The Optimal and the Greedy: Drone Association and Positioning Schemes for Internet of UAVs

Hajar El Hammouti, Doha Hamza, Basem Shihada, Mohamed-Slim Alouini, and Jeff S. Shamma
CEMSE division, King Abdullah University of Science and Technology (KAUST), Thuwal, Makkah Province, KSA.
{hajar.hammouti,doha.hamzamohamed,basem.shihada, slim.alouini,jeff.shamma}@kaust.edu.sa

***Abstract*—This work considers the deployment of unmanned aerial vehicles (UAVs) over a pre-defined area to serve a number of ground users. Due to the heterogeneous nature of the network, the UAVs may cause severe interference to the transmissions of each other. Hence, a judicious design of the user-UAV association and UAV locations is desired. A potential game is defined where the players are the UAVs. The potential function is the total sum-rate of the users. The agents' utility in the potential game is their marginal contribution to the global welfare or their so-called wonderful life utility. A game-theoretic learning algorithm, binary log-linear learning (BLLL), is then applied to the problem. Given the potential game structure, a consequence of our utility design, the stochastically stable states using BLLL are guaranteed to be the potential maximizers. Hence, we optimally solve the joint user-UAV association and 3D-location problem. Next, we exploit the submodular features of the sum rate function for a given configuration of UAVs to design an efficient greedy algorithm. Despite the simplicity of the greedy algorithm, it comes with a performance guarantee of $1-1/e$ of the optimal solution. To further reduce the number of iterations, we propose another heuristic greedy algorithm that provides very good results. Our simulations show that, in practice, the proposed greedy approaches achieve significant performance in a few iterations.**

***Index Terms*—UAV-enabled networks, users-UAVs association, UAV 3D placement, potential game, binary log-linear learning, greedy algorithm.**

I. Introduction

According to a recent report of the federal aviation authority (FAA) [1], the number of drones in the USA has reached 2 millions in 2019 and is estimated to attain 2.5 millions by 2025. Indeed, in the near future, thousands of drones are expected to navigate autonomously over cities to deliver a plethora of services such as traffic reporting, package delivery, and public surveillance [2], [3]. The main virtue of such technology is the high mobility of drones, their versatile nature, their rapid deployment, and the wide range of services they can provide.

One of the earliest applications of drone-related services is in the telecommunications industry [4], [5]. Equipped with smart transceivers, drones can be deployed as flying base stations that extend coverage in crowded places and remote areas. They can also be deployed as aerial relays that collect or disseminate data in an Internet of things environment. Also, thanks to their fast deployment, drones can be used in a post-disaster scenario to replace damaged ground base stations.

Although the application of drones in the telecommunications industry is very appealing, their efficient deployment still faces several technical challenges that range from trajectory planning to channel modeling [6] and 3D placement [7]–[14]. In this paper, we are interested in the optimal deployment of drones coupled with the optimal drone-users association. This problem is widely investigated in the literature but, to the best of authors' knowledge, none of the existing works provides an optimal solution to the studied problem, specifically when interference is considered. Furthermore, only a few works measure the efficiency of their proposed approach against the optimal one. In this paper, we undertake this task by answering the following questions: What is the optimal deployment of UAVs and the optimal drone-users association to maximize the downlink sum-rate, in the presence of interference and with bandwidth and quality of service constraints? What is the cost of such an optimal solution? Are there any alternative approaches that reach an efficient solution to the problem in a fewer number of iterations? And how efficient is this solution?

A. Related Work

The optimal 3D placement of UAVs received considerable attention in the last few years. One of the earliest works to study the placement of the drones in the 3D space for communications purposes is the work by Mozaffari *et al.* in [15]. In that paper, the authors provide closed-form expressions for an optimal height that maximizes the drones' coverage area. The work mainly focuses on the cases of single and two drones. For the two drones scenario, the authors show that the presence of interference increases the complexity of the system leading to a challenging optimization problem. This problem has been extended in [16] to a multiple drones scenario. In [16], the authors consider interference coming from the nearest neighbor only. This approximation results in a tractable coverage optimization problem that is solved using circle packing theory. In general, when interference is not considered, the objective function becomes convex with respect to the 3D placement. To solve this problem, the authors in [17] adopt a gradient descent based algorithm to efficiently place the UAV in order to minimize the transmit power required to cover indoor users. The problem of the 3D placement of the UAVs is also tackled in [18] and solved by decoupling the horizontal and the vertical placements. The objective in [18] is to maximize the number of covered users.

Moreover, the overall UAV-enabled network performance is tightly related to the number of served users. In the classical network association, users are either served by the closest base station (Voronoi association), or they are assigned to the base station with the best signal-to-interference-and-noise-ratio (weighted Voronoi association). In either case, the distance-only based association may result in highly congested base stations and unbalanced resource allocation across the network. Hence, many works can be found in the literature that study the association rule along with the 3D placement of the UAVs. However, the joint 3D optimization and users' association is challenging. One commonly used approach is to decompose the studied optimization into subproblems where each subproblem is addressed separately. The results of each subproblem are used as inputs for the next one, and the process is repeated until convergence is reached. While such an approach can provide satisfactory results, it is not guaranteed to reach the global optimum. When using a decomposition process, the algorithm will often halt at a suboptimal solution with no guaranteed bounds on the suboptimality gap. Moreover, most of the proposed iterative approaches have no provable convergence properties.

For example, by using k-means and particle swarm optimization sequentially, the joint users' association and 3D location was addressed in [19] in order to maximize the logarithmic rate of the users under delay and backhaul constraints. A similar decomposition approach is proposed in [20] where devices are first connected to UAVs using matching theory, and then the UAVs are placed optimally in the 2D space by leveraging tools from control theory. In [21], cell partitioning is proposed to cluster the users, and then, the non-convex 3D placement optimization problem is solved using sequential quadratic programming. In line with the previous works, the approach proposed in [22] relies on combining distributed algorithms in order to address the users' association, the 2D placement, and the altitude adjustment subproblems separately. In [23], some other practical requirements such as power control and channel access are considered. In order to achieve a minimum consumed energy, the authors adopt an iterative mechanism based on a coordinate descent algorithm. The authors solve each resource allocation subproblem separately, which leads to a sub-optimal yet efficient solution. The same concept is used in [24] where the primary problem that aims at minimizing the total transmit power is decoupled into two subproblems: 2D locations along with users' association, and joint optimization of altitudes and power control. Another iterative process is proposed in [25], where the authors propose a decomposition approach to first position the UAVs, then assign the channels to the users. First, the 2D positions are obtained using a mixed integer second order cone optimization. Second, the channels' assignment is performed by minimizing interference. Similarly, in [26], the optimization problem is decoupled into two subproblems: An optimal altitude placement, and joint 2D and users association subproblem. A sequential approach is also suggested in [27] where the number of UAVs is optimized in the context of mobile edge computing with the aim of minimizing the consumed energy. The authors propose a two-layer optimization mechanism in order to first find the required number of UAVs and their optimal locations, and second, determine if the edge computing tasks should be offloaded to the UAVs or performed locally at the devices.

Due to the complexity of the studied problem, none of the previously cited papers provides an approach that exactly solves the target optimization. Indeed, the studied problem is not only non-convex and challenging to solve but it is also NP-hard. Therefore, a polynomial-time algorithm that exactly solves the optimization problem does not exist [28]. This implies that the optimal solution will necessarily lead to an exponential-time search.

It is important to note that under the terrestrial communications setup, similar resource allocation problems have been investigated, and approaches to reach the exact optimum exist. For example, in [29], the authors propose an algorithm based on a Gibbs sampler to optimize the joint channel selection and users' association in WLAN networks. A more general work is presented in [30] where the authors develop a framework based on Markov Random Fields and Gibbs measures to exactly solve the resource allocation problem in OFDMA networks. Unlike the previously cited works, we tackle the 3D placement problem which is inherent to air-to-ground communications and present a distributed learning mechanism that requires little knowledge of the search space. The learning algorithm, binary log-linear learning (BLLL), is a game-theoretic algorithm that was introduced in [31] and since then has found wide applicability in wireless communications [32]–[34]. The idea is simple: by designing the agents' utilities, we formulate our problem as a potential game among the UAVs. Then only one agent, a UAV, is active at a time. The active agent compares the utilities of two actions: its current action and another feasible one. A Gibbs sampler then chooses the actual action based on probabilities calculated from the potential utilities of the two actions. The work in [31] confirms that such a simple learning rule is guaranteed to linger at the potential maximizers in potential games.

Since the considered problem is NP-hard, the convergence of BLLL can be exponentially slow. Hence, we also provide a greedy algorithm with a performance guarantee of achieving at least $1 - 1/e$ of the optimal solution. Our greedy approach leverages the submodular properties of the studied problem in order to guarantee an efficient performance. We also refer to the papers [35]–[38], that reformulate the resource allocation optimization problem as a submodular maximization problem to provide a lower bound approximation on the proposed solutions. These papers either ignore 3D positioning (in [38]), or interference in the objective function (for the rate maximization in [35], [37]) or consider a very specific objective function with innate monotonicity and submodularity properties (for the caching problem in [36]). The comparison of our work with different works in the literature is summarized in Table I.

