Joint Optimization of Caching Placement and Trajectory for UAV-D2D Networks

Tiankui Zhang, *Senior Member, IEEE,* Yi Wang, Wenqiang Yi, *Member, IEEE,* Yuanwei Liu, *Senior Member, IEEE,* and Arumugam Nallanathan, *Fellow, IEEE*

Abstract

With the exponential growth of data traffic in wireless networks, edge caching has been regarded as a promising solution to offload data traffic and alleviate backhaul congestion, where the contents can be cached by unmanned aerial vehicle (UAV) and user terminal (UT) with local data storage. In this article, a cooperative caching architecture of UAV and UTs with scalable video coding (SVC) is proposed, which provides the high transmission rate content delivery and personalized video viewing qualities in hotspot areas. In the proposed cache-enabling UAV-D2D networks, we formulate a joint optimization problem of UT caching placement, UAV trajectory, and UAV caching placement to maximize the cache utility. To solve this challenging mixed integer nonlinear programming problem, the optimization problem is decomposed into three sub-problems. Specifically, we obtain UT caching placement by many-to-many swap matching algorithm, then obtain the UAV trajectory and UAV caching placement by approximate convex optimization and dynamic programming, respectively. Finally, we propose a low complexity iterative algorithm for the formulated optimization problem to improve the system capacity, fully utilize the cache space resource, and provide diverse delivery qualities for video traffic. Simulation results reveal that: i) the proposed cooperative caching architecture of UAV and UTs obtains larger cache utility than the cache-enabling UAV networks with same data storage capacity and radio resource; ii) compared with the benchmark algorithms, the proposed algorithm improves cache utility and reduces backhaul offloading ratio effectively.

Index Terms

Tiankui Zhang and Yi Wang are with School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China (e-mail: {zhangtiankui, wangyi96}@bupt.edu.cn);

Wenqiang Yi, Yuanwei Liu and Arumugam Nallanathan are with School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. (e-mail: w.yi, yuanwei.liu, a.nallanathan@qmul.ac.uk).

Cooperative caching architecture, D2D communications, UAV flight trajectory, SVC caching

I. Introduction

In recent years, unmanned aerial vehicles (UAVs) have been developed rapidly and widely used in many industries due to its small size, low deployment, and high flexibility. The characteristics of UAV make it possible to solve problems effectively in traditional cellular networks, such as high deployment cost and poor adaptability to special deployment scenarios. Therefore, UAV-assisted cellular networks have attracted much attention recently [1, 2]. The main application scenarios of UAV communications include high-speed coverage of hotspots [3], emergency communication [4], relay communication [5], and information dissemination and collection [6]. With the exponential growth of data traffic, it is predicted that by the end of 2021, the monthly traffic in mobile cellular network will reach 60 Exabyte (EB), of which 72% will be used for content delivery [7]. Edge caching has been proposed as a key enabling technique for cellular networks to alleviate traffic load [8]. Popular contents can be placed close to the mobile users, such as at the base stations (BSs) [9], user terminals (UTs) [10], and UAVs [11], so as to reduce the content acquisition delay and alleviate backhaul traffic load. UT caching is to store the contents at UTs and then share the contents among UTs by device-to-device (D2D) communications [12]. The combination of UT caching and D2D communications can effectively reduce the repeated content transmission, improve the quality of experience (QoE) of UTs and reduce BS traffic load in hotspot areas [13]. In cache-enabling UAV-assisted D2D-enabled cellular networks, the collaborative caching architecture of UAV and UTs can not only make full use of limited cache space, but also further improve system capacity performance. The cache and contents management in UAV and UTs can be realized with the aid of mobile edge computing technology [14]. With the cache-enabling UAV-D2D Networks, the massive popular content delivery can be achieved easily for some temporary hotspot coverage, emergency communication and rescue scenarios.

Nowadays, users prefer personalized viewing qualities for different types of video files. Users usually prefer basic perceptual qualities for sports games and news, while high qualities for movies. In order to fully take advantage of diverse quality requirements feature of video traffic, scalable video coding (SVC) is proposed which encodes video in the form of layer files [15]. SVC is able to flexibly remove part of the video bit streams to adjust to diverse user requirements and network states while guaranteeing acceptable video quality. With SVC, the original video file is encoded into L layers including a base layer (BL) file and $(L-1)$ enhancement layer (EL) files. The successful reception of BL file can make the video with basic quality to be retrieved. The EL files contain supplementary information of BL file, which can upgrade the quality of the received video. However, the l-th EL file cannot be decoded without the previous $(l-1)$ layer files. Obviously, the quality of received video depends on received layer files. In cache-aided scalable video delivery, the successful delivering required video file with quality level l should be make sure the successful delivering previous $(l-1)$ layer files. So the caching utility of a video file with SVC not only depends on the cached l-th EL file, but also is decided by the caching placement of previous $(l-1)$ layer files [16]. In cache-enabling D2D networks with SVC, the joint optimization of probabilistic caching placement and power allocation on scalable video transmission was studied [17].

A. Related Works

Many researchers have carried out research in the field of UAV communications. Generally speaking, the current research on UAV-assisted cellular networks is mainly on UAV location deployment, UAV trajectory design, and resource allocation, aiming at enhancement system capacity, providing high-speed transmission, and improvement energy efficiency.

Due to its flexible deployment, UAV can adjust the flight altitude and location to cover the area on the ground and establish a reliable line-of-sight (LoS) link with ground users [18–22]. A novel framework for UAV networks with massive access capability supported by non-orthogonal multiple access (NOMA) was proposed in [18]. The authors in [19] investigated dynamic resource allocation of multi-UAV networks by multi-agent reinforcement learning. Three-dimensional (3D) deployment of UAVs was studied with the goal of maximizing the total amount of data transmitted by UAVs with the tradeoff among flight altitude, energy expense, and flight time in [20]. A distributed algorithm was proposed to allow UAVs to learn their 3D locations and associate with ground users dynamically while maximizing the sum rate in [21]. Aim at maximizing the instantaneous sum rate of all ground pairs, while ensuring the rate requirements of the cellular user, UAV location deployment and transmit power allocation were jointly optimized in [22].

Due to the mobility and energy limitation of UAV, the UAV trajectory optimization plays a very important role in UAV-assisted cellular networks. Reasonable UAV trajectory can help improve system performance, such as energy efficiency [23] and transmission rate [24–26], etc. With the target of maximizing energy efficiency with guaranteed QoE of users, joint optimization of UAV communication scheduling, UAV trajectory, power allocation, and bandwidth allocation were studied in [23]. The authors in [24] proposed a multi-agent Q-learning-based trajectory design and power control algorithm for sum transmission rate maximization. In order to maximize the minimum data rate of the mobile users, joint optimization of power allocation, user association, channel assignment and trajectory optimization were studied in [25]. By jointly optimizing the user association and UAV communication scheduling, transmit power, and UAV trajectory over a finite period, an optimization problem was formulated to maximize the minimum secrecy rate among users in [26]. In the above work, both the static UAV deployment and UAV trajectory were studied jointly with resource allocation, since the deployment and movement of UAV has a great impact on the performance of resource allocation, and vice versa.

