Joint Optimization of UAV Placement and Caching under Battery Constraints in UAV-aided Small-Cell Networks

Emmanouil Lakiotakis
University of Crete and FORTH,
Greece
manoslak@ics.forth.gr

Pavlos Sermpezis
FORTH,
Greece
sermpezis@ics.forth.gr

Xenofontas Dimitropoulos
University of Crete and FORTH,
Greece
fontas@ics.forth.gr

ABSTRACT

Unmanned Aerial Vehicles (UAVs), or commonly known as drones, have recently attracted the interest of the wireless communications community. Owing to their agility and fast deployment, UAVs can act as flying base stations and adjust their locations based on users mobility in order to provide higher network coverage and service. The network performance can be further improved by using cache-equipped UAVs, i.e., by combining wireless connectivity with mobile edge caching principles. In this context, this paper studies how to allocate the UAV resources in a mobile network with UAVs, and specifically, how to optimize the cache hit ratio (CHR) by selecting the placement of UAVs, the filling of their caches, and the UAV-user associations. We show that the joint UAV placement and caching problem is NP-hard, and propose an approximation algorithm with performance guarantees. In the case of UAVs with energy constraints, we propose a heuristic algorithm that takes also into account the battery levels and UAV-user association. Our evaluation shows that the proposed algorithms outperform previous approaches that consider the problems of UAV placement and caching separately, or do not consider the energy constraints.

CCS CONCEPTS

• **Networks** → *Mobile networks*; *Wireless local area networks*; *Ad hoc networks*;

KEYWORDS

Unmanned Aerial Vehicle (UAV), wireless communications, optimization, deployment, caching, energy efficiency

ACM Reference Format:
Emmanouil Lakiotakis, Pavlos Sermpezis, and Xenofontas Dimitropoulos. 2019. Joint Optimization of UAV Placement and Caching under Battery Constraints in UAV-aided Small-Cell Networks . In *ACM SIGCOMM 2019 Workshop on Mobile AirÂŋGround Edge Computing, Systems, Networks, and Applications (MAGESys'19), August 19, 2019, Beijing, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3341568.3342106

MAGESys'19, August 19, 2019, Beijing, China

ACM ISBN 978-1-4503-6879-7/19/08…$15.00
https://doi.org/10.1145/3341568.3342106

1 INTRODUCTION

Caching in small-cell (SC) base stations has been recently proposed and attracted a lot of attention for 5G cellular networks [5, 12, 13]. Caching contents in small-cells can increase the network capacity, improve the QoS, and alleviate congested links during peak hours. Moreover, hybrid cellular-UAVs architectures, with UAVs acting as SCs, have been also proposed for cellular networks [20, 21, 30].

A network architecture that takes into account both approaches, i.e., caching in small-cell UAVs, has been recently considered as a promising solution for enhancing the performance of 5G networks [3, 4, 22, 27–29, 31]. The main differences between SCs in ground networks and using UAVs as SCs, are (i) the flexibility in choosing (and changing) the position of the UAVs, which is an advantage compared to static SCs typically considered in ground networks, and (ii) the energy constraints of UAVs, which might constrain their operation time (flying, wireless transmissions, etc.).

In this paper, we explicitly take into account these special characteristics, and study the problem of resource allocation in settings where UAVs are used as SCs with caches. Specifically, we propose caching strategies, by leveraging the advantages of UAVs compared to static SCs (i.e., flexibility in UAV placement), and taking care about their limited energy constraints.

Caching in SC base stations has attracted a lot of interest, and optimal resource allocation has been studied [8, 23]. However, SCs base stations are static. On the contrary, UAVs have the flexibility to move, and thus can be placed in the optimal locations, where they better serve users. Works that considered UAVs as SC, include [1, 2, 18, 19, 25, 26]. The works of [18, 25, 26] examine the minimum required number and locations of UAVs that provide total network coverage in parallel with terrestrial base stations. Their objective is to place UAVs in locations that can maximize the number of users that they can serve, improving total network coverage.

However, a strategy that maximizes the number of covered users and/or their throughput or provide the required QoE [6], will not necessarily maximize the benefits for the network as well. For instance, the set of covered users may not generate a high user demand, or may request a set of contents that not all of them can be cached; this would lead to low *cache hit rates*, and sub-optimal offloading of the main network.

To this end, we first consider the joint problem of UAV placement and caching, by allowing overlaps in the coverage areas of UAVs (i.e., in analogy to the femto-caching architecture of [9, 10, 24]). Jointly optimizing the UAV placement and caching, can lead to better performance than considering these problems separately (e.g., first select the locations of the UAVs and then select the contents to be cached at each UAV). Then, we extend our approach by taking into account also the second major characteristic of UAVs (in comparison to ground SCs), i.e., their limited battery. While UAV energy constraints have been considered in [11, 14, 27], they have not been taken into account jointly with the caching policy.

Summarizing, the contributions of this paper are:

- We formulate the problem of optimal *joint UAV placement and caching*. We show that the problem is NP-hard, and propose a greedy algorithm that achieves at least $\frac{1}{2}$ of the optimal solution.
- We extend the analysis for the case of UAVs with limited battery. In this more generic case, the association of user requests to available UAVs becomes crucial. We formulate the optimization problem of *joint UAV placement, caching, and user association*, and propose efficient algorithms.
- We evaluate the proposed algorithms in simulation scenarios, and show that optimizing jointly the UAV placement and caching (and user association), outperforms state-of-the art approaches.

The following section (Section 2), presents the system model we consider in our analysis. Next, we formulate, analyze and propose algorithms for the optimization problems of joint UAV placement and caching without (Section 3) and with (Section 4) battery constraints. Performance evaluation is presented in Section 5, and, finally, we conclude the paper in Section 6.