Reference	Objective	Optimized variable	Adopted Approach	Performance	Presence of interference
[16]	Maximum coverage + UAVs lifetime	3D placement	Circle packing theory	Sub-optimal	Interfering UAVs
[17]	Minimize total transmit power	3D placement	Particle swarm optimization	Sub-optimal	Single UAV
[18]	Maximize coverage	3D placement	Circle placement problem	Sub-optimal	Single UAV
[19]	Maximize sum-rate	3D placement + User association + bandwidth allocation	Decomposing problem	Sub-optimal	Interfering UAVs
[20]	Maximize coverage	2D placement+association	decomposing problem : Matching+k-means	Sub-optimal	Interfering UAVs
[21]	Maximize uplink energy efficiency	3D placement+ mobility +association+ power	sequential quadratic programming	Sub-optimal	Interfering UAVs
[22]	Maximize sum-rate	3D placement+ association+	Decomposing problem: Matching+k-means +Best response	Sub-optimal	Interfering UAVs Interfering UAVs
[24]	Maximize total transmit power	3D placement+ Association	Decomposing problem: k-means+matching+ alternating iterative method	Sub-optimal	Interfering UAVs
[23]	Minimize energy	3D placement+ Association power+channel access	Iterative coordinate descent optimization	Sub-optimal	No interference between UAVs
[25]	Maximize sum-rate	3D placement+ association	Decomposing problem: Mixed integer program+ assignment algorithm	Sub-optimal	Interfering UAVs
[26]	Average throughput	3D placement+ association	Altitude optimization+ concave–convex procedure	Sub-optimal	Interfering UAVs
[27]	Minimize the system energy	Number of UAVs+ 3D location+ task scheduling	Two-layer optimization: differential evolution algorithm +greedy algorithm	Sub-optimal	No interference
[29]	Maximize sum-rate	Channel+ client selection + channel access+ client scheduling	Decomposing problem: Convex Optimization+ Gibbs sampler	Sub-optimal	Interfering BS
[30]	Maximize sum-rate	Association+ Power control+ Scheduling	Gibbs sampler	Optimal	Interfering BS
[34]	Maximize coverage	2D positions+ range+direction	Gibbs sampler	Optimal	Interfering sensors
[35]	Maximize sum-rate	Association	Submodular maximization	$1 - 1/e$ bound	No interference between BS
[37]	Maximize sum-rate	Beam allocation	Submodular maximization	$1 - 1/e$ bound	Interferings BSs
[36]	Maximize cache hit ratio	3D placement+caching	Submodular maximization	$1/2$ bound	No interference between UAVs
[38]	Maximize sum-rate	Association	Greedy algorithm	$1/2$ bound	Interfering UAVs
[39]	Maximize coverage	3D placement+ Association	Swarm algorithm	Sub-optimal	Interfering UAVs
Our approach	Maximize sum rate	3D placement+ Association	Game-Theoretic+ Submodular maximization	Optimal+ $1 - 1/e$ bound	Interfering UAVs

TABLE I: Summary of literature review

B. Contribution

Based on the previous subsection, we conclude that most of the existing works tackle the problem of 3D placement and resource allocation in multi-UAV enabled networks using a decomposition approach that breaks the primary problem into several subproblems, and address each subproblem separately. Although these approaches can provide efficient solutions, it is not guaranteed that they will reach the global optimum. To the best of our knowledge, none provide the exact solution for the studied optimization problem, specifically under the presence of interference between the UAVs. This is especially true since the formulated optimization problem is NP-hard. On the other hand, few works leverage the submodular framework in order to guarantee an efficient performance of the proposed approach. These papers mainly address the resource allocation problem under the terrestrial communications setup. Hence, our contributions can be summarized as follows:

- We design the UAVs utilities according to marginal contribution utility. We then use BLLL to reach equilibrium. BLLL guarantees convergence to the global optimum. This approach, however time-consuming, is used as a benchmark to assess the performance of other existing approaches.
- We formulate the UAV localization and association problem as a submodular maximization problem subject to a matroid constraint. This formulation enables us to make use of a greedy approach with a performance guarantee of $1 - 1/e$ of the maximum. We believe this is the first work that adapts the submodularity framework to the joint 3D positioning and association problem under interference in a multi-UAV enabled network.

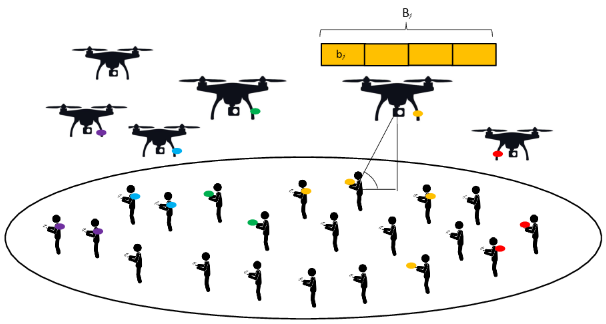


Fig. 1: The figure shows the setup of our problem. UAVs have designated bandwidths B_j divided into subchannels b_j that they can assign to different users. Users assigned to UAVs are indicated by circles of the same color as the UAV.

- We further provide a heuristic greedy algorithm with low information and implementation requirements. Our simulations show that this algorithm achieves efficient results in only a few iterations despite its simplicity.

C. Organization of the Paper

The rest of the paper is organized as follows. In the next Section, we describe the adopted system model. Then, we formulate our optimization problem in Section III. In Section. IV, we formulate our problem as a potential game among the UAVs, and implement BLLL in order to find the optimal 3D placement and users' association that maximize the sum-rate function. Next, we study the submodularity of the objective function and the matroid structure of the constraints in Section V. Two greedy approaches are studied in Section VI. The proposed algorithms are compared in Section VII. Finally, simulation results are provided in Section VIII and we conclude the paper in Section IX.

D. Notation

We adopt the following notation: The Cartesian products of two sets A and B is denoted $A \times B$. $|A|$ denotes the cardinality of set A. Vectors and matrices are denoted using boldface letters $\mathbf{x}$, whereas scalars are denoted by x.

II. System Model

We assume a drones-enabled network where a set $\mathcal{J}$ of UAVs are deployed over a target area to serve a set $\mathcal{I}$ of ground users. The system model is depicted in Fig. 1. In order to capture the channel variations between the user and the UAV, we adopt the commonly used air-to-ground channel model where the path loss is averaged over line-of-sight (LoS) and non-line-of-sight (NLoS) links and the probability of LoS is given by [40]:

$$p_{ij}^{\mathrm{LoS}}(r_{ij}, d_{ij}) = \frac{1}{1 + \epsilon \cdot \exp\left(-\beta \frac{180}{\pi} \arctan \frac{\sqrt{r_{ij}^2 - d_{ij}^2}}{d_{ij}} - \epsilon\right)}, \tag{1}$$

where d_{ij} is the 2D plane distance from the projected position of UAV j to user i, r_{ij} is the distance between the UAV and the user, ϵ and β are environment-dependent parameters.

Consequently, the path loss between UAV j and user i can be formulated as:

$$L_{ij}(r_{ij}, d_{ij}) = \left(\frac{4\pi f r_{ij}}{c}\right)^{-\alpha} \left(\zeta_{\mathrm{LoS}} p_{ij}^{\mathrm{LoS}}(r_{ij}, d_{ij}) + \zeta_{\mathrm{NLoS}}(1 - p_{ij}^{\mathrm{LoS}}(r_{ij}, d_{ij}))\right)^{-1}, \tag{2}$$

where f is the carrier frequency, c is the speed of light, and α is the path loss exponent. ζ_{LoS} and ζ_{NLoS} are the parameters for LoS and NLoS losses, respectively.

Accordingly, the signal-to-interference-and-noise-ratio (SINR) received at user i from UAV j can be written as:

$$\gamma_{ij} = \frac{P_j L_{ij}(r_{ij}, d_{ij})}{\sigma^2 + \sum_{\ell \neq j} P_\ell L_{i\ell}(r_{i\ell}, d_{i\ell})}, \tag{3}$$

where P_j is the transmit power of UAV j and, σ^2 is the variance of the additive white Gaussian noise at user i's receiver, and subscript ℓ denotes the ℓth interfering UAV.

We consider the downlink communication channel. We are interested in the spectral efficiency η_{ij} between user i and UAV j, given by:

$$\eta_{ij} = \log_2(1 + \gamma_{ij}). \tag{4}$$

Due to backhaul limitations, we assume that each UAV j has a limited number of users N_j to connect with. Furthermore, each UAV has a limited amount of bandwidth B_j divided into equal subchannels b_j so that $B_j = N_j b_j$. Allocating equal subchannels enforces some fairness among the UAV's served users since otherwise all bandwidth should go to the user with the highest spectral efficiency. Therefore, each ground user associated with UAV j receives a throughput R_{ij} that can be formulated as:

$$R_{ij} = b_j \eta_{ij}. \tag{5}$$

We note that the UAVs are deployed in dense and/or uncovered areas, that is the number of users is, most of the time, higher than the allowed UAV quota N_j. However, if N_j exceeds the number of effectively associated users, N_j^{eff}, each UAV can improve the bandwidth allocated to its served users by equally dividing its total bandwidth into bigger subchannels such that $b_j^{\mathrm{eff}} = B_j / N_j^{\mathrm{eff}}$ between its associated users so that $b_j^{\mathrm{eff}} > b_j$ and all the UAV bandwidth is allocated.