A few work has studied the edge caching combining with UAV communications [27–33]. The main purpose of the cache-enabling UAV is to cache popular contents in the UAV BSs related to their associated users so that most frequently requested contents can be served from local caches, instead of forwarding the users' requests over the bandwidth-limited wireless backhaul links. 3D placement of UAV BSs, user-BS association, and bandwidth allocation were jointly studied to minimize the total transmit power while considering the data rate requirements and limited backhaul capacity in [27]. However, the caching placement was not optimized in [27]. In [28], caching placement was predicted based on content request distribution and optimized by cache space allocation and resource allocation, but the UAV deployment was not considering in the optimization. In [29], both static and dynamic UAV deployments were considered. For static UAV deployment, an optimization problem was formulated to design the cache placement at UAV and users in order to maximize the cache hit probability. For dynamic UAV deployments, a spiral algorithm was proposed to minimize the number of UAV path points while covering all users. Although caching placement and UAV deployment were both considered in [29], the advantages of joint optimization of caching placement and UAV deployment/trajectory were ignored. In the afterwards research literature, the caching placement and UAV trajectory were jointly studied in cache-enabling UAV networks [30–33]. Caching UAV assisted secure transmission in hyper-dense networks based on interference alignment was studied in [30]. The authors in [31] proposed a joint caching placement, UAV trajectory design, and resource allocation algorithm in dynamic cache-enabling UAV NOMA networks. The joint optimization of caching placement and static UAV deployment was carried out to maximize QoE in [32]. [33] investigated the cache-enabling UAV NOMA networks for augmented reality (AR) applications. The above works mainly consider the traditional content caching without SVC. We should note that, the caching placement of cache-aided scalable video delivery should consider both the BL file and EL files, since the utility of caching the l-th EL file is highly dependence on the caching placement of previous $(l-1)$ layer files. It means that, the existing solutions of caching placement and UAV trajectory design for cache-enabling UAV networks cannot be used straightforward in cache-enabling UAV networks with SVC. Meanwhile, we study the caching placement in a collaborative caching architecture of UAV and UTs, in which, UT caching placement, UAV trajectory and UAV caching placement should be jointly optimized to improve the system capacity, fully utilize the cache space resource, and provide diverse delivery qualities for video traffic.

B. Motivation and Contribution

As discussed above, reasonable SVC caching placement strategy can help save time for UTs to obtain requested videos and alleviate the backhaul pressure of macro BS. Optimization UAV trajectory can effectively improve the transmission rate between UAV and UTs. However, there are few works considering the joint optimization of SVC caching placement and UAV trajectory. In this paper, we take maximum cache utility as our optimization target in a cooperative SVC video caching architecture of UAV and UTs. If the layer files of requested video file are cached in UTs or UAV, files can be shared by D2D communication or UAV communication link. The more files shared, the higher cache utility will be. Moreover, UAV trajectory optimization will adjust the relative position between UAV and UTs, which can enhance the transmission rate so as to improve the cache utility. In other words, the caching placement and UAV trajectory are closely coupled. Motivated by this, we study the joint optimization of UT caching placement, UAV trajectory and UAV caching placement. The main contributions of this paper are summarized as follows:

- We propose a collaborative SVC video caching architecture in cache-enabling UAV-D2D cellular networks, where the UAV and UTs are both equipped with local storage unit. UTs can share local layered files with nearby UTs by D2D communications. The UAV can also provide layered files to UTs. Then a joint optimization problem of UT caching placement, UAV flight trajectory, and UAV caching placement is formulated to maximize the cache utility of all files in the networks.

- We propose a joint iterative algorithm to solve the formulated optimization problem. Since the original optimization problem is a mixed integer nonlinear programming problem and hard to solve directly, the original optimization problem is decomposed into three sub-problems. Specifically, we model the UT caching placement as a many-to-many swap matching, obtain the UAV flight trajectory by approximate convex optimization method, then propose a dynamic programming based UAV caching placement algorithm. Based on the proposed three algorithms of the sub-problems, we propose a low complexity joint iterative algorithm and analyse the computational complexity of the proposed algorithm.
- We demonstrate the network performance to verify the feasibility and effectiveness of the proposed algorithm. Simulation results show that the proposed algorithm can reach the convergence within 150 iterations and obtain a sub-optimal solution. Then the advantage of cooperative SVC video caching architecture of UAV and UTs is demonstrated by simulation results. Furthermore, we investigate the influence of the number of UTs, UAV cache space, Zipf parameter and UAV height on cache utility and backhaul offloading ratio, respectively. Simulation results verify that the cooperation caching architecture can take full advantage of edge caching and cache utility is effectively improved by subdividing users' requests for different quality and SVC caching.

C. Organization

The rest of the paper is organized as follows. The system model and problem formulation are presented in Section II. Section III is the proposed algorithm to solve the optimization problem. Section IV is the numerical simulation results. Finally, conclusions are drawn in Section V.

II. System Model

In the cache-enabling UAV-assisted D2D-enabled cellular networks, there is a ground macro BS and a cache-enabling UAV BS flying along a predefined trajectory to assist the ground macro BS. Fig. 1 shows an example of the practical scenario. The macro BSs near the stadium are overloaded which cannot fulfill the traffic requirement of users in peak hours, for example,

TABLE I: Main Symbol and Variable List

Parameter	Description
F	Number of video files
N	Number of UTs
$\mathbf{q}_n$	Coordinate of UT n
$\mathbf{w}[t]$	Coordinate of UAV at time slot t
L	Video quality level
S_V, S_U	Cache space of UAV, UT
o	Size of each layer file
γ	Zipf parameter
H	Height of UAV flight trajectory
V_{max}	Maximum flight speed of UAV
τ	Length of time slot
T	Number of time slots
P_V, P_U	Transmit power of UAV, UT
B_V, B_D	UAV communication link bandwidth of UAV, D2D UT
N_0	Power spectral density of additive white Gaussian noise
ψ_U,ψ_V	Unit content caching profit of UT, UAV
ξ_U,ξ_V	Unit content caching cost of UT, UAV
$r_{nn'}$	Transmission rate from UT n to UT n'
$d_{nV}[t]$	Distance between UAV and UT n at time slot t
$r_n[t]$	Transmission rate from UAV to UT n at time slot t
U_{nf}^1	Cache utility of the BL file of video file f cached in UT n
U_{Vf}^l	Cache utility of the l-th layer file of video file f cached in UAV
q_{nf}^1	Indicator of whether UT n requests the video file f with quality level l
C_{nf}^1	Indicator of whether the BL file of video file f is cached in UT n
C_{Vf}^l	Indicator of whether the l-th layer file of video file f is cached in UAV

the time during a football game. In this case, the traffic offloading is assisted by UAV and UTs equipped with cache storages. The main symbols and variables used in this paper are summarized in Table I.