2 SYSTEM MODEL

Network Model. We consider a set of users[1] $\mathcal{N}$ ($\|\mathcal{N}\| = N$) residing in an area $\mathcal{A}$ at known locations. A set of UAVs $\mathcal{M}$ ($\|\mathcal{M}\| = M$) can hover or fly above this area. UAVs are equipped with wireless communication interfaces and storage capacity $C_m, m \in \mathcal{M}$, can store contents in their caches, and serve requests (for these contents) of users within their transmission range.

The above setting can capture various use cases of UAV networking: the area $\mathcal{A}$ might be covered by a ground cellular network, where the UAVs act complementary to the base stations (e.g., as small cells), or users are not connected at all to any ground network (rural areas, failure of cellular infrastructure, etc.). In the former scenario, UAVs are used to offload traffic from the cellular network, while in the latter scenario, users can be served only by UAVs.

Traffic Model. We consider a set (catalogue) of contents $\mathcal{K}$ ($\|\mathcal{K}\| = K$); for simplicity, we assume contents of equal size 1. In most practical settings the content catalogue is much larger than the storage capacity of UAVs, i.e., $C_m \ll K$, $\forall m \in \mathcal{M}$. Users' requests for contents are given by independent random processes with density (mean value) $p_{n,k}$, $\forall n \in \mathcal{N}, k \in \mathcal{K}$. We denote as $\mathcal{P}^{N\times K}$ the corresponding traffic density matrix.

PHY Model. The coverage area of a UAV m, $\mathcal{A}_m \subseteq \mathcal{A}$, depends on the position (latitude, longitude, elevation) of the UAV, and the wireless communication channel and interface settings. We denote with the following quantity whether a user can be served by a UAV (i.e., the user lies within the UAV's coverage area)

$$z_{q,n} = \begin{cases} 1, & \text{user } n \text{ is covered by a UAV located in position } q \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

[1]In a different (but equivalent) definition, a "user" can correspond to user density in a given spot/area.

Table 1: Important Notation

$\mathcal{N}$	set of users ($\|\mathcal{N}\| = N$))
$\mathcal{M}$	set of UAVs ($\|\mathcal{M}\| = M$))
C_m	storage capacity of a UAV $m \in \mathcal{M}$
$\mathcal{K}$	set of contents ($\|\mathcal{K}\| = K$))
$\mathcal{P}^{N\times K} = \{p_{n,k}\}$	traffic density matrix mean value of content requests by user n for content k
$\mathcal{A}$ $A_m \in \mathcal{A}$	network ground area coverage area of a UAV $m \in \mathcal{M}$
$\mathcal{Q}$	set of distinct (possible) positions of UAVs ($\|\mathcal{Q}\| = Q$)
$Z^{Q\times N} = \{z_{q,n}\}$	position-user association indicator matrix ($z_{q,n} \in \{0,1\}$) $z_{q,n} = 1$: user n is in the coverage area of a UAV located in position q
$Y^{M\times Q} = \{y_{m,q}\}$	UAV position indicator matrix $y_{m,q} \in \{0,1\}$) $y_{m,q} = 1$: UAV m is located in position q
$X^{K\times M} = \{x_{k,m}\}$	caching indicator matrix ($x_{k,m} \in \{0,1\}$) $x_{k,m} = 1$: content k is cached in UAV m

We further define a set of distinct (possible) positions of UAVs $\mathcal{Q}$ ($\|\mathcal{Q}\| = Q$; in most practical settings $M \ll Q$). Finally, we assume line-of-sight communication between UAVs and the users within their transmission range, and thus ignore the interference impact.

For ease of reference, the important notation is summarized in Table 1.

3 JOINT UAV PLACEMENT AND CACHING

3.1 Problem Formulation

Let us first define the optimization variables of the UAV placement and cache allocation problem. We denote the (i) locations that the UAVs are placed, and (ii) the contents that are cached in UAVs, with the following quantities:

$$y_{m,q} = \begin{cases} 1, & \text{the UAV } m \in M \text{ is in position } q \in Q \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$x_{k,m} = \begin{cases} 1, & \text{the content } k \in K \text{ is cached in UAV } m \in M \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Our goal is to maximize the amount of traffic that is served by the UAVs. To this end, we define the *cache hit ratio, CHR*, as the percentage of content requests that are served by the UAVs. In the following result, we calculate the expression that gives the *CHR*.

Result 1 (Cache Hit Ratio). *The cache hit ratio is given by*

$$CHR = \sum_{n\in\mathcal{N}}\sum_{k\in\mathcal{K}} p_{k,n}\cdot\left(1-\prod_{m\in\mathcal{M}}\prod_{q\in\mathcal{Q}}\left(1-x_{k,m}\cdot y_{m,q}\cdot z_{q,n}\right)\right) \tag{4}$$

Proof. Whether a user $n \in N$ requesting a content $k \in K$, is served by a UAV $m \in M$ located in position $q \in Q$, is expressed by

$$CHR(n,k,m,q) = x_{k,m}\cdot y_{m,q}\cdot z_{q,n} \tag{5}$$

where $CHR(n,k,m,q) = 1$ only if (i) the content k is cached in UAV m ($x_{k,m} = 1$), *and* (ii) the UAV m is located in position q ($y_{m,q} = 1$), *and* (iii) the user n is within the coverage of location q ($z_{q,n} = 1$). If at least one of the above three conditions does not hold, then $CHR(n, k, m, q) = 0$.