III. Problem Formulation

We are interested in the downlink sum-rate of the ground users. Our objective is to optimally deploy the UAVs in the 3D space and associate the users in order to maximize the sum-rate function. Let $\mathbf{q} = (q_{ij})$ be the binary UAVs-users association matrix and $(\mathbf{x}, \mathbf{y}, \mathbf{h})$ the UAVs 3D positions. Let $\mathcal{I}$ be the set of users and $\mathcal{J}$ the set of UAVs. Our optimization problem is formulated as follows:

$$\begin{aligned}
\underset{\mathbf{q},(\mathbf{x},\mathbf{y},\mathbf{h})}{\text{maximize}} \quad & \sum_{j\in\mathcal{K}}\sum_{i\in\mathcal{U}} q_{ij}R_{ij} & \text{(6a)}\\
\text{subject to} \quad & \sum_{i} q_{ij} \leq N_j \quad \forall j\in\mathcal{J}, & \text{(6b)}\\
& \frac{q_{ij}}{\eta_{ij}} \leq \frac{1}{\eta^{\min}} \quad \forall (i,j)\in\mathcal{I}\times\mathcal{J}, & \text{(6c)}\\
& x^{\min} \leq x_j \leq x^{\max} \quad \forall j\in\mathcal{J}, & \text{(6d)}\\
& y^{\min} \leq y_j \leq y^{\max} \quad \forall j\in\mathcal{J}, & \text{(6e)}\\
& h^{\min} \leq h_j \leq h^{\max} \quad \forall j\in\mathcal{J}, & \text{(6f)}\\
& \sum_{j} q_{ij} \leq 1 \quad \forall i\in\mathcal{I}, & \text{(6g)}\\
& q_{ij}\in\{0,1\} \quad \forall (i,j)\in\mathcal{I}\times\mathcal{J}. & \text{(6h)}
\end{aligned}$$

Constraint (6b) ensures that the number of associated users for each UAV j does not exceed its maximum quota of users N_j. Constraint (6c) guarantees a certain quality of service for each associated user by ensuring that its spectral efficiency is no less than a predefined threshold $\eta^{\min}$. Constraints (6d), (6e) and (6f) ensure that the 3D coordinates of all the UAVs are bounded to a target cubic space. Finally, constraints (6g) and (6h) restrict the ground user to be associated with, at most, one UAV.

The problem under analysis is mathematically challenging as it involves a non-convex objective function and a non-convex constraint (constraint (6c)). It also includes integer and continuous variables which make it a mixed integer non-linear program (MINLP). Moreover, the association problem can be formulated as a knapsack problem, which is known to be NP-hard.

In the following, we solve this optimization problem using a game-theoretic approach. The optimal solution of the studied problem is therefore obtained using BLLL, a learning algorithm that provides guarantees on reaching the maximizers of the objective function when the underlying game is a potential game.

IV. Sum-Rate Maximization

We discretize the 3D space and represent it in the form of a 3D grid. We formulate the interactions between the UAVs as a potential game where the downlink sum-rate is the potential function. Then, BLLL is implemented by the UAVs in order to find the optimal 3D placement and users' association that maximize the sum-rate function.

A. Game Formulation

1) Background: In game theory, a potential game is a game where any unilateral change in a player's utility results in an equal change in a global welfare function called the *potential function*. Therefore, whenever a player performs an action that improves its utility, it also improves the potential function. More formally, the definition of a potential game is given below.

Definition 1 (Player actions). *We will use a_j to denote the action of player j. Player j's actions belong to a finite strategy set $\mathcal{A}_j$. Let $\mathcal{A}=\prod_j \mathcal{A}_j$ be the resulting strategy profile of all players. It will be useful to decompose a given action $\boldsymbol{a}\in\mathcal{A}$ into two components, $(a_j,\boldsymbol{a}_{-j})$ where $\boldsymbol{a}_{-j}$ denotes the actions of players other than j.*

Definition 2 (Potential game). *[41] $\mathcal{G}$ is a potential game if there exists a potential function $F:\mathcal{A}\longrightarrow\mathbb{R}$ such that for each player j, $\forall(a_j,\boldsymbol{a}_{-j})$ and $(a'_j,\boldsymbol{a}_{-j})\in\mathcal{A}$*

$$F(a_j,\boldsymbol{a}_{-j})-F(a'_j,\boldsymbol{a}_{-j})=U_j(a_j,\boldsymbol{a}_{-j})-U_j(a'_j,\boldsymbol{a}_{-j}), \quad (7)$$

where U_j is the utility of player j. Note that U_j depends on a, the action taken by all agents.

2) UAVs potential game: Let us consider the 3D grid, where $X=\{x^{\min},x^{\min}+\delta x,x^{\min}+2\delta x,\dots,x^{\max}\}$, $Y=\{y^{\min},y^{\min}+\delta y,y^{\min}+2\delta y,\dots,y^{\max}\}$ and $H=\{h^{\min},h^{\min}+\delta h,h^{\min}+2\delta h,\dots,h^{\max}\}$ represent the x-,y- and z-axis, respectively and with $\delta x,\delta y,\delta h>0$ representing their respective step granularity. Let $Q=\{0,1\}$ be an indicator. $\mathcal{G}^d=\{\mathcal{J},\mathcal{A},\{U_j\}_{j\in\mathcal{J}}\}$ is the game where the UAVs are the players, $\mathcal{A}=X\times Y\times H\times Q^{\mathcal{I}}$ is the set of their actions, and $U_j:\mathcal{A}\to\mathbb{R}$ the utility function such as, given the 3D deployment of all UAVs and association for all the users, the outcome of UAV j is given by its marginal contribution:

$$U_j(\mathbf{x},\mathbf{y},\mathbf{h},\mathbf{q})=\sum_{j'\in\mathcal{J}}\sum_{i\in\mathcal{I}} q_{ij'}R_{ij'}-\sum_{j'\in\mathcal{J}\setminus\{j\}}\sum_{i\in\mathcal{I}} q_{ij'}R_{ij'}(-j), \quad (8)$$

where $R_{ij'}(-j)=b_{j'}\log_2\big(1+\frac{P_{j'}L_{ij'}(r_{ij'},d_{ij'})}{\sigma^2+\sum_{\ell\neq j,j'}P_\ell L_{i\ell}(r_{i\ell},d_{i\ell})}\big)$ is the perceived rate at user i when interference from UAV j is not considered. Here, the marginal contribution of UAV j is given by the difference of the sum-rate when UAV j is part of the network configuration and when it is not. When UAV j is removed, the rates of its associated users are not considered in the sum-rate.

Proposition 1 (UAVs potential game). *The game $\mathcal{G}^d=\{\mathcal{J},\mathcal{A},\{U_j\}_{j\in\mathcal{J}}\}$ is a potential game where the potential function is the sum-rate function.*

Proof. This result is straightforward and stems from the design of the utility function. □

B. Binary Log-Linear Learning (BLLL)

Binary log-linear learning is a game-theoretic algorithm that belongs to the class of log-linear learning (LLL) mechanisms. Contrary to LLL, BLLL mechanisms do not assume complete knowledge of all the agent's action set and their associated utilities. Instead, when BLLL is implemented, an active agent compares its current action against a single *trial* action from a constrained action set [31]. Furthermore, only one player is active in any given time instance and all other players repeat their previous actions. Since our problem is formulated as a potential game, the BLLL fits our framework. The actions of the agents are the choice of locations and the associations with the users. Due to geographical constraints, the agents, the UAVs, can only consider trial actions within the neighborhood of their current locations and a subset of users that are nearby. Here is how the BLLL runs in our setup: Only one agent from the players' set is activated at any time instant. The activated agent chooses a trial action from its current feasible actions, uniformly at random. The player then plays this action a_j with the probability:

$$p(a_j) = \frac{e^{U_j(a)/T}}{e^{U_j(a')/T} + e^{U_j(a)/T}}, \tag{9}$$

where $a = (a_j, \mathbf{a}_{-j})$ $a' = (a'_j, \mathbf{a}_{-j})$ are the collective actions when agent j plays a_j and a'_j, respectively. $T > 0$ is a tuning parameter called the temperature of the algorithm. The temperature determines how likely a player is to select a suboptimal action. As the temperature goes to 0, players are likely to select only optimal actions, a so-called exploitation phase in the learning mechanisms. High values of T lead to more exploration of different actions, and a higher probability of choosing a suboptimal action. A judicious choice of T leads to a balanced exploitation and exploration and is a desired feature of the learning mechanism.

Corollary 1. *[31] If all players adhere to BLLL, then the only stochastically stable states are the maximizers of the potential function.*

Using BLLL, the active player, j, selects -at random- a feasible strategy a_j (i.e., a neighboring 3D location and a subset of users to associate with without exceeding its maximum quota). Then, the player computes its utility regarding the selected strategy a_j. It then calculates the related Gibbs probability $p(a_j)$ as described in equation (9). Finally, the strategy a is adopted with probability $p(a_j)$. Clearly, the probability of adopting an action increases when the utility with respect to this action increases versus the existing action. Hence, the better the strategy, the higher the probability to adopt it. However, it is far from intuitive how such an updating rule can converge to a global optimum or how it may even converge. The proof of convergence of such a process is based on *the theory of resistance trees* and can be found in details in [31].