There are N UTs equipped with cache capability, which can communicate directly without the macro BS. The set of UTs is denoted as $\mathcal{N} = \{1, 2, ..., N\}$. The cache space of UAV is S_V bits and the cache space of each UT is S_U bits. The UAV first flies along the pre-designed trajectory and then returns to the starting point at the end of each flight. During the flight, the UAV transmits data to UTs. UTs can also obtain the data from nearby UTs by D2D communications. During the content distribution, each UT can be both sending UT and receiving UT. To encourage the files sharing and D2D communications of UTs, the macro BS rewards each UT that successfully delivers the cached files to its surrounding UTs. Therefore, the UTs have the motivation to participate in the files sharing to increase cache space utilization.

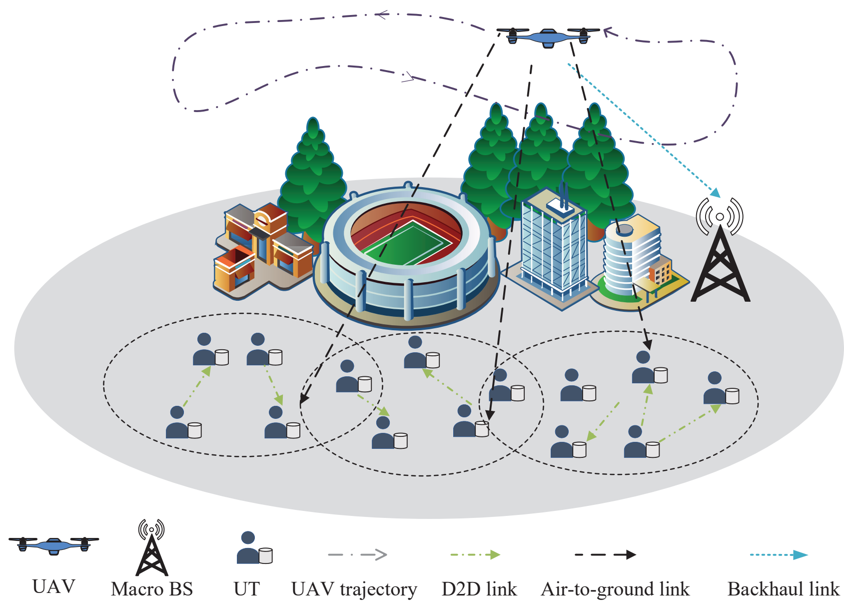

Fig. 1: Cache-enabling UAV-assisted D2D-enabled cellular networks.

There are F contents in video file library. The set of video file is denoted as $\mathcal{F}=\{1,2...,F\}$. Each video file is encoded into L layer files. The size of each layer file is o bits. The encoded layer files are denoted as $\mathcal{L}=\{1,2,...,L\}$, where the first element $\{1\}$ is the BL file and all rest elements $\{2,...,L\}$ are EL files. Since each video file is encoded into L layer files, there are L quality levels of each video file as well. In this framework, UAV and UTs cache replicas of BL and EL files of popular video files. In order to make full use of the limited cache space and the advantages of D2D communications, it is assumed that UTs only cache the BL files, while UAV can cache both the BL files and EL files. Let $q^l_{nf}=1$ indicate that UT n requests video file f with quality level l, which means UT n needs l layer files including $\{1,2,...,l\}$, otherwise $q^l_{nf}=0$. The caching placement indication matrix for UTs is denoted as $C_{\mathcal{N}}$. When the BL file of video f is cached in UT n, it is denoted as $C^1_{nf}=1$, otherwise $C^1_{nf}=0$. The caching placement indication matrix for UAV is denoted as $\mathcal{C}_V$. When the l-th layer file of video f is cached in UAV, it is denoted as $C^l_{Vf}=1$, otherwise $C^l_{Vf}=0$.

A. D2D Communication Model

D2D communications can directly transmit data between UTs without passing through the BS [34]. It can effectively increase network throughput and improve the QoE of UTs. Considering

3D Cartesian coordinate system, we assume that UT's position on the ground is constant for a period of time and the location of UT n is denoted as $\mathbf{q}_n = (x_n, y_n, 0), n \in \mathcal{N}, \mathbf{q}_n \in \mathbf{R}^{3\times 1}$. The spectrum resources of UAV communications are orthogonal to D2D communications.

The channel gain between UT n and UT n' is defined as $g_{nn'}$, including path loss and small-scale fading, which can be known in advance by channel sounding. Only when the channel gain between UT n and n' reaches the predefined D2D communication threshold δ, that is $g_{nn'} > \delta$, the D2D communications link can be established. When UT n and UT n' establish the D2D communication link, it is denoted as $a_{nn'} = 1$, otherwise, $a_{nn'} = 0$. The transmit power of each UT is P_U. The spectrum bandwidth of D2D communications is B_U, which is allocated equally for all the D2D communication pairs in the macro cell. The transmission rate between sending UT n and receiving UT n' is

$$r_{nn'} = B_{nn'} \log_2 \left(1 + \frac{P_U g_{nn'}}{B_{nn'} N_0}\right), \tag{1}$$

where $B_{nn'}$ denotes the bandwidth allocated to UT n and UT n', N_0 is the power spectral density of additive white Gaussian noise. $B_{nn'} = \frac{B_D}{N_{nn'}}$, where $N_{nn'}$ is the number of D2D communication pairs transmitted simultaneously with UEs n and UT n'.

B. *UAV Communication Model*

To simplify the analysis, we use the discrete state space approximation method to discretize the continuous time [35]. A suitable time slot length τ is used to divide the flight time into T time slots. The UAV trajectory is denoted as $\{\mathbf{v}[t] = (x[t], y[t], H), t = 1, 2, ..., T\}$. Correspondingly, the UAV's location projected on the horizontal plane at time slot t can be denoted as $\{\mathbf{w}[t] = (x[t], y[t]), t = 1, 2, ..., T\}$. The maximum flight speed of UAV is $V_{\max}$. At the end of the UAV flight time, the UAV returns to the starting position, that is, $\mathbf{w}[1] = \mathbf{w}[T]$. During the UAV's flight, the UAV provides service for all UTs who request BL or EL files from UAV at the same time. The distance between the UAV and UT n in time slot t is denoted as $d_{nV}[t] = \sqrt{\|\mathbf{w}[t] - \mathbf{q}_n\|^2 + H^2}$.