Hence, whether a user $n \in N$ requesting for a content $k \in K$, is served by *any* UAV is given by

$$CHR(n, k) = 1 - \prod_{m \in \mathcal{M}} \prod_{q \in Q} \left(1 - x_{k,m} \cdot y_{m,q} \cdot z_{q,n}\right) \quad (6)$$

where $CHR(n, k) = 1$ *iff* the user n resides in the coverage area of at least one UAV that stores the content k; if no nearby –to user n– UAVs exist, or none of the nearby UAVs stores content k, then $CHR(n, k) = 0$.

Finally, using Eq. (6) and summing over all the users and content requests (using as weights the respective probabilities $p_{k,n}$ of user/content request), gives the expression of the Result 1. □

Having defined the optimization variables (Eq. (2) and Eq. (3)) and derived the expression for CHR (Result 1), which is the quantity we want to maximize, we can now formulate the optimization problem of joint UAV placement and caching selection.

Optimization Problem 1. *[Joint UAV placement and caching]*

$$max_{\overline{x},\overline{y}}\ CHR(\overline{x}, \overline{y})$$

subject to

$$\sum_{k \in \mathcal{K}} x_{k,m} = C_m,\ \forall m \in \mathcal{M} \text{ // cache capacity}$$

$$\sum_{q \in Q} y_{m,q} = 1,\ \forall m \in \mathcal{M} \text{ // UAV only in one location}$$

$$x_{k,m}, y_{m,q} \in \{0, 1\},\ \forall k \in \mathcal{K}, m \in \mathcal{M}, q \in Q$$

3.2 Greedy Algorithm

The Optimization Problem 1 is an integer 0-1 optimization problem, with two cardinality constraints. In the following Lemmas, whose detailed proofs are omitted due to space limitation, we prove important properties for Optimization Problem 1.

Lemma 1. *The Optimization Problem 1 is NP-hard.*

Lemma 2. *The Optimization Problem 1 has (i) a submodular and monotone increasing objective function, and (ii) a matroid constraint.*

The above Lemmas are proved by following a process similar to [24]. Specifically, Lemma 1 is proved by showing that a sub-case of the Optimization Problem 1 can be reduced to the *maximum coverage problem with weighted elements*, which is NP-hard [15]. To prove Lemma 2 we follow the standard steps (see, e.g., [16]) to show that the constraints form a matroid, and the objective function satisfies the necessary conditions for monotonicity and submodularity, i.e., (i) show that $\Delta CHR(\mathcal{S}, e) = CHR(\mathcal{S} \cup \{e\}) - CHR(\mathcal{S}) \geq 0$, and (ii) show that $\forall S \subseteq V$, $\Delta CHR(\mathcal{V}, e) - \Delta CHR(\mathcal{S}, e) \leq 0$.

Lemmas 1 and 2 lead to the following result, which states that a greedy algorithm for Optimization Problem 1 comes with guaranteed performance, i.e., the greedy solution is never less than $\frac{1}{2}$ of the optimal solution.

Lemma 3. *A greedy algorithm for the Optimization Problem 1 achieves an approximation ratio of $\frac{1}{2}$, i.e.,*

$$greedy\_solution \geq \frac{1}{2} \cdot optimal\_solution \quad (7)$$

Algorithm 1 Greedy Algorithm for Optimization Problem 1.

Input: *association array* $\{z_{k,n}\}$, *content demand* $\{p_{n,k}\}$, *combinations pool* $\{m, q, k\}, \forall m \in \mathcal{M},\ q \in Q,\ k \in \mathcal{K}$

1: $S_0 \leftarrow \emptyset;\ i \leftarrow 0$
2: **while** $i < M \cdot C$ **do**
3: $\quad i \leftarrow i + 1$
4: $\quad (m', q', k') \leftarrow \underset{\ell \in \{c_{m,q,k}\} \setminus S_{i-1}}{argmax} CHR(S_{i-1} \cup \{\ell\})$
5: $\quad S_i \leftarrow S_{i-1} \cup \{(m', q', k')\}$,
6: **end while**
7: $S^* \leftarrow S_i$
8: **return** S^*

Proof. For problems with submodular and monotone increasing objective function, subject to a matroid constraint, the greedy algorithm is guaranteed to achieve a $\frac{1}{2}$-approximation ratio [16]. □

The greedy algorithm. We design a greedy algorithm for selecting the UAV placement and cache allocation in Algorithm 1. To select the location and cache contents for each UAV, we start from UAVs with empty caches that are not already placed (line 1). Then, we start deciding the combination of UAV m, location q and content k (one by one) that increases the most the objective function (line 4), which expresses the *Cache Hit Ratio* in our case. This process is repeated till all M UAVs are located and contain C contents in their cache. The computation complexity of the algorithm is $O(M^2 \cdot K \cdot Q \cdot C)$, since the complexity of line 4 is $O(M \cdot K \cdot Q)$, and line 4 is executed $M \cdot C$ times.

Region Reduction The possible locations Q in area $\mathcal{A}$ for a UAV in the continuous space are infinite. However, for an efficient implementation of Algorithm 1, we need to have a discrete/finite space Q; the smaller the space, the faster the algorithm.

The simplest solution would be to use a grid. Instead, we use the following methodology, inspired by [7], which returns a set of points Q that (i) covers the entire area $\mathcal{A}$ of interest, i.e., where users are located, and (ii) any other point in $\mathcal{A}$ is at least as good as at least one point in the selected Q.

Specifically, we create an initial set Q that contains the points of (called "1-center locations"):

- the users positions
- the middle-points of each pair of users
- the circumcenter of each triplet of users

Then, we reduce the points in Q, by aggregating points based on their connectivity capability: (i) we merge points that provide connectivity to the same set of ground users, and (ii) we remove all points that cover a subset of users already covered by another point in Q.