Algorithm 1 summarizes the BLLL mechanism employed by the UAVs. When a UAV wakes up, according to its timer (line 6), it selects at random a neighboring location in the 3D grid -with respect to its current position- and a random association with the users -up to its quota limit. The UAV then computes the new utility regarding this joint 3D position and association, and then decides whether to adopt this new action with a probability calculated using equation (9). In order to meet constraint (6c), users who are not satisfied with their spectral efficiency are disconnected from the UAV and their rates are not included in the utility of that UAV. This process is iterated while slowly decreasing the temperature T (line (4)).

Algorithm 1 BLLL for joint 3D position and users association selection

1: **Initialization**:
2: $(\mathbf{x}, \mathbf{y}, \mathbf{h})$ random matrix for 3D locations of UAVs
3: $q_{ij} = 0,\ U_j(\mathbf{a}') = 0\ \forall (i,j) \in \mathcal{I} \times \mathcal{J}$.
4: **for** $T \to 0$ **do**
5:   **for** $j \in \mathcal{J}$ **do**
6:     **if** $rand(1) > 0.5$ **then**
7:       Active UAV j calculates the following:
8:       Select a location at random from $(x_j \pm \delta x, y_j, h_j)$, $(x_j, y_j \pm \delta y, h_j)$,$(x_j, y_j, h_j \pm \delta h)$
9:       Select at random a number of unconnected users, s.t. the number of selected users $\leq N_j$ and $\eta^{\min}$ satisfied.
10:       Compute $U_j(\mathbf{a})$ as in equation (8)
11:       Use $U_j(\mathbf{a})$, and $U_j(\mathbf{a}')$ to sample new location and association using the probability in equation (9)
12:       **if** action a' is selected **then**
13:         UAV j updates current location, associations, and $U_j(\mathbf{a}') \leftarrow U_j(\mathbf{a})$.

We note that our proposed approach is versatile and suitable for many applications and scenarios. For example, in work [27], the authors considered a two-layer optimization method for jointly optimizing the deployment of UAVs and user task scheduling, with the aim of minimizing system energy consumption. The BLLL approach can be extended to deal with this situation. In this case, the action of the UAV involves the choice of location and whether or not to participate. The combined action (location + participation) yields a certain utility. Once the game is defined with these combined actions, our results follow and hold. In fact, the BLLL approach will yield the optimal value for many objectives provided we are able to correctly define the agents' actions and associated utilities. A similar modification also is possible to the greedy and adapted greedy algorithms that we propose in the next section.

We also note that BLLL helps serve as a benchmark to assess the optimal performance that can be attained in the network. However, in order to reach the global optimum, BLLL requires an exponential time for convergence. To circumvent this problem, we propose a greedy approach that is guaranteed to converge to at least $1 - 1/e$ of the optimum. Our greedy algorithm relies on the submodular property of the objective function that we discuss in the next section.

V. Submodularity of the Objective Function

In this Section, we proceed to analyze the submodularity of our objective function and the matroid structure of the constraints similar to the approach in [42]. This analysis will facilitate the greedy algorithm which we employ to solve our problem. First, we introduce the mathematical definitions of submodularity and matroids. Then, we reformulate the problem as a set function maximization problem under a partition matroid constraint. This lays the foundation for the greedy algorithm we use to solve our problem in the next Section.

We note here that submodular function maximization arises in several optimization problems. For example, the knapsack problem, the coverage problem, and the traveling salesman problem can be formulated as submodular set function maximization problems. While submodular minimization problems can generally be solved efficiently, submodular maximization problems are often NP-hard to solve. The main virtue of such a mathematical framework, however, is that one can construct simple greedy algorithms with provably good results to solve the problem. The performance of the greedy approach depends on the monotonicity of the objective function and the nature of the constraints. In particular, when the objective function is submodular and monotone, the greedy algorithm is guaranteed to achieve at least $1 - 1/e$ of the optimal value [43]. In the following, we show that our optimization problem is endowed with the submodular structure as adding more users to any UAV follows a law of diminishing returns subject to a matroid constraint. Hence, we adopt the submodularity framework to the joint positioning and association problem, and propose a greedy algorithm to achieve desirable performance. In reality, our algorithms perform better than the $1 - 1/e$ bound as evidenced by comparison to the optimal bound achieved using BLLL.

A. Basic Definitions

Assume a ground set V. Let 2^V be the collection of all subsets of V. In discrete optimization, we say that a set function $f : 2^V \to \mathbb{R}$ is submodular if it satisfies the following property.

Definition 1 (Submodularity [44])**.** *f is said to be submodular if for $A \subseteq B \subseteq V$ and $a \in V \setminus B$:*

$$f(A \cup \{a\}) - f(A) \geq f(B \cup \{a\}) - f(B) \tag{10}$$

An intuitive interpretation of submodularity suggests that the marginal gain of adding an element a to a small set A is greater than or equal to adding the same element to the larger set B.

Furthermore, a set function is said to be monotone if its value increases when more elements are added to any given set. More formally,

Definition 3 (Monotonicity)**.** *A set function $f : 2^V \to \mathbb{R}$ is monotone if $\forall A \subseteq B \subseteq V$:*

$$f(A) \leq f(B). \tag{11}$$

We will shortly show that one of our constraints can be described as a matroid. A matroid is an algebraic structure that generalizes the concept of independent vectors in linear algebra. In particular,

Definition 4 (Matroid)**.** *A matroid $M = (V, \mathcal{L})$ consists of a non-empty finite set ground V and a non-empty collection $\mathcal{L}$ of subsets of V that satisfy the following properties:*

1) $\emptyset \in \mathcal{L}$
2) If $I \in \mathcal{L}$ and $J \subset I$, then $J \in \mathcal{L}$.
3) $I, J \in \mathcal{L}$ and $|I| > |J|$, then there exists $e \in I \backslash J$ such that $J \cup \{e\} \in \mathcal{L}$

The first two conditions describe the *"hereditary property"*. This property suggests that each subset of a set in the collection $\mathcal{L}$ inherits the independence property. The third condition is usually called the *"augmentation property"*. It implies that each element of the collection can be augmented to a larger set while maintaining the independence property.

When the ground set V is partitioned into disjoint subsets $V_1, V_2, \ldots, V_t$, where t is a strictly positive integer, a particular class of matroids, called *partition matroid* emerges:

Definition 5 (Partition matroid)**.** *A partition matroid $M = (V, \mathcal{L})$ is a matroid such that V is partitioned into t disjoint partition sets $V_1, V_2, \ldots, V_t$ and $\mathcal{L} = \{X \subseteq V : |X \cap V_i| \leq k_i,\ \forall i = 1, 2, \ldots t\}$, where $0 \leq k_i \leq |V_i|$ are some given integer parameters.*

Now that we have introduced the proposed mathematical framework, let us reformulate the studied problem as a set function maximization problem.

B. Problem Reformulation

To begin, let $\mathcal{K}$ be the set of all possible configurations of the UAVs[1]. If there are $|\mathcal{J}| = J$ UAVs and $|\mathcal{L}| = L$ locations, there are $K = L^{2J-1}$ possible configurations where $K = |\mathcal{K}|$. We also add a null UAV for each user to allow for the possibility that some users will be unassigned.

We define our ground set $V = \{(i, j, k) : i \in \mathcal{I}, j \in \mathcal{J}, k \in \mathcal{K}\}$. V contains all the tuples formed by users, UAVs, and network configurations. We then partition the ground set V into K disjoint subsets, $V_1^C, V_2^C, ..., V_K^C$, where $V_k^C = \{(i, j, k), i \in \mathcal{I}, j \in \mathcal{J}, \}$ where V_k^C is the set of all possible associations under a given configuration k and the superscript indicates that the partition is according to the configuration index. Hence, the constraint that only one configuration is possible can be written as finding set $A \in I_C$ where:

$$I_C = \begin{cases} A \subseteq V : |A \cap V_k^C| \leq e_k \text{ for some k} \in \mathcal{K}, \\ A \subseteq V : |A \cap V_n^C| = 0 \ \forall \ n \in \mathcal{K} \setminus \{k\} \end{cases} \tag{12}$$

where e_k is some number denoting the intersection of the two sets[2]. This constraint merely implies that, in the end, only one configuration is selected.

Remark. *It is noted that we could also set up a constraint for the UAV quota and another for the users' quota. However, we will show that this is not needed. We simply delegate the UAV quota, the users' quota and the minimum spectral efficiency conditions, conditions (6c), (6g) and (6h) in the optimization problem, to the set function evaluation. We also note that*

[1]A network configuration designates a given network realization where the 3D locations of UAVs are fixed at some positions of the 3D grid.

[2]e_k can be seen as the total number of associated users for a given configuration.

considering the set function evaluation over configurations helps us fix the interference experienced by users for a given configuration. As we show in the proof, this helps recover monotonicity and submodularity of the set function evaluated over a given configuration.