It is assumed that the air-to-ground communication link is dominated by LoS link which is modeled by the free space path loss model. Thus, the channel gain between the UAV and UT n

in time slot t is denoted as

$$\beta_n[t] = \frac{\beta_0}{\|\mathbf{w}[t] - \mathbf{q}_n\|^2 + H^2}, \tag{2}$$

where β_0 denotes the channel gain at the reference distance $d_0 = 1$ m. The transmission rate between the UAV and UT n in time slot t is

$$r_n[t] = B_n \log_2\left(1 + \frac{\gamma_0 P_0}{\|\mathbf{w}[t] - \mathbf{q}(n)\|^2 + H^2}\right), \tag{3}$$

where $\gamma_0 = \beta_0 / B_n N_0$ denotes the reference signal-to-noise-ratio (SNR) between the UAV and UT n at the reference distance $d_0 = 1$ m. B_n denotes the bandwidth allocated to UT n. In order to make full use of the spectrum resource, the bandwidth resource B_V is allocated equally to all the UTs associated with the UAV. The user association is decided by whether the UTs obtain the files from the UAV, which is provided in the following part.

C. Content Service Model

The popularity of F video files follows a Zipf-like distribution [36]. The popularity distribution of the video files is assumed to remain static over a certain duration. Without loss of generality, The video files are ranked in a descending order according to their popularity. The popularity of the f-th video file is denoted as

$$\rho_f = \frac{1/f^\gamma}{\sum\limits_{f=1}^{F} (1/f^\gamma)}, \tag{4}$$

where the Zipf parameter γ determines the skewness in the UTs' preference. Although different UTs might request the same video file, the video quality requested by different UTs may be different. The quality preference for video files should be considered as well. According to [37], the popularity of the l-th layer file of video file f is denoted as

$$\rho_{fl} = \begin{cases} \rho_f \frac{f-1}{F-1}, & l = 1, \\ \rho_f \frac{F-f}{(F-1)(L-1)}, & 2 \le l \le L. \end{cases} \tag{5}$$

After caching layer files of popular video files, UTs can fulfill the requests themselves and share files to nearby UTs through D2D transmissions as well. The UAV can serve UTs with layer files of video files through UAV communications. Fig. 2 demonstrates the way that UT n

obtains the l layer files of video f when UT n requests video file f with quality level l, i.e., $q_{nf}^{l} = 1$.

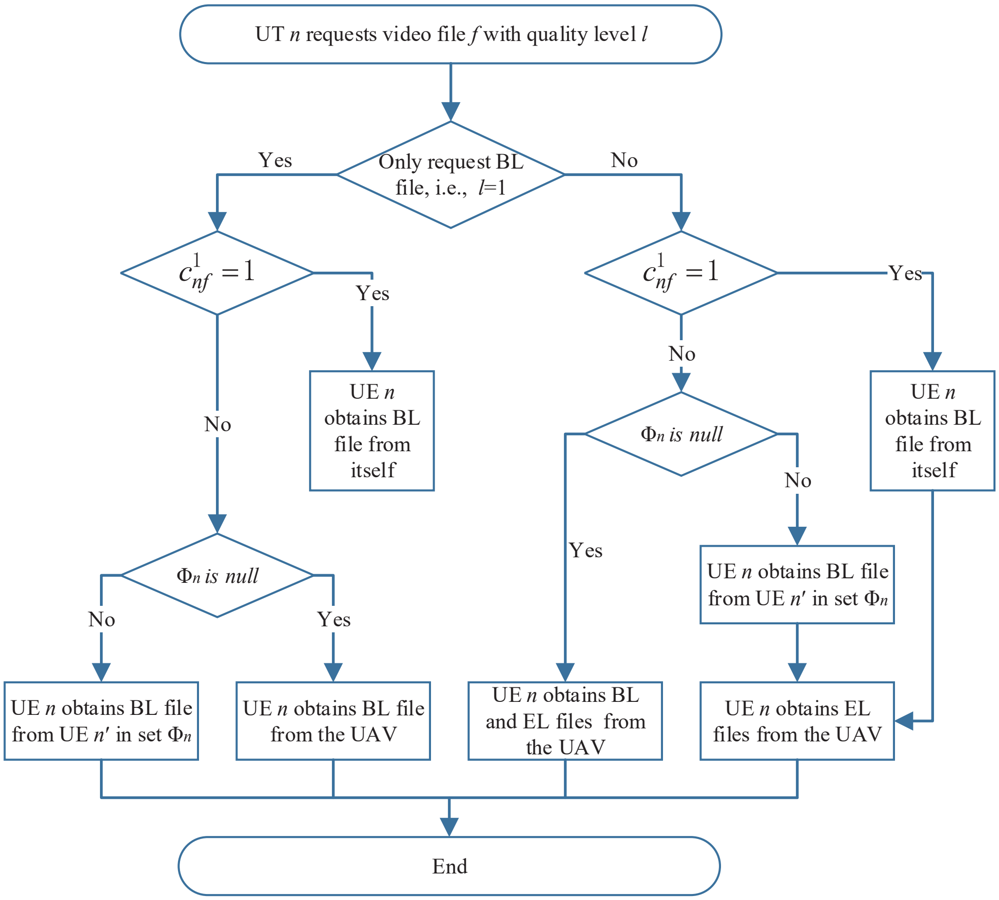

Fig. 2: Process of UT n obtaining files of video f.

When $q_{nf}^{1} = 1$:

- If $C_{nf}^{1} = 1$, UT n obtains the BL file from itself.
- If $C_{nf}^{1} = 0$, the set of UTs who cache the BL file of content f and satisfy the D2D communication condition is denoted as $\Phi_n = \left\{ n' | g_{nn'} > \delta, C_{n'f}^{1} = 1 \right\}, n' \in \mathcal{N}$. UT n receives the BL file from UT n', who is decided by

$$n' = \arg\max_{n' \in \Phi_n} g_{nn'}. \tag{6}$$

- If $C_{nf}^{1} = 0$ and $\Phi_n = \emptyset$, which means UT n cannot receive the BL file of video f from other UTs, UT n requests the BL file of video f from the UAV.

When $q_{nf}^{l}=1, 2\le l\le L$:

- If $C_{nf}^{1}=1$, UT n obtains the BL file from itself and requests $\{2,3,\ldots,l\}$ EL files of video f from the UAV.
- If $C_{nf}^{1}=0$, UT n requests the BL file of video f from UT n' who is decided by (6) and $\{2,3,\ldots,l\}$ EL files of video f from the UAV.
- If $C_{nf}^{1}=0$ and $\Phi_n=\emptyset$, UT n requests the layer files $\{1,2,\ldots,l\}$ of video f from the UAV.