As the number of ground users in the network increases, the 1-center locations set contains more candidate UAV positions. The reduction process described in the last step of the methodology is critical since it reduces the number of possible UAV locations, decreasing the greedy algorithm complexity and execution time.

4 JOINT UAV PLACEMENT AND CACHING UNDER ENERGY CONSTRAINTS

We now extend our framework to settings where UAVs have limited battery, and each cache hit consumes a (non-negligible) fraction of this battery. When the battery of a UAV is depleted, then it stops serving users, i.e., each request to this UAV is a cache miss.

Under the aforementioned setting, the cache hit ratio is not given by Result 1, and thus the greedy algorithm (cf. Lemma 3) becomes less efficient. In the following, we derive the cache hit ratio as a function of time (or, equivalently, number of requests) and UAVs' battery constraints, formulate the corresponding optimization problem, and propose efficient algorithms.

4.1 Battery-aware Model

UAV battery. Let us assume that each UAV is equipped with a battery to support its communication interfaces[2]. We denote the battery of a UAV $m \in \mathcal{M}$ as B_m. Moreover, we assume that a request for a content k served by a UAV, reduces its battery level by a quantity b_k.

User association. When a user request can be served by more than one UAVs (i.e., the user is in the transmission range of more than one UAVs storing the requested content), we now need to know which UAV will serve the request, since this will affect its battery. To this end, we define below the *user association.* A user may request contents from the same UAV always, or from a different UAV each time. To capture all possible scenarios, we define the user association in a probabilistic manner.

Definition 1. *The probability that, when a user $n \in \mathcal{N}$ requests a content $k \in \mathcal{K}$, she requests it from a UAV $m \in \mathcal{M}$, is $a_{n,k,m}$.*

We require the following conditions to hold for $a_{n,k,m}$, so that a user requests contents only from UAVs that are in range and store the content. First, $\forall n \in \mathcal{N}, k \in \mathcal{K}, m \in \mathcal{M}$:

$$a_{n,k,m} = \begin{cases} \in [0,1] & , x_{k,m} \cdot \sum_{q \in Q} y_{m,q} \cdot z_{q,n} = 1 \\ & \text{(i.e., if UAV } m \text{ stores the content } k \\ & \text{ and is in range with user } n\text{)} \\ 0 & , \text{otherwise} \end{cases} \tag{8}$$

and, second, $\forall n \in \mathcal{N}, k \in \mathcal{K}$:

$$\sum_{m \in \mathcal{M}} a_{n,k,m} = \begin{cases} 1 & , \sum_{m \in \mathcal{M}} x_{k,m} \cdot \sum_{q \in Q} y_{m,q} \cdot z_{q,n} > 0 \\ & \text{i.e., if at least one UAV } m \\ & \text{in range with user } n\text{, stores} \\ & \text{the content } k\text{)} \\ 0 & , \text{otherwise} \end{cases} \tag{9}$$

4.2 Problem Formulation

We are interested in optimizing the network for operating during a time interval $[0, t]$. This can be the total duration of the scenario we are interested in, or a period at the end of which we re-allocate the resources. At the begginning of each interval, we select the locations of UAVs (variables $y_{m,q}$), their caches (variables $x_{k,m}$), and the user association (variables $a_{n,k,m}$).

In the following Result, we derive the expression that gives the CHR in the scenario of UAVs with limited energy resources.

Result 2 (Cache Hit Ratio - Limited Battery). *The cache hit ratio (CHR) in a time interval of length t under UAV battery constraints is given by*

$$CHR(t) = \frac{1}{\sum_{n \in \mathcal{N}} \sum_{k \in \mathcal{K}} p_{n,k} \cdot t} \cdot \sum_{m \in \mathcal{M}} \min \Big\{ B_m \,, \sum_{n \in \mathcal{N}} \sum_{k \in \mathcal{K}} \sum_{q \in Q} p_{n,k} \cdot b_k \cdot a_{n,k,m} \cdot x_{k,m} \cdot y_{m,q} \cdot z_{q,n} \cdot t \Big\}$$

Proof. Whether a user $n \in N$ requesting a content $k \in K$ can be served by a UAV $m \in M$ located in position $q \in Q$, can be indicated by the quantity $x_{k,m} \cdot y_{m,q} \cdot z_{q,n}$.

Each user n has $p_{n,k}$ requests per time unit t for content k, and the fraction of these requests that are addressed to UAV m is $a_{n,k,m}$. Hence, the number of times that a user n requests a content k during time $[0, t]$ from a UAV m is given by $p_{n,k} \cdot t \cdot a_{n,k,m}$.

In total, the content requests of a user n for a content k from a UAV m that can be served during a time interval of duration t are $p_{n,k} \cdot t \cdot a_{n,k,m} \cdot x_{k,m} \cdot y_{m,q} \cdot z_{q,n}$, and the total requests to a UAV m are

$$\sum_{n \in \mathcal{N}} \sum_{k \in \mathcal{K}} \sum_{q \in Q} p_{n,k} \cdot t \cdot a_{n,k,m} \cdot x_{k,m} \cdot y_{m,q} \cdot z_{q,n}$$

Finally, serving a content k reduces the battery of a UAV by b_k, and each UAV can serve requests that sum up to B_m at most. Extra requests are dropped. Hence, taking this into account, and considering all UAVs in the network, and normalizing with the total number of requests, gives the expression of Result 2. □

Having derived the CHR, which is the quantity we want to maximize, we can proceed to formulate the optimization problem.