Proposition 2. *$M_C = (V, I_C)$ is a partition matroid.*

Proof. We consider feasible sets $A \subseteq B \subset \mathcal{V}$. To maintain feasibility, A and B must belong to the same configuration. The proof follows immediately using the approach in [45]. □

In light of the above definitions, our optimization problem can now be written as:

$$\underset{A \in 2^V}{\text{maximize}} \quad f(A) \tag{13a}$$
$$\text{subject to} \quad A \in I_C \tag{13b}$$

The above problem can be equivalently written as:

$$\underset{k \in \mathcal{K}}{\text{maximize}} \ \underset{A \in I_C}{\text{maximize}} \ f^k(A) \tag{14}$$

where,

$$f^k(A) = \sum_{v_{ijk} \in A} R^k_{ij}. \tag{15}$$

and $f^k(.)$ refers to the function evaluation over a given configuration. Since we must enforce that $A \in I_C$, we can only consider sets taking elements that belong to the same configuration, and R^k_{ij} is the rate of user i when it is associated with UAV j and configuration k is adopted. We now use the superscript k to emphasize that the rate of user i with UAV j is calculated for a particular configuration k, so that we can set the interference for a particular configuration at a constant value.

Proposition 3. *$f^k(.)$ is monotone and submodular.*

Proof. We prove monotonicity first. Without loss of generality (WLOG), consider two subsets $A \subseteq B \subseteq V^C_k$, i.e., belonging to the same configuration set k. Let A contain a number of UAVs with a given association for the users. Let B contain A in addition to another UAV with its associated users, then $f^k(A) \leq f^k(B)$ is always true.

We proceed to prove submodularity. Consider any subset $A \subseteq B \subseteq V^C_k$ and $a \in V^C_k \setminus B$. WLOG, let A be the set containing possible associations for users with UAV j at configuration k such that $|A| = N_j - 1$, let $B = A \cup \{b\}$, where b is some feasible element to be added to the set of users associated with UAV j. It is clear that $|B| = N_j$, hence B is at UAV$_j$'s quota limit. WLOG, let $\{b\} = \arg\min_B R^k_{ij}$, i.e. $\{b\}$ is also the element with the minimum contribution to the value $f^k(B)$. Now, consider the addition of another feasible element, a to sets A and B:

$$f^k(A \cup \{a\}) - f^k(A) = f^k(\{a\}), \tag{16}$$

while

$$f^k(B \cup \{a\}) - f^k(B) = \begin{cases} 0 < f^k(\{a\}) \text{ if } \eta_b \geq \eta_a, \\ f^k(\{a\}) - f^k(\{b\}) < f^k(\{a\}), \text{ if } \eta_a > \eta_b, \end{cases} \tag{17}$$

where in the above, and with a slight abuse of notation, we use η_b to denote the spectral efficiency of element b. Hence, in both cases ($\eta_b \geq \eta_a$ and $\eta_a > \eta_b$), we have $f^k(\{a\}) - f^k(\{b\}) < f^k(\{a\})$, while $f^k(A \cup \{a\}) - f^k(A) = f^k(\{a\})$ as shown in equation (16). Accordingly,

$$f^k(B \cup \{a\}) - f^k(B) \leq f^k(A \cup \{a\}) - f^k(A), \tag{18}$$

therefore, f^k is submodular.

□

C. K Instances of the Greedy Algorithm

Using the fact that $f^k(.)$ is monotone and submodular, we can now use a simple greedy algorithm to find the locations and associations for the UAVs and users. We use a greedy algorithm to evaluate the maximum for $f^k(.)$ for a given configuration, and then exhaustively find the maximum value for the set function over all configurations. The overall guaranteed performance is $1 - 1/e$-optimal. This is facilitated by the following lemma:

Lemma 1. *Let $(\mathcal{P})$ be the problem of maximizing a monotone and submodular set function, i.e. $f^k(.)$. Consider the greedy algorithm which starts with an empty set A_0, and at each iteration i, it adds an element e that maximizes $f^k(A_{i-1} \cup \{e\}) - f^k(A_{i-1})$, i.e.,*

$$A_i = A_{i-1} \cup \{\arg\max_{\{e\}} f^k(A_{i-1} \cup \{e\}) - f^k(A_{i-1})\}. \tag{19}$$

The greedy algorithm provides $1 - 1/e$-approximation to the optimal solution of $(\mathcal{P})$ [43].

While the above greedy algorithm ensures a good network performance, it requires listing all the possible configurations, which is time and memory consuming. However, we do not in fact need to list all the possible configurations. One approach to reduce the search space is to select the locations that are critical and are most likely to provide the best performance; in particular, the barycenters of the users' concentrations. For this purpose, we first run k-means as described in **Algorithm** 2. Each UAV is moved in the 2D plane to the barycenter of a cluster of users. The users within the same cluster are selected based on their SINR. Specifically, users are grouped with the UAV that maximizes their SINR. Hence, k-means selects the best 2D locations based on an SINR criterion. Then, a list of 3D configurations is formed by the 2D locations and the various possible heights of the setup. This process will drastically reduce the number of possible configurations without jeopardizing performance as we show in the numerical simulations. The k-means combined with the greedy algorithm is described in **Algorithm** 3.

Algorithm 2 K-means

```
Initialization:
  UAVs uniformly distributed in the 2D space,
  C_j = ∅, ∀j ∈ 𝒥
Choose N, the maximum number of iterations.
for n = 1 : N do
    for j ∈ 𝒥 do
        for i ∈ ℐ do
            if i = arg max_i η_ij then
                C_j = C_j ∪ {i}
        x_j = (Σ_{i∈ℐ} x_i) / |C_j|
        y_j = (Σ_{i∈ℐ} y_i) / |C_j|
```

VI. Greedy Approaches

In this section, we describe the greedy algorithm that efficiently solves the underlying optimization with $1 - 1/e$-approximation. Then, we provide a faster heuristic that achieves very good results in practice.

A. Greedy Algorithm

Algorithm 3 Greedy algorithm

```
1:  Initialization:
2:    Run Algorithm 2 to reduce the number of 2D points.
3:    List in 𝒦 all the possible configurations of UAVs that are
      formed by the 2D points and the studied heights.
4:  S = 0, initilization of the maximal sum-rate
5:  N_j^Current = 0, ∀j ∈ 𝒥, initialization of number of associ-
      ated users to each UAV.
6:  k^best = k_1, initilization of the best configuration
7:  q_ij^k = 0, ∀i ∈ ℐ, j ∈ 𝒥, k ∈ 𝒦
8:  ℒ = ℐ, the set of not associated users
9:  for k a potential configuration do
10:     for n = 1 : I × J do
11:         find (i, j) s.t. (i, j) = arg max_(i,j) (R_ij^k)
12:         if user Σ_j q_ij^k = 0 is not associated, Σ_i q_ij^k ≤ N_j,
      and η_ij ≥ η^min then
13:             q_ij^k = 1
14:             N_j^Current = N_j^Current + 1
15:     if Σ_(i,j) q_ij^k R_ij^k > S then
16:         S = Σ_(i,j) q_ij^k R_ij^k
17:         k^best = k
18:         q^best = q^k
```

As stated in **Lemma** 1, the greedy algorithm will start by selecting the maximum rate for each configuration. Indeed, as described in **Algorithm** 3, for each configuration, the greedy algorithm connects the user-UAV pair associated with the maximum rate among all possible users-UAV pairs of the selected configuration (line (11)). The associated user is then removed from the list of considered users and the quota of its serving UAV is decremented (lines (13),(14)). Then, the second best rate is considered, and the associated user-UAV pair are connected. This process is repeated until all users are either associated or cannot be provided with satisfying rates (i.e. constraint (6c) cannot be satisfied for unassociated users), or all UAVs reach their maximum quota. At each configuration, the algorithm compares with the previous configurations (line (15)). If the selected configuration provides a better sum-rate, then the best configuration is updated (line (17)). The process is repeated until all configurations are tested.

B. Adapted Version of the Greedy

Algorithm 4 Adapted greedy algorithm

```
1:  Initialization:
2:  Sort the UAVs in a decreasing order according to their
      maximum quota, let Ĵ be the set of ordered UAVs
3:  q_ij = 0, ∀i ∈ ℐ, j ∈ 𝒥
4:  ℒ = ℐ, the set of not associated users
5:  for j ∈ Ĵ do
6:      Sort the non-associated users according to their decreas-
      ing rates order (rates provided by UAV j)
7:      Try all available 3D locations for the UAV and save
      the one that maximizes the sum-rate of the N_j best non-
      associated users. (Only interference from already placed
      UAVs is considered)
8:      Update the location of UAV j
9:      Associate UAV j with the N_j best non-associated users,
      from ℒ, for which the quality of service is satisfied
10:     Update ℒ by removing associated users
```

We are now interested in developing a fast algorithm that does not come with a guaranteed performance but provides very good results in practice. We refer to this algorithm as *the adapted version of the greedy* or simply *adapted greedy*.

In Algorithm 4, we first sort the UAVs in decreasing order according to their maximum quota (line (6)). The first UAV selects -among all the possible locations of the 3D grid- the location that provides the best sum-rate for the best N_j users' rates (line (7)). The N_j users with the best rates are therefore associated with the UAV (line (9)), and their association is never reconsidered in the next steps of the algorithm (line (10)). Then, the process is repeated for the remaining UAVs and users. The process ends when the UAV with the minimum quota has been associated with its users.

