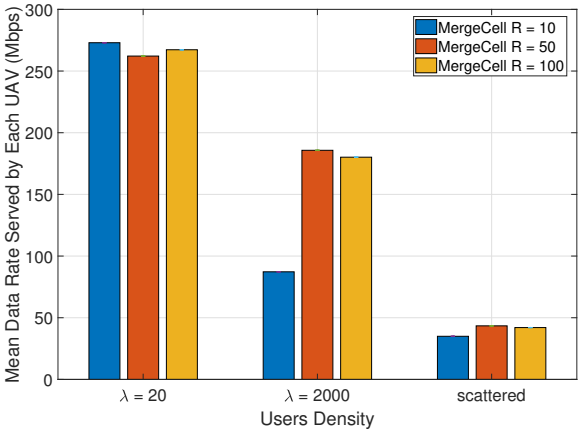
Fig. 10: Comparing results of methods in the scenario of covering 700 users with $\lambda = 2,000$ density.

Overall, the MergeCells method has better results in terms of the number of UAVs, simulation time, and mean data rate served by each UAV. We considered three different parameters for the maximum width of the cell to reach the best parameter for the MergeCells method. Figure 11a shows that both R = 50 and R = 100 have approximately the same results in terms of the number of UAVs, but the average data rate result of running the optimization problem using the MergeCells method with a parameter of R = 100 is less than R = 50 in $\lambda = 2,000$ and scattered densities as shown in Figure 11b.

As discussed in Section III, we proposed a mathematical model to solve the positioning problem. Figure 12 shows the results of the proposed model in comparison with the model of [7] with the scenario of covering 90% of 700 users. The number of UAVs required results in our proposed model being less than the 3D P-median [7] one as Figure 12a illustrated. The mean data rate served by each UAV using our proposed model is greater than the results of the 3D P-median in every



(a) Number of required UAVs



(b) Mean data rate served by each UAV

Fig. 11: Comparing different parameters of the MergeCells method in the scenario of covering 80% of 700 users.

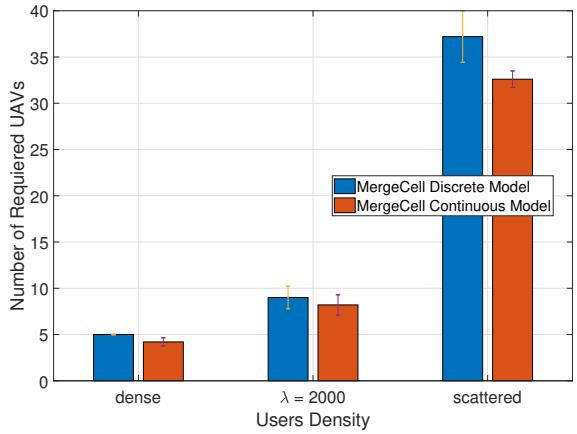
other scenario due to the total data rate of UAVs is fixed and the more number of UAVs causes less average data rate.

It should be mentioned that as shown in 12b because the proposed model itself decides the altitude of UAVs instead of searching the set of Z coordinator candidate points where the 3D P-median solution does, the running time of the proposed model is much less than the running time of the 3D P-median one.

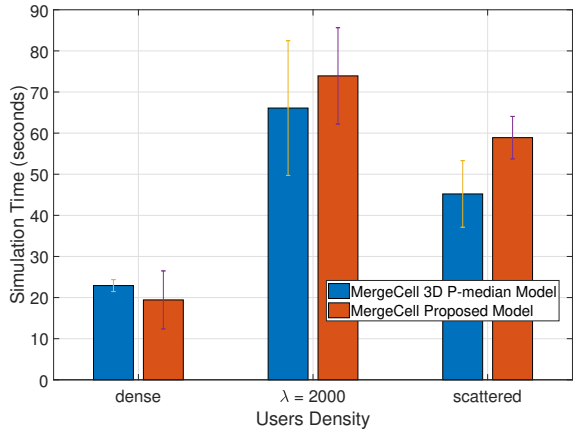
VIII. Conclusion

In this paper, we proposed a mathematical model for 3D UAV positioning to cover IoT nodes and wireless users. One of the main advantages of the proposed model is the determination of the most efficient altitude of the UAVs. To solve the model, we needed some candidate points to determine UAV positions. Therefore, we also proposed the MergeCells method to find candidate points for the proposed model. since the simulation time of our proposed model is not complex, if the position of the users changes significantly, the proposed solution must be run again. Re-running is not needed until the number of covered users by a UAV or the percentage

of covered users change. A user may have moved a lot but not yet out of the UAV coverage. As discussed in the Numerical Results Section, the results of the MergeCells method were better overall than other candidate point methods, despite the number of required UAVs, mean UAV data rate served, and the running time. Also, the results of the proposed model are significantly better than the 3D P-median.



(a) Number of required UAVs



(b) Simulation time

Fig. 12: Comparison of MergeCells in both mathematical model in the scenario of covering 90% of 700 users.

REFERENCES

[1] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Wireless communication using unmanned aerial vehicles (UAVs): Optimal transport theory for hover time optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 12, pp. 8052–8066, Sep. 2017.

[2] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2334–2360, Third quarter 2019.

[3] H. Ahmadi, K. Katzis, and M. Z. Shakir, "A novel airborne self-organising architecture for 5G+ networks," in *Proceeding of IEEE 86th Vehicular Technology Conference (VTC2017-Fall)*, pp. 1–5.

[4] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3D placement of drone base stations in wireless cellular networks," in *Proceeding of IEEE 84th Vehicular Technology Conference (VTC2016-Fall)*, pp. 1–6.