Optimization Problem 2. *[Joint UAV placement, caching, and user association - Limited battery]*

$$\begin{aligned} & max_{\overline{x},\overline{y},\overline{a}} \; CHR(t, \overline{x}, \overline{y}, \overline{a}) \\ & subject\ to \\ & \qquad \sum_{k \in \mathcal{K}} x_{k,m} = C_m, \; \forall m \in \mathcal{M} \; \textit{// cache capacity} \\ & \qquad \sum_{q \in Q} y_{m,q} = 1, \; \forall m \in \mathcal{M} \; \textit{// UAV only in one location} \\ & \qquad x_{k,m}, y_{m,q} \in \{0,1\}, \; \forall k \in \mathcal{K}, m \in \mathcal{M}, q \in Q \\ & \qquad Eq. (8), \; Eq. (9) \end{aligned}$$

The Optimization Problem 2 is a mixed integer programming problem (with $\overline{x}, \overline{y}$ discrete, and $\overline{a}$ continuous, variables), and is NP-hard. In the following, we propose a heuristic algorithm, which while is not coming with approximation guarantees, it efficiently allocates resources (i.e., low complexity) and achieves higher CHR than previous approaches (see Section 5).

[2]Of course, extra power is needed to support its movements, hovering, etc. This is orthogonal to our discussion, and we here we focus only on the part of the battery that is allocated to communication interfaces.

4.3 Heuristic Algorithm

Battery Greedy Algorithm. We propose a greedy algorithm, Algorithm 2, for Optimization Problem 2 that at each iteration i selects the tuple {content k, UAV m, location q}, which achieves the best CHR under the optimal user association decision, i.e.,

$$tuple(i) = \arg\max_{\mathcal{S}^i} CHR(t, \mathcal{S}^i, \overline{a}^*(i)) \tag{10}$$

where $\overline{a}^*(i) = \arg\max_{\overline{a}} CHR(t, \mathcal{S}^i, \overline{a})$ and $\mathcal{S}^i = \bigcup_{j=1}^{i} tuple(j)$.

Lemma 4. *The optimization problem* $\arg\max_{\overline{a}} CHR(t, \overline{a})$ *is convex.*

Proof. The function $CHR(t, \overline{a})$ is a sum of concave (i.e., min) functions of affine functions of $\overline{a}$, and the value space for $\overline{a}$ is subject to the set linear constraints of Eq. (8) and Eq. (9). This makes the problem convex. □

Fast Battery Greedy Algorithm. Algorithm 2 requires to solve at each iteration $O(K \cdot M \cdot Q)$ convex optimization problems: for the calculation of the $\overline{a}^*$ corresponding to each set $\mathcal{S}^{i-1} \cup tuple(i)$ for all candidate tuples $tuple(i)$. To reduce complexity, in the following, we also propose a faster version of this algorithm. The only modification that takes place is in line 7 of Algorithm 2.

Instead of calculating the optimal $\overline{a}^*(i)$ for each candidate set $\mathcal{S}^{i-1} \cup tuple(i)$, we use the optimal value of $\overline{a}^*(i-1)$ for the previous step, i.e., the set $\mathcal{S}^{i-1}$, (which is already calculated). We modify $\overline{a}^*(i-1)$ to take into account the newly added $tuple(i) = \{k', m', q'\}$ (note that $a^*_{n,k',m'}(i-1) = 0,\ \forall n$, since m' does not store k' at step $i-1$; see Definition 1) as follows:

$$a_{n,k,m}(i; k', m', q') = \begin{cases} 1 & \text{, if } n \in \mathcal{N}_{m'}, \\ & k = k', m = m' \\ 0 & \text{, if } n \in \mathcal{N}_{m'}, \\ & k = k', m \neq m' \\ a^*_{n,k,m}(i-1) & \text{, otherwise} \\ & \text{(i.e., if } n \notin \mathcal{N}_{m'}) \end{cases} \tag{11}$$

where $\mathcal{N}_{m'}$ is the users within the range of UAV m'

$$\mathcal{N}_{m'} = \left\{ n \in \mathcal{N} : y_{m',q'} \cdot z_{n,q'} = 1, \right\} \tag{12}$$

The above modification of $\overline{a}$ states that all the users in the range of m' will request the content k' from m' and not from any other UAV. Based on this modification, we can select the tuple $\{k', m', q'\}$ with the greatest potential to increase the cache hit ratio if only m' could serve the users in its transmission range for content k'.

Finally, after selecting the $tuple(i)$, we calculate the optimal $\overline{a}^*(i)$ (where $\mathcal{S}^{i-1} \cup tuple(i)$). This reduces the complexity of the approximation algorithm, since it needs to solve only one convex problem per iteration.

Summarizing, the *fast battery greedy* at each iteration proceeds as follows. It selects a $tuple(i)$ such that

$$tuple(i) = \arg\max_{(k',m',q')} CHR(t, \mathcal{S}^{i-1} \cup \{(k', m', q')\}, \overline{a}(i; k', m', q')) \tag{13}$$

and then sets the optimal value for $\overline{a}(i)$, i.e.,

$$\overline{a}^*(i) = \arg\max_{\overline{a}} CHR(t, \mathcal{S}^{i-1} \cup \{(k', m', q')\}, \overline{a}) \tag{14}$$

Algorithm 2 Greedy Algorithm for Optimization Problem 2.