VII. BLLL vs. Greedy: a Fair Comparison

In this section, we compare the previously proposed approaches in terms of convergence rate, computational complexity, memory requirement, and exchanged information.

A. Convergence Time

The BLLL approach allows us to adopt an action with a certain probability. This probability is dependent on the utility of the action and the temperature parameter. The higher the utility, the higher the probability to select the action. Initially, the temperature is set to a high value in order to allow a wide exploration of the search space. As the number of iterations increases, the temperature is cooled down -reduced- in order to eliminate unsuccessful strategies. Clearly, the convergence rate of the BLLL depends on two main parameters: the initial temperature and the cooling scheme of the temperature. It has been shown in [46] that the logarithmic scheme is one of the most efficient temperature decays. This scheme suggests that at each iteration t, the temperature is given by $T(t) = \frac{T_0}{log(1+t)}$, where T_0 is the initial temperature. Although such a cooling approach allows a very slow decrease of the temperature, it ensures the convergence to the global optimum when enough iterations are provided. It is also important to note that when the initial temperature is too low, the search space will be reduced, and the algorithm can get trapped in a local optimum. One guideline is to tune the initial temperature based on the first realizations of the utility function, or to set the initial temperature to a high value.

It is important to note that there are many parameters to fine-tune the performance of BLLL. For example, the performance of BLLL is sensitive to the initial temperature and the choice of the cooling scheme as highlighted in [46]. Furthermore, BLLL can take exponentially long to converge unless we modify it using, for example, the approach in [47]. All these details are important but are not the focus of our work since BLLL is proposed here mainly as an optimal benchmark to assess the performance of both the greedy and the adapted greedy algorithms.

The greedy algorithm also requires a large number of iterations, especially if the search space is not reduced. This is because it has to go through all the possible configurations of the network. However, when we remove configurations that are unlikely to be efficient, the convergence time is significantly reduced. In general, the greedy algorithm loops over $\mathcal{O}(K \times I \times J)$ iterations, where K is the number of possible configurations. At every single iteration, the algorithm looks for the unconnected user-UAV pair with the maximum rate, which requires at most $\mathcal{O}(I \times J)$ iterations. Therefore, the run-time complexity of the greedy algorithm can be written as $\mathcal{O}(K \times (I \times J)^2)$.

On the other hand, the adapted greedy first sorts the UAVs in decreasing order according to their maximum quota which requires $\mathcal{O}(J^2)$ iterations. At each iteration j, $\mathcal{O}(I^2)$ iterations are needed to sort the users according to their rates. Moreover, the algorithm needs an additional number of iterations, which are equal to the number of available 3D locations, to find the best 3D position that maximizes the sum-rate of the considered UAV. Hence, assuming G is the number of 3D positions in the 3D grid. The number of iterations needed for the adapted greedy to converge is: $\mathcal{O}(J^2) + \mathcal{O}(J \times (I^2 + G))$. Assuming that the number of users is larger than the number of UAVs, the time complexity convergence can be written as $\mathcal{O}(I^3)$.

Network Size	BLLL	Greedy	Adapted Greedy
5 UAVs and 45 users	125 s	105.6 s	1.28 s
4 UAVs and 25 users	50.38 s	6.48 s	0.63 s
3 UAVs and 15 users	26.11 s	0.66 s	0.5 s
Theoretical Performance	Optimal	$1 - 1/e$	–
Averaged Performance in % over the 3 scenarios	100%	78%	84%

TABLE II: Performance of proposed approaches in seconds.

Parameter	Value	Parameter	Value
Area	1000×1000	δ_x	10m
δ_y	10m	δ_h	10m
$h^{\min}$	100m	$h^{\max}$	200m
$\eta^{\min}$	-3 dB	I	45
J	5	P_j	10 dBm
α	9.61	β	0.16
c	$3.10^8 m/s$	ζ_{LoS}	1 dB
ζ_{NLoS}	20dB	N_j	4

TABLE III: Simulation settings.

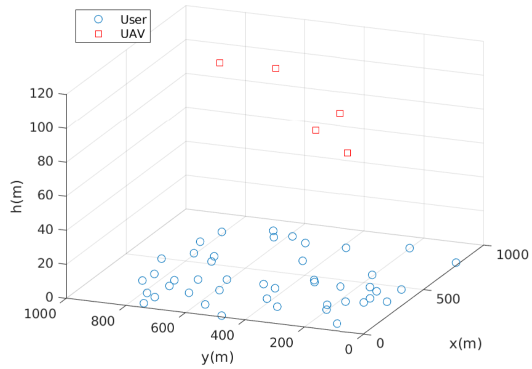
In Table II, the convergence time of three scenarios of users and UAVs is depicted. The UAVs are assumed to associate with at most 4 users for all the proposed scenarios. As it can be seen from the table, BLLL takes the longest time to converge, followed by the greedy algorithm and then the adapted greedy approach. Although BLLL takes the longest time to converge, it always achieves the best performance. Also, even though the adapted greedy algorithm is a heuristic, it still achieves very good performance for the studied scenarios. We highlight here a case where it achieves even better performance (in terms of sum-rate) than the greedy algorithm. This is generally not the case as highlighted in Fig. 5 in the simulation results in Section VIII. Finally, and as expected, the performance of the greedy is better than the lower bound $1 - 1/e$.

B. Computational Complexity and Memory Requirement

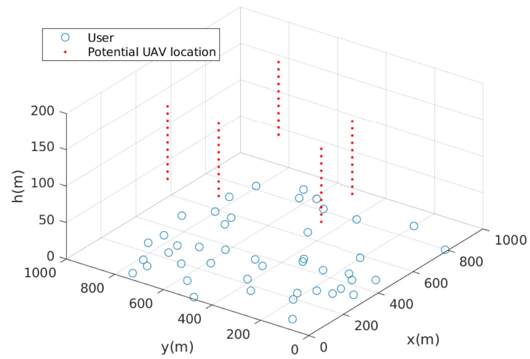
From a computational perspective, the UAVs perform simple algebraic operations when they adhere to the BLLL. Essentially, the active UAV, as well as the other UAVs, need to observe the impact of the action on the throughput of their users. Then, each UAV has to compute and broadcast its aggregated throughput (i.e., local sum-rate of its served users) to the active UAV. Also, the UAVs have only to memorize the utility of their previous action, leading to very low memory requirements.

Similarly, the greedy algorithm does not require computational complexity as it only computes the rates of the users at various UAVs locations. However, it requires high memory storage capacity as it compares rates at different heights.

On the other hand, the adapted greedy approach requires more computational efforts as every UAV has to solve a local optimization problem. In particular, the first UAV has to select, among all the possible locations in the 3D gird, the one that maximizes its local sum-rate. Similarly, the second UAV



(a)



(b)

Fig. 2: (a) Initial configuration (b) Potential locations of UAVs using k-means.

chooses from the remaining locations the one that maximizes its aggregated throughput. This process is repeated for all the UAVs, one by one, to select from the remaining locations the one that improves their local sum-rates. At the same time, the algorithm does not require significant memory storage.

C. Exchanged Information

Based on its formulation in equation (8), the UAVs' utility relies on global and complete information of the network when BLLL is adopted. Indeed, in order to fit into the potential game framework, the utility is designed as the marginal contribution of the player (i.e., UAV). This implies that each UAV has to compute the sum-rate of all associated users when this UAV is part of the game and when it is not. Clearly, significant knowledge is required. Not only does the UAV need to know the throughput of its served users at its selected 3D location, but also knowledge is needed of the throughput of users that are connected to all other UAVs. This will entail a considerable amount of exchanged information in the network. The convergence of the BLLL to the global optimum comes at the expense of complete network knowledge. In the simulation results section, we consider a version of the BLLL which uses only local neighborhood information versus global information. Our results show that the performance of this version of BLLL deteriorates compared to the BLLL with a global knowledge of the network.

Compared to BLLL, the greedy algorithm implementation is centralized. This also suggests a high information exchange between users, UAVs, and the centralized entity. Instead, the adapted version of the greedy involves much less information exchange. At each iteration of the algorithm, the UAV needs only to observe the throughput of its served users. No information is required from the previously deployed UAVs.

VIII. Simulation Results

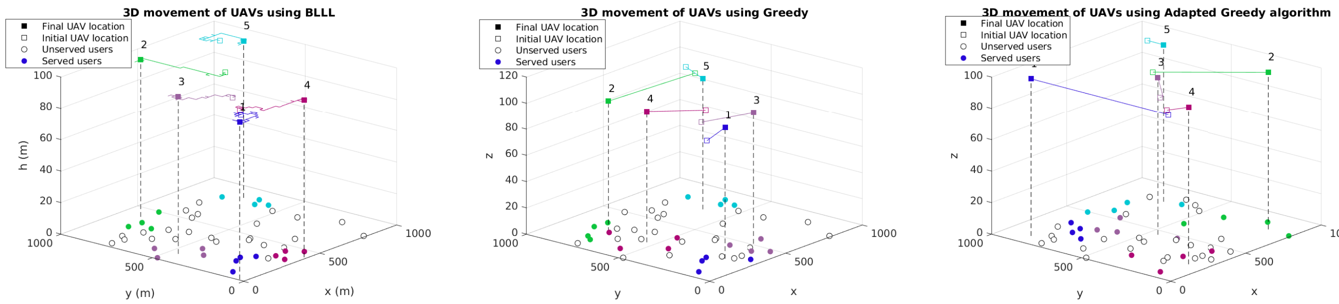
To assess the performance of the studied algorithms, we consider the following scenario. We assume 45 users, uniformly distributed in an area of $1000 \times 1000 m^2$. 5 UAVs are considered to provide connectivity to the ground users. The UAVs positions are initially set to a random position as shown in Fig 2(a). All the drones are assumed to transmit with the same power $P = 10$ dBm. In order to account for the path loss, we assume $\zeta_{\mathrm{LoS}} = 1$ dB, $\zeta_{\mathrm{NLoS}} = 20$ dB, $\alpha = 9.61$, $\beta = 0.16$, $f_c = 2$ GHz, and $c = 3 * 10^8 m/s$. The simulation settings are summarized in TABLE III.