Define g_{nf}^{l} as the indicator of UT n obtaining the l-th layer file of video f from the UAV, which is expressed as

$$g_{nf}^{l}=\begin{cases} q_{nf}^{l}\left(1-C_{nf}^{1}\right)\left(1-\sum\limits_{n'\in\Phi_n}a_{nn'}\right), l=1, \\ q_{nf}^{l}, \qquad\qquad\qquad\qquad\qquad\quad 2\le l\le L. \end{cases} \tag{7}$$

D. Problem Formulation

Firstly, the cache utility is defined. Cache utility defined as the content sharing profit minus caching cost. The cache utility of BL file of video f cached in UT n is defined as

$$U_{nf}^{1}=o\psi_U\sum_{n'\in\Phi_n}\mathrm{q}_{n'f}^{1}a_{nn'}r_{nn'}-o\xi_U, \tag{8}$$

where o is the size of each layer file, ψ_U is unit content sharing profit of D2D UEs, and ξ_U is unit content caching cost of D2D UEs. The content sharing profit is defined based on the product of the number of shared layer files and the transmission rate. When the transmission rate is constant, the more layer files are shared, the greater the cache utility. When the number of shared layer files is fixed, the higher the transmission rate, the greater the cache utility, which means that UTs can receive the files with high data rate and achieve a better QoE.

Similarly, the cache utility of the l-th layer file of video f cached in UAV is defined as

$$U_{Vf}^{l}=o\psi_V\sum_{n=1}^{N}g_{nf}^{l}\frac{1}{T}\sum_{t=1}^{T}r_n\left[t\right]-o\xi_V, \tag{9}$$

where ψ_V is unit content sharing profit of UAV, ξ_V is unit content caching cost of UAV. According to the characteristics of SVC, if the video content is lack of low-level layer file, the requested video file cannot be decoded even if it has a higher EL file. In other words, if the l-th layer file can yield cache utility, UT should have received $\{1, 2, \ldots, l-1\}$ layer files already. Therefore, U^l_{fV} is denoted as follows

$$U^l_{Vf} = \begin{cases} o\psi_V \sum\limits_{n=1}^{N} g^l_{nf} \frac{1}{T} \sum\limits_{t=1}^{T} r_n[t] - o\xi_V, & l = 1, \\ o\psi_V \sum\limits_{n=1}^{N} g^l_{nf} o \frac{1}{T} \sum\limits_{t=1}^{T} r_n[t] - o\xi_V, & \left(C^1_{nf} + C^1_{Vf}\right) C^2_{Vf} \cdots C^{l-1}_{Vf} {=} 1, 2 \le l \le L, \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

According to (7) and (10), the UT caching placement strategy determines whether user would request layer files from the UAV, which will affect the UAV caching placement strategy subsequently.

Then the joint optimization problem is formulated, which considers the UT caching placement, UAV trajectory, and UAV caching placement. The optimization objective is to maximize the cache utility of all video files, which is formulated as

$$\max_{C_{\mathcal{N}}, C_V, \mathbf{w}} \sum_{n=1}^{N} \sum_{\mathrm{f}=1}^{F} C^1_{nf} U^1_{nf} + \sum_{f=1}^{F} \sum_{l=1}^{L} C^l_{Vf} U^l_{Vf} \tag{11}$$

$$\text{s.t.} \quad C^1_{nf} \in \{0, 1\}, \forall f, n, \tag{11a}$$

$$C^l_{Vf} \in \{0, 1\}, \forall f, l, \tag{11b}$$

$$\sum_{f=1}^{F} \sum_{l=1}^{L} C^l_{Vf} \le S_V, \forall f, l, \tag{11c}$$

$$\sum_{f=1}^{F} C^1_{nf} \le S_U, \forall f, n, \tag{11d}$$

$$R_n[t] \ge R_{\min}, n, t, \tag{11e}$$

$$\|\mathbf{w}[t+1] - \mathbf{w}[t]\|^2 \le \tau V_{\max}, \tag{11f}$$

$$\mathbf{w}[1] = \mathbf{w}[T]. \tag{11g}$$

The constraints (11a) and (11b) show two discrete variables representing UT caching placement and UAV caching placement, respectively. The constraints (11c) and (11d) indicate that cache space of UTs and UAV is limited. The constraint (11e) represents that the transmission rate between UAV and UTs should be above the threshold $R_{\min}$ so as to ensure the service quality of UTs. The constraint (11f) indicates that the flight speed of UAV in each time slot is limited.

The constraint (11g) means that the UAV will return to the departure point at the end of the flight time.

III. Proposed Algorithm for Maximum Cache Utility

Based on the formulated optimization problem formulated in Section II, we analyze the problem and propose a low complexity algorithm for maximum cache utility in this section. The formulated optimization problem is the joint optimization of UT caching placement, UAV trajectory and UAV caching placement. UT caching placement and UAV caching placement are both 0-1 discrete variables, while UAV trajectory is continuous variable. The optimization problem is a mixed integer nonlinear programming problem. We decompose the optimization problem into three sub-problems and propose corresponding low complexity algorithms. Although the UT caching placement can be solved by brute-force search, it is not suitable for large scale networks since the algorithm complexity is exponential. To solve the sub-problem of UT caching placement, we introduce the many-to-many swap matching, which can reach a local solution after finite swap operations and converge to a two-sided exchange-stable status. The sub-problem of UAV flight trajectory can be approximated by first-order Taylor expansion, which is a common method for such non-convex problem. We notice that the UAV caching placement sub-problem can be regarded as a grouped knapsack problem by analysis, which will be solved by dynamic programming in this paper. So the three sub-problems are solved by the following algorithms:

(1) many-to-many swap matching based UT caching placement algorithm;

(2) approximate convex optimization based UAV flight trajectory algorithm;

(3) dynamic programming based UAV caching placement algorithm.

The solution of the three sub-problems are denoted as $C_{\mathcal{N}}$, $\mathbf{w}$ and C_V, respectively.

A. Many-to-Many Swap Matching for UT caching placement

With fixed UAV trajectory $\mathbf{w}$ and UAV caching placement strategy C_V, the optimization problem (11) is rewritten as follows

$$\max_{C_{\mathcal{N}}} \sum_{n=1}^{N} \sum_{\mathrm{f}=1}^{F} C_{nf}^{1} U_{nf}^{1} + \sum_{f=1}^{F} \sum_{l=1}^{L} C_{Vf}^{l} U_{Vf}^{l} \tag{12}$$

$$\text{s.t.} \quad (11\text{a}), (11\text{d}).$$

Based on the system model, the copy of a BL file can be cached in several UTs and a UT can cache several copy of BL files with limited cache space. However, the copy of BL file can only be cached once in one UT. To solve this problem, we introduce the many-to-many swap matching to obtain the UT caching placement strategy. The definition of many-to-many matching is defined as follows.

Definition 1. *A many-to-many matching Ψ is a function from the set $\mathcal{N} \cup \mathcal{F}$ into the all subsets of $\mathcal{N} \cup \mathcal{F}$, such that*

- $\Psi(f) \subseteq \mathcal{N}, |\Psi(f)| \leqslant N, \forall f \in \mathcal{F}$
- $\Psi(n) \subseteq \mathcal{F}, |\Psi(n)| \leqslant S_U, \forall n \in \mathcal{N}$
- $\Psi(n) = f \Leftrightarrow \Psi(f) = n, \forall n \in \mathcal{N}, \forall f \in \mathcal{F}$

When the BL file of video f is cached in UT n, the matching pair is denoted as (n, f).