Input: *association array* $\{z_{kn}\}$, *content demand* $\{p_{n,k}\}$, *combinations pool* $\{c_{m,q,k}\}$, $\forall m \in \mathcal{M},\ q \in \mathcal{Q},\ k \in \mathcal{K}$

1: $S_0 \leftarrow \emptyset;\ i \leftarrow 0$
2: $\overline{a} \leftarrow 0$
3: **while** $i < M \cdot C$ **do**
4: $\quad i \leftarrow i + 1$
5: $\quad (m', q', k') \leftarrow \underset{\ell \in \{c_{m,q,k}\} \setminus S_{i-1}}{argmax} CHR(t, S_{i-1} \cup \{\ell\}, \overline{a}_{i,\ell}),$
6: $\quad S_i \leftarrow S_{i-1} \cup \{(m', q', k')\},$
7: $\quad \overline{a}^*_i \leftarrow \arg\max_{\overline{a}} CHR(t, \mathcal{S}^{i-1} \cup \{(k', m', q')\}, \overline{a}_{i,\ell})$
8: **end while**
9: $S^* \leftarrow S_i$
10: $\overline{a}^* \leftarrow \overline{a}^*_i$
11: **return** $S^*, \overline{a}^*$

5 PERFORMANCE EVALUATION

5.1 Experimental Setup

In this section, we evaluate the performance of the proposed Algorithm 1 and Algorithm 2 through simulations, and compare them against the following baseline or previously proposed algorithms:

- **Random.** The UAV locations and contents to be cached are selected randomly without considering users locations/preferences.
- **First Locate.** An algorithm that first places the UAVs in the locations $q \in \mathcal{Q}$ where they can serve the most users, and then populates the cache of each UAV with the most popular contents among the served users. The main difference with Algorithm 1 is that the UAV placement and caching problems are not treated jointly as in our approaches.
- **K-means.** A modified version (that conforms to the model we consider) of the algorithm proposed in [6]. It groups the users to $|\mathcal{M}|$ clusters using the k-means algorithm (based on their distance), and places each UAV to the centroid of the cluster. Then it populates caches with the most popular contents among the users associated with each UAV. Similarly to *First Locate*, this algorithm considers separately the placement/caching problem.

Area and user locations. We assume a square area $\mathcal{A}$ of dimensions d_1, d_2. N users are located in $\mathcal{A}$. Users are organized in hotspots. Each hotspot is characterized by its center and radius (which expresses the maximum distance between a user-member of the hotspot and its center), and the probability h_i for a user to belong to hotspot i. In the simulation scenarios, we consider three hotspots (two of them are overlapping) of equal radius, with probabilities $h_i = \{\frac{1}{8}, \frac{3}{4}, \frac{1}{8}\}$, respectively.

Content demand. Users request contents from a catalogue $\mathcal{K}$. Content popularity follows a Zipf-law distribution [17] with exponent a. The arrival of content requests, e.g., from a user n for a content k, follows a Poisson distribution with rate $p_{n,k}$.

Energy consumption. We assume contents of equal sizes and equal battery consumption, i.e., $b_k,\ \forall k \in \mathcal{K}$.

Scenario duration. Each scenario has a duration of t time units. Decisions about resource allocation are taken at the beginning of each scenario (content and UAV placement phase), at time 0, and users requests arrive at time $(0, t]$ (content delivery phase).

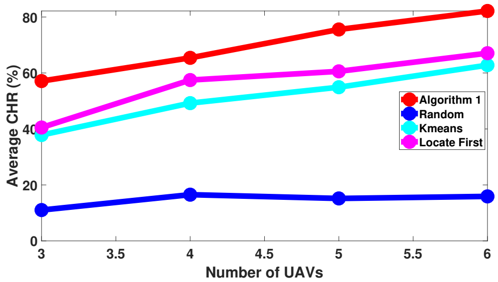

Figure 1: Average CHR for Scenario 1.

The parameters of the simulation scenarios are given in Table 2.

5.2 Results

Gains due to joint optimization of UAV placement and caching. We first simulate scenarios ("Scenario 1") with short duration, in which none of the batteries is depleted (i.e., practically no energy constraints). In Fig. 1 we compare the average *Cache Hit Ratio* (y-axis) when the UAV placement and caching decisions are made by Algorithm 1, Locate First, K-means and Random schemes. We present results for scenarios with different number of UAVs (x-axis).

Figure 1 shows that Algorithm 1 has the highest performance among all schemes, since it examines jointly the UAV placement and cache filling problems. *K-means* and *Locate First* approaches achieve lower performance than Algorithm 1 since they emphasize on finding the popular UAVs locations and, as second step, they decide the cache contents based on the users demand in these locations. For example, we noticed in some scenarios that Algorithm 1 placed more than one UAVs in the same location; this was a location that could serve many users, and in this way Algorithm 1 increased the aggregate number of cached contents in this location. Finally, we can observe that as number of UAVs increases, the CHR increases for each approach, but the difference between them is similar.

Gains due to battery awareness. In Scenario 2, we investigate the effects of the energy constraints. To this end, we increase the experiment duration to $t = 10$ time units in order to increase the probability of a UAV to run out of battery due to user requests. Figure 2 compares the average *CHR* of Algorithm 1 and Algorithm 2 against the baseline approaches.

As shown in the figure, Algorithm 2 outperforms Algorithm 1. This is due to the fact that Algorithm 2 takes into account the UAVs battery levels when deciding UAV placement and caching, as well as selects the user-UAV association. This means that apart from the total content demand and user connectivity criteria, it tries to maximize *CHR* by placing the UAV with the appropriate battery capacity in each selected location.

Table 2: Model parameters used in experiments.

Notation	Value	Notation	Value
$\mathcal{K}$	20	$\mathcal{M}$	{3, 4, 5, 6}
C	3	N	24
d_1,d_2	20	Tr	5
B_m	{21,21,70}	b_c	1
a	0.8	t	Scenario 1:0.8 Scenario 2:10

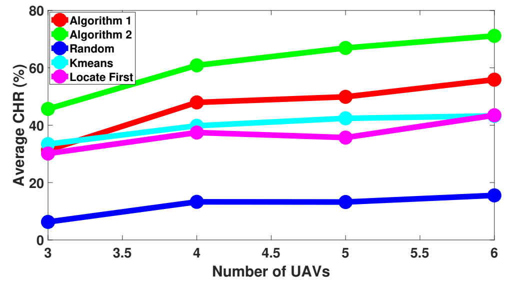

Figure 2: Average CHR for Scenario 2.