Fig. 2(a) plots the initial 3D locations of UAVs for the studied scenario. In Fig. 2(b), we show the selected 3D positions after the reduction of the search space using k-means.

In Fig. 3, we plot the 3D movement of the UAVs under the studied algorithms setup. We see from Fig. 3(a), that the UAVs move sequentially in the 3D space before reaching their final 3D locations. Each UAV finds its best location in order to cover the maximum number of users allowed by its quota. The heights and 2D coordinates of UAVs are adjusted in order to reduce interference and ensure the best network sum-rate. In Fig. 3(b), we notice that each UAV has only to move once in order to reach its final location. This is because the greedy algorithm will not allow the UAVs to move unless a better location is found. In the studied scenario, the best locations for UAVs were found in the second iteration. The adapted version of the greedy, plotted in Fig. 3(c), allows one UAV movement at a time. The UAVs are moved one by one to the 3D location that maximizes their aggregate sum-rate.

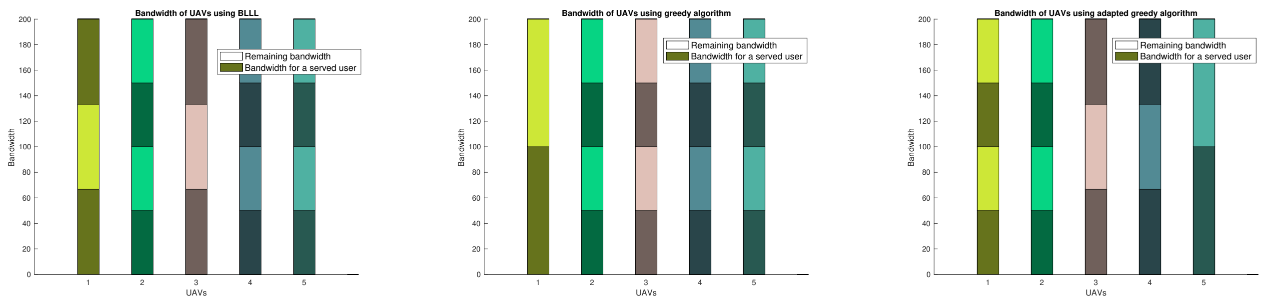
In Fig. 4, we plot the bandwidth allocation when the total maximum quota exceeds the number of users. It can be seen from Fig. 4(b) that the bandwidth allocated to each one of the users associated with UAV1 is higher than the bandwidth allocated by other UAVs to their served users. Indeed, once UAV1 has selected its associated users and noticed that it still has an additional free bandwidth, it equally divides its total amount of bandwidth among its users in order to improve the global sum-rate. The same conclusions can be made for other UAVs using a similar mechanism in Fig. 4(a) and (c).

Fig. 5 plots the network sum-rate vs. the number of iterations. As can be seen from Fig. 5, although BLLL requires the highest number of iterations to converge, it still provides the best performance. On the other side, less performance is achieved



(a) (b) (c)

Fig. 3: 3D movements of UAVs under (a) BLLL, (b) greedy, and (c) adapted greedy algorithms.



(a) (b) (c)

Fig. 4: Bandwidth allocation for the studied approaches. Size of the network: 5 UAVs with a maximum quota of 4 for each UAV, and 18 users.

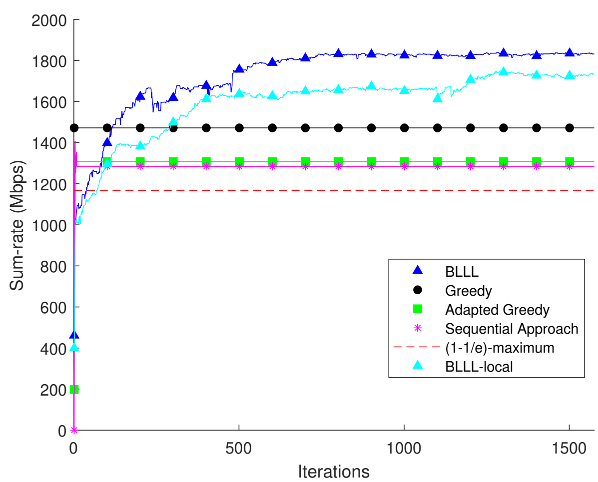


Fig. 5: Sum-rate convergence of BLLL, greedy, adapted greedy, and BLLL-local algorithms, and a suboptimal approach that solves the problem sequentially. Here, $\theta = 82.5$ dBm for BLLL-local algorithm.

when the greedy algorithm is adopted. However, only a few iterations are needed to reach an efficient value of the network sum-rate. Even fewer iterations are needed for the adapted version of the greedy algorithm in order to ensure convergence. The adapted greedy, however, achieves the lowest performance compared to the greedy and BLLL approaches. We note that the results provided by all approaches are better than the minimum 1-1/e.

To assess the proposed approaches, we compare our results to an iterative approach similar to the one proposed in [19]. First, we solve the association problem as an integer-linear-program, and then we address the 3D locations problem as a continuous optimization using the interior-point algorithm. As can be seen from Fig. 5, the convergence times of our greedy algorithms are comparable with the sequential approach of [19]. However, we obtain a better sum-rate using the greedy approaches.

As mentioned in subsection VII-C, the convergence of BLLL comes at the expense of complete knowledge of the network. The formulation of equation (8) assumes that each UAV knows the rates of all the users in the network. In this section, we consider a version of BLLL where only local information is required. This version is referred to as *BLLL-local*. BLLL-local requires the rates of users that are associated with neighboring UAVs only. The utility of UAVs for this version can be written as follows:

$$U_j(\mathbf{x},\mathbf{y},\mathbf{h},\mathbf{q},\mathcal{N}_j)=\sum_{j'\in\mathcal{N}_j}\sum_{i\in\mathcal{I}} q_{ij'}R_{ij'}-\sum_{j'\in\mathcal{N}_j\setminus\{j\}}\sum_{i\in\mathcal{I}} q_{ij'}R_{ij'}(-j), \tag{20}$$

where $\mathcal{N}_j$ is the set of UAVs neighbors to UAV j. N_j is given by:

$$\mathcal{N}_j=\{l\in\mathcal{J}, \exists i\in\mathcal{I} \text{ where } P_jL_{ij}(r_{ij},d_{ij})>\theta \text{ and } P_lL_{il}(r_{il},d_{il})>\theta\}, \tag{21}$$

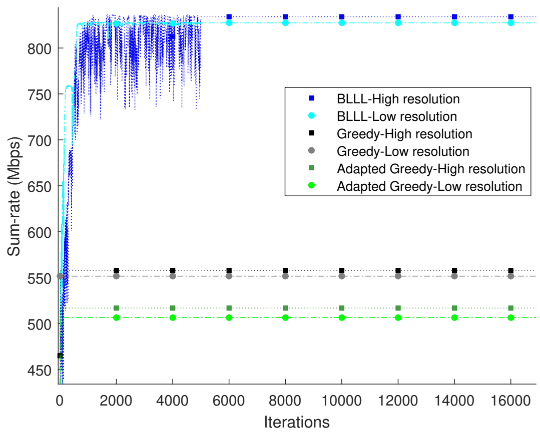


Fig. 6: Grid effect. In the legend, "low resolution" describes a grid with 10 meters granularity, whereas, "high resolution" corresponds to a 5 meters 3D grid step granularity. A setup of 5 UAVs and 45 users in 450m×450m is considered.

with θ a received power threshold. In other words, UAV l is neighbor to UAV j if there exists a common user that receives a power higher than θ from both UAVs. The neighborhood relation is symmetrical (i.e., if UAV l is neighbor to UAV j, UAV j is also neighbor to UAV l). When $\theta = 0$, all UAVs are neighbors. In that case, we restore the original BLLL setup. In particular, the UAV utilities described by equation (8) assume that all UAVs are neighbors. In Fig. 5, we plot the performance of BLLL-local against the number of iterations. Here θ is equal to 82.5 dBm. In that studied case, each UAV has, on average, 3 neighboring UAVs. We see from the figure that BLLL-local performance is significantly degraded as compared to BLLL with complete knowledge of the network. Full information is crucial to the performance of BLLL. This is because interference at distant users (i.e., users associated with non-neighboring UAVs) is not considered. Each UAV adjusts its 3D location to improve the rates of its associated users and users associated with its neighbors, ignoring any possible deterioration at faraway users.