Remark 1. *The proposed many-to-many matching has externalities, where the cache utility from the matching pair (n, f) does not only depend on the BL file of video f cached by UT n, but also on the other UTs caching the same BL file. Therefore, the proposed many-to-many matching is a matching with externality.*

A swap matching is defined as follows

$$\begin{aligned} \Psi_{nf}^{n'f'} = & \{\Psi \backslash \{(n, f), (n', f')\}\} \cup \\ & \{(n, \{\{\Psi(n) \setminus \{f\}\} \cup \{f'\}\}), (n', \{\{\Psi(n') \setminus \{f'\}\} \cup \{f\}\})\}, \end{aligned} \tag{13}$$

where $f \in \Psi(n), f' \in \Psi(n'), f \notin \Psi(n'), f' \notin \Psi(n)$. A swap operation enables UT n and n' to switch one of their matching BL files while keeping other UTs and BL files' matching pairs unchanged. Define $\mathrm{E}_f(\Psi)$ as the utility of agent f under matching Ψ. Based on the definition of swap operation, the definition of a two-sided exchange-stable matching is introduced as follows.

Definition 2. *A matching is two-sided exchange-stable if and only if there does not exist a pair of agents (f, f'), such that:*

- $\mathrm{E}\left(\Psi_{nf}^{n'f'}\right) < \mathrm{E}\left(\Psi_{nf'}^{n'f}\right)$

- $\mathrm{E}_f\left(\Psi_{nf}^{n'f'}\right) \leqslant \mathrm{E}_f\left(\Psi_{nf'}^{n'f}\right)$, $\mathrm{E}_{f'}\left(\Psi_{nf}^{n'f'}\right) \leqslant \mathrm{E}_{f'}\left(\Psi_{nf'}^{n'f}\right)$

(f, f') *is called a blocking pair.*

The characteristics of the blocking pair ensure that if a swap matching is approved, the total achievable cache utility of all agents in the matching will increase and the cache utility of two involved BL files will not decrease. The swap operations are expected to take place between the blocking pair. If two UTs want to swap their BL files, the two BL files involved must approve the cache utility. The definition indicates that a swap matching is two-sided exchange-stable when there does not exist any blocking pair under the matching state. To avoid the meaningless cycle of swap matching, we ensure the number of swap between UTs and BL files is less than 2.

As discussed above, the UT caching placement problem is a many-to-many matching problem with externality. To model the externality, the preference list of UT n is formulated as the cache utility of the BL file of video f cached in UT n, which is denoted as follows

$$\varsigma_n^f = U_{nf}^1 = o\psi_U \sum_{n' \in \Phi_n} q_{n'f}^1 a_{nn'} r_{nn'} - o\xi_U, \Psi(n) = f. \tag{14}$$

According to (10), the cache utility of layer files cached in the UAV will be affected by UT caching placement strategy. Therefore, the preference list of the BL file of video f is formulated as the cache utility of the BL file of video f cached in UT n and the UAV, which is denoted as follows

$$\begin{aligned}\varsigma_f^n &= U_{nf}^1 + \sum_{f=1}^{F}\sum_{l=1}^{L} C_{Vf}^l U_{Vf}^l \\ &= o\psi_U \sum_{n' \in \Phi_n} \mathrm{q}_{n'f}^1 a_{nn'} r_{nn'} - o\xi_U + \sum_{f=1}^{F}\sum_{l=1}^{L} C_{Vf}^l U_{Vf}^l, \Psi(f) = n.\end{aligned} \tag{15}$$

The Gale-Shapley (GS) algorithm proposed in [38] is used to construct the initial matching state between UTs and BL file of video $f \in \mathcal{F}$. In the GS based initialization procedure, it is assumed that $C_{nf}^1 = 1, C_{n'f}^1 = 0, n, n' \in \mathcal{N}, n \neq n'$ when we calculate the preference list ς_n^f and ς_f^n according to (14) and (15). Based on the established preference list, each BL file of video file proposes to the favorite UT. At the UT acceptance phase, each UT accepts the BL file with prior preference and rejects others. The initial matching terminates when the cache space of all UTs have been filled with BL files or the UT has rejected left BL files whose cache space is not filled. Based on the initial matching state, the swap operation procedure is employed to further enhance the cache utility. The process of many-to-many swap matching based UT caching placement algorithm is summarized in **Algorithm 1**.

Algorithm 1 Many-to-many swap matching based UT caching placement algorithm

1: Construct the initial matching state $\mathcal{S}_I$ based on the GS algorithm. The matching state is denoted as $\mathcal{S}$. Let $\mathcal{S} = \mathcal{S}_I$
2: **repeat**
3: For any BL file of video $f \in \mathcal{F}$, it searches for another BL file of video $f' \in \mathcal{S} \backslash \mathcal{S}(\Psi(f))$
4: **if** (f, f') is a blocking pair **then**
5: Swap (f, f'), $\Psi = \Psi_{nf'}^{n'f}$
6: Update UT caching placement strategy $C_{\mathcal{N}}$
7: **else**
8: Keep the current matching state
9: **end if**
10: **until** No blocking pair in the matching state
11: **Output:** matching state Ψ and the UT caching placement strategy $C_{\mathcal{N}}$

After performing **Algorithm 1**, any BL file cannot find another BL file to form a swap-blocking pair under the current matching Ψ. Hence, a two-sided exchange-stable matching is formed between UTs and BL file of video files. The stability and convergence of **Algorithm 1** are proved in **Remark 2**.

Remark 2. *The caching placement matching problem considered in this paper is consistent with many to many matching with externalities in [39], which has been proved for stability in [39]. Therefore, there is a stable match between UTs and BL files by **Algorithm 1**. Since the numbers of UTs and BL files are finite and the preference list is strict, the number of proposals from BL files is finite. Meanwhile, the cache utility function increases monotonically by the swap operation in **Algorithm 1** and the cache utility is bounded due to the limited spectrum resources and cache space. Therefore, **Algorithm 1** obtains a local solution after finite swap operations and converges to a two-sided exchange-stable status.*

The complexity of **Algorithm 1** is analyzed as following. The GS algorithm requires each BL file proposes to one UT based on its preference list, and each UT accepts its favorite BL file according to its preference list. The number of cached files in UT n is $I_U = S_U/o$. The computational complexity of the initialization GS algorithm is $\mathcal{O}(I_U N F)$. In the swap matching process, the number of potential swap process is upper bounded by $\mathrm{C}_N^2 F$. For a given number of total iteration I_L, the complexity is $\mathcal{O}\left(L\mathrm{C}_N^2F\right)$. Hence, the complexity of **Algorithm 1** is $\mathcal{O}\left(I_UNF + I_L\mathrm{C}_N^2F\right)$.