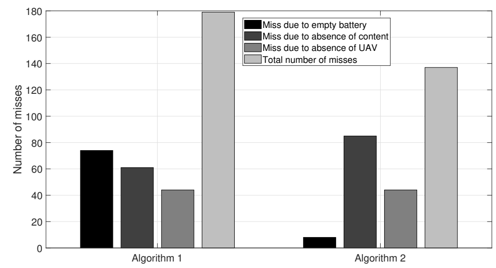

Figure 3: Cache misses analysis.

The effect of the different mechanisms of Algorithm 1 and Algorithm 2 becomes clearer in Fig. 3, which quantifies the effect of different factors in the CHR. Specifically, Fig. 3 depicts the number of cache misses (grouped by the factors that caused them) in the scenarios where the two algorithms were used. While Algorithm 2 has a higher number of cache misses due to cache filling than Algorithm 1, it achieves lower overall number of cache misses. This happens because Algorithm 2 achieves lower number of misses due to empty battery which means that it tries to deal with user demand during UAV placement and cache filling by considering UAVs batteries capacity. In other words, Algorithm 2 satisfies higher number of user requests using optimally the available UAVs batteries. This is the main difference between the two approaches.

6 CONCLUSIONS

In this paper, we studied the problem of joint UAV placement and cache filling, with and without battery constraints, and proposed efficient algorithms for maximizing the *Cache Hit Ratio*. The proposed algorithms differentiate from existing approaches since they solve the two problems jointly and not separately. Simulation results have shown that the proposed approaches increase network performance in terms of *Cache Hit Ratio* compared to existing approaches.

REFERENCES

[1] Mohamed Alzenad, Amr El-Keyi, Faraj Lagum, and Halim Yanikomeroglu. 2017. 3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage. *IEEE Wireless Communications Letters* 6, 4 (Aug. 2017), 434–437. https://doi.org/10.1109/LWC.2017.2700840

[2] Mohamed Alzenad, Amr El-Keyi, and Halim Yanikomeroglu. 2018. 3-D Placement of an Unmanned Aerial Vehicle Base Station for Maximum Coverage of Users With Different QoS Requirements. *IEEE Wireless Communications Letters* 7, 1 (Feb. 2018), 38–41. https://doi.org/10.1109/LWC.2017.2752161

[3] Ramy Amer, Walid Saad, Hesham ElSawy, Majid Butt, and Nicola Marchetti. 2018. Caching to the Sky: Performance Analysis of Cache-Assisted CoMP for Cellular-Connected UAVs. *arXiv preprint arXiv:1811.11098* (2018).

[4] Ejder Bastug, Mehdi Bennis, Engin Zeydan, Manhal Abdel Kader, Ilyas Alper Karatepe, Ahmet Salih Er, and Merouane Debbah. 2015. Big data meets telcos: A

proactive caching perspective. *Journal of Communications and Networks* 17, 6 (Dec. 2015), 549–557. https://doi.org/10.1109/JCN.2015.000102

[5] Ejder Bastug, Jean-Louis Guenego, and Merouane Debbah. 2013. Proactive small cell networks. IEEE, 1–5. https://doi.org/10.1109/ICTEL.2013.6632164

[6] Mingzhe Chen, Mohammad Mozaffari, Walid Saad, Changchuan Yin, Merouane Debbah, and Choong Seon Hong. 2017. Caching in the Sky: Proactive Deployment of Cache-Enabled Unmanned Aerial Vehicles for Optimized Quality-of-Experience. *IEEE Journal on Selected Areas in Communications* 35, 5 (May 2017), 1046–1061. https://doi.org/10.1109/JSAC.2017.2680898

[7] Emily M Craparo, Jonathan P How, and Eytan Modiano. 2011. Throughput optimization in mobile backbone networks. *IEEE Transactions on Mobile Computing* 10, 4 (2011), 560–572.

[8] Jocelyne Elias and Bartlomiej Blaszczyszyn. 2017. Optimal geographic caching in cellular networks with linear content coding. In *2017 15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*. IEEE, Paris, France, 1–6. https://doi.org/10.23919/WIOPT.2017.7959865

[9] Negin Golrezaei, Andreas F. Molisch, Alexandros G. Dimakis, and Giuseppe Caire. 2013. Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution. *IEEE Communications Magazine* 51, 4 (April 2013), 142–149. https://doi.org/10.1109/MCOM.2013.6495773

[10] Negin Golrezaei, Karthikeyan Shanmugam, Alexandros G Dimakis, Andreas F Molisch, and Giuseppe Caire. 2012. Femtocaching: Wireless video content delivery through distributed caching helpers. In *Proc. IEEE INFOCOM*. 1107–1115. https://arxiv.org/pdf/1109.4179.pdf.