Finally, we study the impact of the grid on the performance of our algorithms. The use of BLLL requires a finite set of strategies as mentioned in [31]. This is why we discretize the 3D space to a 3D grid to allow users to choose their actions from a discrete and finite strategy set. The step granularity of the grid determines the number of possible actions. In particular, the finer the grid, the more accurate the results. However, a small granularity step results in a higher number of actions, which leads to a relatively large time of convergence. The same applies to greedy approaches as more time is needed to find out the best 3D locations. In Fig. 6 below, we show the effect of the grid on the convergence of the studied approaches. As it can be seen from the figure, when the granularity step is small, slightly higher sum-rates are achieved. It is also clear from the figure that the BLLL converges earlier when a large 3D grid step (low resolution setup) is considered.

IX. Conclusion

In this paper, we addressed the problem of joint 3D placement and users association in UAVs-enabled networks. We proposed three algorithms. The first is guaranteed to reach the global optimum of the sum-rate function at the expense of exponential convergence time. The second exploits the submodularity of the studied problem and has a performance guarantee of $1 - 1/e$. The third is a heuristic which requires only few iterations. While having no guaranteed performance, this heuristic achieves very good results in simulations.

References

[1] www.google.com/loon Federation of aviation authority (FAA), "Unmanned aircraft systems," 2019.

[2] N .H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–27, 2016.

[3] B. Li, Z. Fei, and Y. Zhang, "UAV communications for 5G and beyond: Recent advances and future trends," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2241–2263, 2018.

[4] 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.

[5] 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, 2019.

[6] C. Yan, L. Fu, J. Zhang, and J. Wang, "A comprehensive survey on UAV communication channel modeling," *IEEE Access*, vol. 7, pp. 107769–107792, 2019.

[7] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *arXiv preprint arXiv:1803.00680*, 2018.

[8] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2016.

[9] M. E. Mkiramweni, C. Yang, J. Li, and W. Zhang, "A survey of game theory in unmanned aerial vehicles communications," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3386–3416, 2019.

[10] E. Bulut and I. Güvençl, "Trajectory optimization for cellular-connected UAVs with disconnectivity constraint," in *IEEE International Conference on Communications (ICC)*, Kensas,USA, May 2018.

[11] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, 2017.

[12] A A. Khuwaja, Y. Chen, N. Zhao, M.-S. Alouini, and P. Dobbins, "A survey of channel modeling for UAV communications," *IEEE Communications Surveys Tutorials*, vol. 20, no. 4, pp. 2804–2821, 2018.

[13] V. V. C. Ravi and H. Dhillon, "Downlink coverage probability in a finite network of unmanned aerial vehicle (UAV) base stations," in *IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Edinburgh, United Kingdom, Jul. 2016, pp. 1–5.

[14] A. M. Hayajneh, S. A. R. Zaidi, D. C. McLernon, and M. Ghogho, "Optimal dimensioning and performance analysis of drone-based wireless communications," in *IEEE Global communications conference workshops (GLOBECOM)*, Washington DC, USA, Dec. 2016.

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

[16] 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.

[17] H. Shakhatreh, A. Khreishah, A. Alsarhan, I. Khalil, A. Sawalmeh, and N. S. Othman, "Efficient 3D placement of a UAV using particle swarm optimization," in *2017 8th International Conference on Information and Communication Systems (ICICS)*. IEEE, 2017, pp. 258–263.

[18] 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.

[19] 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 *International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Montreal, QC, Canada, Sep. 2017, pp. 1–6.

[20] M. J. Farooq and Q. Zhu, "A multi-layer feedback system approach to resilient connectivity of remotely deployed mobile Internet of Things," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 2, pp. 422–432, 2018.

[21] 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.

[22] H. El Hammouti, M. Benjillali, B. Shihada, and M.-S. Alouini, "Learn-as-you-fly: A distributed algorithm for joint 3D placement and user association in multi-uavs networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 12, pp. 5831–5844, 2019.

[23] B. Shang and L. Liu, "Mobile edge computing in the sky: Energy optimization for air-ground integrated networks," *IEEE Internet of Things Journal*, 2020.

[24] Y. Liu, K. Liu, J. Han, L. Zhu, Z. Xiao, and X. Xia, "Resource allocation and 3D placement for uav-enabled energy-efficient IoT communications," *IEEE Internet of Things Journal*, 2020.

[25] C. Zou, X. Li, X. Liu, and M. Zhang, "3d placement of unmanned aerial vehicles and partially overlapped channel assignment for throughput maximization," *Elsevier Digital Communications and Networks*, 2020.

[26] C. Pan, C. Yin, N. C Beaulieu, and J. Yu, "3d UAV placement and user association in software-defined cellular networks," *Springer Wireless Networks*, vol. 25, no. 7, pp. 3883–3897, 2019.

[27] Y. Wang, Z. Ru, K. Wang, and P. Huang, "Joint deployment and task scheduling optimization for large-scale mobile users in multi-UAV-enabled mobile edge computing," *IEEE transactions on cybernetics*, vol. 50, no. 9, pp. 3984–3997, 2019.

[28] G. J Woeginge, "Exact algorithms for NP-hard problems: A survey," in *Combinatorial Optimization-Eureka, You Shrink!*, pp. 185–207. 2003.

[29] I-H. Hou and P. Gupta, "Distributed resource allocation for proportional fairness in multi-band wireless systems," in *2011 IEEE International Symposium on Information Theory Proceedings*. IEEE, 2011, pp. 1975–1979.

[30] S. C Borst, M. G Markakis, and I. Saniee, "Nonconcave utility maximization in locally coupled systems, with applications to wireless and wireline networks," *IEEE/ACM Transactions on Networking*, vol. 22, no. 2, pp. 674–687, 2013.

[31] J. R Marden and J. S Shamma, "Revisiting log-linear learning: Asynchrony, completeness and payoff-based implementation," *Games and Economic Behavior*, vol. 75, no. 2, pp. 788–808, 2012.

[32] Y. Xu, J. Wang, and Q. Wu, "Distributed learning of equilibria with incomplete, dynamic, and uncertain information in wireless communication networks," in *Game Theory Framework Applied to Wireless Communication Networks*, pp. 63–86. IGI Global, 2016.

[33] H. Dai, Y. Huang, Y. Xu, C. Li, B. Wang, and L. Yang, "Energy-efficient resource allocation for energy harvesting-based device-to-device communication," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 509–524, 2018.

[34] M. Zhu and S. Martínez, "Distributed coverage games for energy-aware mobile sensor networks," *SIAM Journal on Control and Optimization*, vol. 51, no. 1, pp. 1–27, 2013.

[35] G. Su, B. Chen, X. Lin, H. Wang, and L. Li, "A submodular optimization framework for outage-aware cell association in heterogeneous cellular networks," *Mathematical Problems in Engineering*, 2016.

[36] E. Lakiotakis, P. Sermpezis, and X. Dimitropoulos, "Joint optimization of UAV placement and caching under battery constraints in UAV-aided small-cell networks," in *Proceedings of the ACM SIGCOMM 2019 Workshop on Mobile AirGround Edge Computing, Systems, Networks, and Applications*, 2019, pp. 8–14.

[37] J. Wang, H. Zhu, L. Dai, N. J Gomes, and J. Wang, "Low-complexity beam allocation for switched-beam based multiuser massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8236–8248, 2016.

[38] S. Shah, T. Khattab, M. S. Zeeshan, and M. Hasna, "A distributed approach for networked flying platform association with small cells in 5G+ networks," in *IEEE Global Communications Conference (GLOBECOM)*, Singapore, Singapore, Dec. 2017, pp. 1–7.

[39] E. Chaalal, L. Reynaud, and S. S. Senouci, "A social spider optimisation algorithm for 3d unmanned aerial base stations placement," in *2020 IFIP Networking Conference (Networking)*. IEEE, 2020, pp. 544–548.

[40] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," in *IEEE Global Communications Conference (GLOBECOM)*, Austin, USA, Dec. 2014, pp. 2898–2904.

[41] D. Monderer and L. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, no. 1, pp. 124–143, 1996.

[42] G. Su, B. Chen, X. Lin, and H. Wang L. Li, "A submodular optimization framework for outage-aware cell association in heterogeneous cellular networks," *Mathematical Problems in Engineering*, 2016.

[43] G. L. Nemhauser, L. A Wolsey, and M. L Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical programming*, vol. 14, no. 1, pp. 265–294, 1978.

[44] J. Edmonds, "Submodular functions, matroids, and certain polyhedra," in *Combinatorial Optimization—Eureka, You Shrink!*, pp. 11–26. Springer, 2003.

[45] P. Sermpezis, T. Spyropoulos, L. Vigneri, and T. Giannakas, "Femtocaching with soft cache hits: Improving performance with related content recommendation," in *IEEE Global Communications Conference (GLOBECOM)*, Dec 2017, pp. 1–7.

[46] M. J Brusco, "A comparison of simulated annealing algorithms for variable selection in principal component analysis and discriminant analysis," *Computational Statistics & Data Analysis*, vol. 77, pp. 38–53, 2014.

[47] D. Shah and J. Shin, "Dynamics in congestion games," *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 1, pp. 107–118, 2010.