B. Approximate Convex Optimization for UAV Trajectory

With fixed UT caching placement and the UAV caching placement strategy, we confirm that some UTs obtain the requested BL file from nearby UTs with good communication conditions. However, due to the limited cache space of UTs and only the BL files cached in UTs, some UTs need to obtain the BL and EL files of the requested video files from the UAV and these UTs are denoted as $\mathcal{K} = \{1, 2, ..., K\}$. According to (7), g_{nf}^l is a constant. At this time, the cache utility of files cached in the UAV increases linearly with the transmission rate between the UAV and UTs, in which case, the optimization problem (11) is equivalent to maximize the sum of throughput of UTs who communicate with the UAV. The equivalent problem is

$$\max_{\mathbf{w}} \sum_{\mathrm{k}=1}^{K} Q_k \sum_{t=1}^{T} r_k\left[t\right] \tag{16}$$

$$\text{s.t.} \quad r_k\left[t\right] \geqslant R_{\min}, \forall t, k \in \mathcal{K} \tag{16a}$$

$$(11\text{f}), (11\text{g}),$$

where Q_k denotes the size of the requested files of UT k from the UAV. Q_k is calculated as

$$Q_k = o\sum_{f=1}^{F}\sum_{l=1}^{L} g_{kf}^l C_{Vf}^l. \tag{17}$$

Q_k is not affected by UAV trajectory, and would be a constant when UT caching placement and UAV caching placement are both fixed.

Note that problem (16) is not a convex optimization problem since the left-hand-sides (LHSs) of constraints (11f) and (16a) are not concave with respect to $\mathbf{w}\left[t\right]$. However, we notice that the LHSs of constraints (11f) and (16a) are convex with respect to $\left\|\mathbf{w}\left[t\right] - \mathbf{q}\left(n\right)\right\|^2$. For a convex function, its first-order Taylor expansion is the global under-estimator at any point [40]. This motivates us to use the convex optimization to solve the non-convex optimization problem. The LHSs of constraints (11f) and (16a) will be replaced by more tractable functions derived from the first-order Taylor expansion at a given local point. Specifically, with the given local point

$\mathbf{w}_j\left[t\right]$, the following inequality is obtained,

$$\begin{aligned} r_k\left[t\right] &= B_k \log_2\left(1+\frac{\gamma_0}{H^2+\left\|\mathbf{w}\left[t\right]-\mathbf{q}\left(n\right)\right\|^2}\right) \\ &\geqslant B_k \alpha_{kj}^{lb}\left[t\right] = r_{kj}^{lb}\left[t\right], \end{aligned} \tag{18}$$

$$\begin{aligned} \alpha_{kj}^{lb}\left[t\right] &= \log_2\left(1+\frac{\gamma_0}{H^2+\left\|\mathbf{w}\left[t\right]-\mathbf{q}\left(n\right)\right\|^2}\right) \\ &-\frac{\log_2\left(e\right)\gamma_0\left(\left\|\mathbf{w}\left[t\right]-\mathbf{q}\left(n\right)\right\|^2-\left\|\mathbf{w}_j\left[t\right]-\mathbf{q}\left(n\right)\right\|^2\right)}{\left(H^2+\left\|\mathbf{w}\left[t\right]-\mathbf{q}\left(n\right)\right\|^2\right)\left(H^2+\left\|\mathbf{w}\left[t\right]-\mathbf{q}\left(n\right)\right\|^2+\gamma_0\right)}. \end{aligned} \tag{19}$$

With any given local point $\mathbf{w}_j\left[t\right]$ and lower bounds $r_{kj}^{lb}\left[t\right]$ in (18), problem (16) is approximated as the following problem

$$\max_{\mathbf{w}} \sum_{\mathrm{k}=1}^{K} Q_k \sum_{t=1}^{T} r_{kj}^{lb}\left[t\right] \tag{20}$$

$$\text{s.t.} \quad r_{kj}^{lb}\left[t\right] \geqslant R_{\min}, \forall t, k \in \mathcal{K} \tag{21}$$

$$\text{(11f)}, \text{(11g)}.$$

Now the constraints (11f) and (21) are both convex quadratic constrains, and (11g) is a linear constraint. Therefore, problem (20) is a convex quadratically constrained quadratic program problem and can be solves by standard convex optimization solvers such as CVX [41]. Since the lower bounds adopted from (16), the optimal objective value obtained by solving (20) serves a lower bound of problem (16). The computational complexity of solving problem (16) is based on the complexity analysis of the interior point method for solving convex conic optimization problem. Specifically, problem (16) involves KT second-order cone constrains of size 2, $(T-1)$ second-order cone constrains of size 4, and one linear equality constrain of size 4. The total number of the optimization variables is $2T$. According to [42], the complexity of solving problem (16) is $\sqrt{2\left(KT+T+3\right)}\left(2T\right)\left(2\times 4^3+\left(T-1\right)\times 4^2+KT\times 2^2\right)$, i.e., $\mathcal{O}\left(K^{\frac{3}{2}}T^{\frac{5}{2}}\right)$.

C. Dynamic Programming for UAV Caching Placement

In the case of fixed UT caching placement strategy $C_{\mathcal{N}}$ and UAV trajectory $\mathbf{w}$, the optimization problem (11) is denoted as

$$\max_{C_V} \sum_{f=1}^{F} \sum_{l=1}^{L} C_{Vf}^{l} U_{Vf}^{l} \quad (22)$$
$$\text{s.t.} \ \ (11\text{b}), (11\text{c}).$$

We optimize the UAV caching placement strategy to transfer video layer files for UTs who request layer files from the UAV, so as to maximize cache utility with limited UAV cache space. If an UT requests the l-th quality level of the video, only when the UT obtains $\{1, 2, ..., l-1\}$ layer files successfully, the l-th layer file cached in UAV can yield cache utility. There are F video files in the file library, in which each video file has one BL file and $(L-1)$ EL files. Number of non empty subsets of $\{1, 2, ..., L\}$ is $\left(2^L - 1\right)$. However, some layer file combinations are impossible to generate complete cache utility because of lack of lower level layer files, such as $\{1, 2, 4, ..., L-1\}$. Thus, we only take some layer file combinations into consideration which can yield complete cache utility, such as $\{1\}$, $\{1, 2\}$, $\{2\}$, $\{1, 2, 3\}$, $\{2, 3\}$,...,$\{1, 2, 3, ..., L-1\}$ and so on. The number of total combinations of each video file is $(2L-1)$. The number of layer files in each combination is different, resulting in different cache utility. And at most one combination of video file f will be cached. This sub-problem can be regarded as a grouped knapsack problem. In order to solve this sub-problem more conveniently, the layer file combination is generalized to the concept of "object". There are F groups layer files in the sub-problem, and each group has $(2L-1)$ objects. Corresponding relationship between layer file combination and object index is shown in TABLE II.