[11] Arif Can Gungor and Deniz Gunduz. 2015. Proactive wireless caching at mobile user devices for energy efficiency. In *2015 International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, Brussels, Belgium, 186–190. https://doi.org/10.1109/ISWCS.2015.7454325

[12] Kenza Hamidouche, Walid Saad, Merouane Debbah, Ju Bin Song, and Choong Seon Hong. 2017. The 5G Cellular Backhaul Management Dilemma: To Cache or to Serve. *IEEE Transactions on Wireless Communications* 16, 8 (Aug. 2017), 4866–4879. https://doi.org/10.1109/TWC.2017.2702559

[13] Mingyue Ji, Giuseppe Caire, and Andreas F. Molisch. 2016. Wireless Device-to-Device Caching Networks: Basic Principles and System Performance. *IEEE Journal on Selected Areas in Communications* 34, 1 (Jan. 2016), 176–189. https://doi.org/10.1109/JSAC.2015.2452672

[14] Mohammad G Khoshkholgh, Keivan Navaie, Halim Yanikomeroglu, Kang G Shin, and Victor CM Leung. 2019. Randomized Caching in Cooperative UAV-Enabled Fog-RAN. (2019).

[15] Samir Khuller, Anna Moss, and Joseph Seffi Naor. 1999. The budgeted maximum coverage problem. *Inform. Process. Lett.* 70, 1 (1999), 39–45.

[16] Andreas Krause and Daniel Golovin. 2012. Submodular function maximization. *Tractability: Practical Approaches to Hard Problems* 3, 19 (2012), 8.

[17] Xiaosheng Lin, Junjuan Xia, and Zhi Wang. 2019. Probabilistic caching placement in UAV-assisted heterogeneous wireless networks. *Physical Communication* 33 (2019), 54–61.

[18] Jiangbin Lyu, Yong Zeng, Rui Zhang, and Teng Joon Lim. 2017. Placement Optimization of UAV-Mounted Mobile Base Stations. *IEEE Communications Letters* 21, 3 (March 2017), 604–607. https://doi.org/10.1109/LCOMM.2016.2633248

[19] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, and Mérouane Debbah. 2016. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage. *IEEE Communications Letters* 20, 8 (2016), 1647–1650.

[20] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, and Merouane Debbah. 2016. Unmanned Aerial Vehicle With Underlaid Device-to-Device Communications: Performance and Tradeoffs. *IEEE Transactions on Wireless Communications* 15, 6 (June 2016), 3949–3963. https://doi.org/10.1109/TWC.2016.2531652

[21] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, and Merouane Debbah. 2017. Wireless Communication Using Unmanned Aerial Vehicles (UAVs): Optimal Transport Theory for Hover Time Optimization. *IEEE Transactions on Wireless Communications* 16, 12 (Dec. 2017), 8052–8066. https://doi.org/10.1109/TWC.2017.2756644

[22] Nam Tuan Nguyen, Yichuan Wang, Husheng Li, Xin Liu, and Zhu Han. 2012. Extracting typical users' moving patterns using deep learning. In *2012 IEEE Global Communications Conference (GLOBECOM)*. IEEE, Anaheim, CA, USA, 5410–5414. https://doi.org/10.1109/GLOCOM.2012.6503981

[23] Jun Rao, Hao Feng, Chenchen Yang, Zhiyong Chen, and Bin Xia. 2016. Optimal caching placement for D2D assisted wireless caching networks. In *2016 IEEE International Conference on Communications (ICC)*. IEEE, Kuala Lumpur, Malaysia, 1–6. https://doi.org/10.1109/ICC.2016.7511410

[24] Pavlos Sermpezis, Thrasyvoulos Spyropoulos, Luigi Vigneri, and Theodoros Giannakas. 2017. Femto-caching with soft cache hits: Improving performance with related content recommendation. In *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 1–7.

[25] Hazim Shakhatreh, Abdallah Khreishah, Ayoub Alsarhan, Issa Khalil, Ahmad Sawalmeh, and Noor Shamsiah Othman. 2017. Efficient 3D placement of a UAV using particle swarm optimization. In *2017 8th International Conference on Information and Communication Systems (ICICS)*. IEEE, Irbid, Jordan, 258–263. https://doi.org/10.1109/IACS.2017.7921981

[26] Hazim Shakhatreh, Abdallah Khreishah, and Bo Ji. 2017. Providing wireless coverage to high-rise buildings using uavs. *arXiv preprint arXiv:1705.09770* (2017).

[27] Haichao Wang, Guoru Ding, Feifei Gao, Jin Chen, Jinlong Wang, and Le Wang. 2018. Power Control in UAV-Supported Ultra Dense Networks: Communications, Caching, and Energy Transfer. *IEEE Communications Magazine* 56, 6 (June 2018), 28–34. https://doi.org/10.1109/MCOM.2018.1700431

[28] Xuehe Wang, Lingjie Duan, and Rui Zhang. 2016. User-Initiated Data Plan Trading via a Personal Hotspot Market. *IEEE Transactions on Wireless Communications* 15, 11 (Nov. 2016), 7885–7898. https://doi.org/10.1109/TWC.2016.2608957

[29] Xiaoli Xu, Yong Zeng, Yong Liang Guan, and Rui Zhang. 2018. Overcoming Endurance Issue: UAV-Enabled Communications With Proactive Caching. *IEEE Journal on Selected Areas in Communications* 36, 6 (June 2018), 1231–1244. https://doi.org/10.1109/JSAC.2018.2844979

[30] Pengcheng Zhan, Kai Yu, and A. Lee Swindlehurst. 2006. Wireless Relay Communications using ann Unmanned Aerial Vehicle. In *2006 IEEE 7th Workshop on Signal Processing Advances in Wireless Communications*. IEEE, Cannes, France, 1–5. https://doi.org/10.1109/SPAWC.2006.346492

[31] Nan Zhao, Fen Cheng, F. Richard Yu, Jie Tang, Yunfei Chen, Guan Gui, and Hikmet Sari. 2018. Caching UAV Assisted Secure Transmission in Hyper-Dense Networks Based on Interference Alignment. *IEEE Transactions on Communications* 66, 5 (May 2018), 2281–2294. https://doi.org/10.1109/TCOMM.2018.2792014