[5] 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, May 2017.

[6] E. Kalantari, M. Z. Shakir, H. Yanikomeroglu, and A. Yongacoglu, "Backhaul-aware robust 3D drone placement in 5G+ wireless networks," in *Proceeding of IEEE International Conference on Communications Workshops (ICC2017-workshops)*, pp. 109–114.

[7] M. J. Sobouti, Z. Rahimi, A. H. Mohajerzadeh, S. A. Hosseini Seno, R. Ghanbari, J. M. Marquez-Barja, and H. Ahmadi, "Efficient deployment of small cell base stations mounted on unmanned aerial vehicles for the internet of things infrastructure," *IEEE Sensors Journal*, vol. 20, no. 13, pp. 7460–7471, Feb. 2020.

[8] M. Zahedi, M. Sobouti, A. Mohajerzadeh, A. Rezaee, and S. Hosseini Seno, "Fuzzy based efficient drone base stations (DBSs) placement in the 5G cellular network," *Iranian Journal of Fuzzy Systems*, vol. 17, no. 2, pp. 29–38, Apr. 2020.

[9] 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, Jul. 2014.

[10] G. Fontanesi, A. Zhu, and H. Ahmadi, "Outage analysis for millimeter-wave fronthaul link of UAV-aided wireless networks," *IEEE Access*, vol. 8, pp. 111 693–111 706, June 2020.

[11] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Drone small cells in the clouds: Design, deployment and performance analysis," in *Proceeding of IEEE Global Communications Conference (GLOBECOM2015)*, pp. 1–6.

[12] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 38–41, Sep. 2018.

[13] 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 *Proceeding of IEEE International Conference on Communications (ICC2016)*, pp. 1–5.

[14] C. T. Cicek, H. Gultekin, B. Tavli, and H. Yanikomeroglu, "Backhaul-aware optimization of UAV base station location and bandwidth allocation for profit maximization," *IEEE Access*, vol. 8, pp. 154 573–154 588, 2020.

[15] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Optimal transport theory for power-efficient deployment of unmanned aerial vehicles," in *Proceeding of IEEE International Conference on Communications (ICC20116)*, pp. 1–6.

[16] ——, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient internet of things communications," *IEEE Transactions on Wireless Communications*, vol. 16, no. 11, pp. 7574–7589, Sep. 2017.

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

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

[19] Y. Chen, N. Li, C. Wang, W. Xie, and J. Xv, "A 3D placement of unmanned aerial vehicle base station based on multi-population genetic algorithm for maximizing users with different QoS requirements," in *Proceeding of IEEE 18th International Conference on Communication Technology (ICCT2018)*, pp. 967–972.

[20] Z. Xue, J. Wang, G. Ding, and Q. Wu, "Joint 3D location and power optimization for UAV-enabled relaying systems," *IEEE Access*, vol. 6, pp. 43 113–43 124, Aug. 2018.

[21] M. Koivisto, M. Costa, A. Hakkarainen, K. Leppanen, and M. Valkama, "Joint 3D positioning and network synchronization in 5G ultra-dense networks using UKF and EKF," in *Proceeding of IEEE Globecom Workshops (GC Wkshps2016)*, pp. 1–7.

[22] E. Kalantari, I. Bor-Yaliniz, A. Yongacoglu, and H. Yanikomeroglu, "User association and bandwidth allocation for terrestrial and aerial base stations with backhaul considerations," in *Proceeding of IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC2017)*, pp. 1–6.

[23] Z. Zhu, L. Li, and W. Zhou, "QoS-aware 3D deployment of UAV base stations," in *Proceeding of IEEE 10th International Conference on Wireless Communications and Signal Processing (WCSP2018)*, pp. 1–6.

[24] X. He, W. Yu, H. Xu, J. Lin, X. Yang, C. Lu, and X. Fu, "Towards 3D deployment of UAV base stations in uneven terrain," in *Proceeding of*

IEEE 27th International Conference on Computer Communication and Networks (ICCCN), 2018, pp. 1–9.

[25] M. Mozaffari, A. T. Z. Kasgari, W. Saad, M. Bennis, and M. Debbah, "Beyond 5G with UAVs: Foundations of a 3D wireless cellular network," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 357–372, Nov. 2018.

[26] A. Ahmed, M. Awais, T. Akram, S. Kulac, M. Alhussein, and K. Aurangzeb, "Joint placement and device association of UAV base stations in IoT networks," *Sensors*, vol. 19, no. 9, p. 2157, May 2019.

[27] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV cellular communications: Practical aspects, standardization advancements, regulation, and security challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3417–3442, Fourthquarter 2019.

[28] S. Shakoor, Z. Kaleem, D.-T. Do, O. A. Dobre, and A. Jamalipour, "Joint optimization of UAV 3D placement and path loss factor for energy efficient maximal coverage," *IEEE Internet of Things Journal - Early Access*, Aug. 2020.

[29] 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, Nov. 2016.

[30] X. Zhang and L. Duan, "Fast deployment of UAV networks for optimal wireless coverage," *IEEE Transactions on Mobile Computing*, vol. 18, no. 3, pp. 588–601, May 2018.

[31] T. S. Madhulatha, "Comparison between k-means and k-medoids clustering algorithms," in *Proceeding of International Conference on Advances in Computing and Information Technology*. Springer, 2011, pp. 472–481.

[32] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96. AAAI Press, 1996, p. 226–231.

[33] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[34] D. J. Daley and D. Vere-Jones, *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media, 2007.

[35] R. L. Streit, *Poisson point processes: imaging, tracking, and sensing*. Springer Science & Business Media, 2010.