TABLE II: Corresponding relationship between layer file combination and object index

$\{1\}$	$\{1, 2\}$	$\{1, 2, 3\}$	...	$\{1, 2, 3, ..., L\}$	$\{2\}$	$\{2, 3\}$	...	$\{2, 3, ..., L\}$
1	2	3	...	L	$L+1$	$L+2$	...	$2L-1$

The weight of object i is denoted as s_{fi}, which is the sum of the size of the layer files contained in the layer file combination i. The value of object i is denoted as v_{fi}, which is the cache utility that is obtained by caching the layer file combination i. v_{fi} can be calculated

Algorithm 2 Dynamic programming based UAV caching placement algorithm

Initialize: The size s_{fi} and the cache utility v_{fi} of all $F\left(2L-1\right)$ layer file combinations. The initial value $V\left(0,0\right)$=0. The best choice $num\left(f,h\right)=0$. The left backpack capacity is $h=S_V$.

STEP I: Obtain the optimal value $V\left(F,S_V\right)$

for $f=1$ to F **do**
    for $h=S_V$ to 0 **do**
        for $i=1$ to $2L-1$ **do**
            Obtain $V\left(f,h\right)$ by (23)
            if $V\left(f,h\right)$ changes **then**
                $num\left(f,h\right)=i$
            end if
        end for
    end for
end for

STEP II: Obtain the UAV caching placement

Initialize: The left backpack capacity is $h=S_V$ and $f=F$
while $f>0,h>0$ **do**
    if $num\left(f,h\right)>0$ **then**
        C=1
        $h\leftarrow h-s_{f,i}$
    end if
    $f\leftarrow f-1$
end while

by (10) and (7). The optimal UAV caching placement is obtained by dynamic programming. Firstly, whether the group f can be cached in the UAV is decided. If the group f is cached in the UAV, whether the i-th layer file combination is cached in the UAV is decided. Then all the F groups and $F\left(2L-1\right)$ objects will be traversed. The maximum cache utility obtained by caching some groups of $\left\{1,2,...,f\right\}$ with cache space h is denoted as $V\left(f,h\right)$. $V\left(f,h\right)$ is updated as follows

$$V\left(f,h\right)=\max\left\{V\left(f,h\right),V\left(f,h-s_{fi}\right)+v_{fi}\right\}. \tag{23}$$

$V\left(F,S_V\right)$ is the optimal value of the grouped knapsack problem. The dynamic programming for UAV caching placement is summarized in **Algorithm 2**. The complexity of **Algorithm 2** is $\mathcal{O}\left(F\left(I_V\left(2L-1\right)+1\right)\right)$, where $I_V=S_V/o$.

D. Suboptimal Solution for Optimization Problem

Since we solve the optimization problem by decomposing the problem into three sub-problems, we can obtain the sub-optimal solution by alternate iteration based on the algorithms proposed above. The proposed joint UT caching placement, UAV trajectory and UAV caching placement algorithm is described in **Algorithm 3**.

Algorithm 3 Joint UT caching placement, UAV trajectory, and UAV caching placement algorithm

1: **Initialize:** UT caching placement $C_{\mathcal{N}}$, UAV flight trajectory $\mathbf{w}$, and UAV caching placement C_V, iteration index $\omega = 1$
2: **repeat**
3: update $C_{\mathcal{N}}$ by Algorithm1
4: update $\mathbf{w}$ by approximate convex
5: update C_V by Algorithm 2
6: $\omega \leftarrow \omega + 1$
7: **until** Cache utility no longer grows
8: **End**

In practical scenarios, **Algorithm 3** is deployed in the macro BS, which can be seen as a centralized control unit of the networks. The control overhead of **Algorithm 3** includes information collection and information distribution. The collected information contains the video files requesting of UTs, and the cache space of UTs and UAV. The distributed information includes the UT caching placement decision, UAV trajectory design, and UAV caching placement decision. The macro BS can obtain the information of video files requesting of UTs by two ways considering the practical application in the future networks. Such information can be obtained from the application layers if the macro BS is equipped mobile edge computing server. An alternative method is, the macro BS can know such information if the network is driven by the content-centric protocol, which is designed for content delivery for cache enable networks [43].

Assuming that **Algorithm 3** will reach convergence within I'_L iterations. Based on the analysis on the proposed algorithms of three sub-problems, the complexity of **Algorithm 3** is denoted as

$$\mathcal{O}\left(I'_L\left(I_U NF + I_L \mathrm{C}_N^2 NF + \sqrt{2\left(NT+T+3\right)}\left(2T\right)\left(4NT + 16T + 112\right) + I_V\left(2L-1\right)+1\right)\right).$$

IV. Numerical Results

In this section, the system performance of the proposed algorithm is verified by compared with benchmark algorithms. A 200m $\times$ 200m area is considered. The starting position of UAV trajectory is $\mathbf{w}[1] = (0, 0)$. The parameter setting of D2D communication channel is based on 3GPP [44]. The detailed simulation parameters are given in Table III.

TABLE III: Simulation Parameters

Parameter	**Value**
Quality level L	5
Size of each layer file	2 Mbit
Cache space of each UT	20 Mbit
UAV flight height	120 m
Bandwidth of UAV communication B_V	10 MHz
Bandwidth of D2D communication B_D	10 MHz
Maximum flight speed V_{max}	30 m/s
Length of time slot τ	0.2 s
Transmit power of UAV	23 dBm
Transmit power of UT	23 dBm
Reference channel power β_0	-60 dB
Unit content sharing profit of UT, UAV	10^{-6}, 10^{-6}
Unit content caching cost of UT, UAV	2,1
Variance of the Gaussian noise	-174 dBm/Hz

he proposed algorithm is compared with two benchmark algorithms,

- Classic algorithm: the UT caching placement and UAV caching placement are both decided by max-popular caching placement, the UAV trajectory is a fixed with cycle radius 100 m and flight speed 25 m/s.
- Random algorithm: the UT caching placement and UAV caching placement are both subject to the random distribution, and the flight direction and distance of UAV in each time slot are random.

The effectiveness of the proposed algorithm is demonstrated on cache utility and backhaul traffic offloading ratio. The backhaul traffic offloading ratio is defined as the proportion of the number of layer files obtained by UTs in all requested layer files, which is denoted as

$$\Delta = \frac{\sum_{n=1}^{N}\sum_{f=1}^{F}\left(\sum_{l=1}^{L} C_{Vf}^{l} g_{n,f}^{l} + E_{nf}^{1}\right)}{\sum_{n=1}^{N}\sum_{f=1}^{F}\sum_{l=1}^{L} q_{nf}^{l}}, \tag{24}$$

where E^1_{nf} represents whether UT can obtain the BL file,

$$E^1_{nf} = \begin{cases} 0, & C^1_{nf} = 1, C^1_{n'f} = 1, n' \in \Phi_n \\ 1, & else. \end{cases} \tag{25}$$

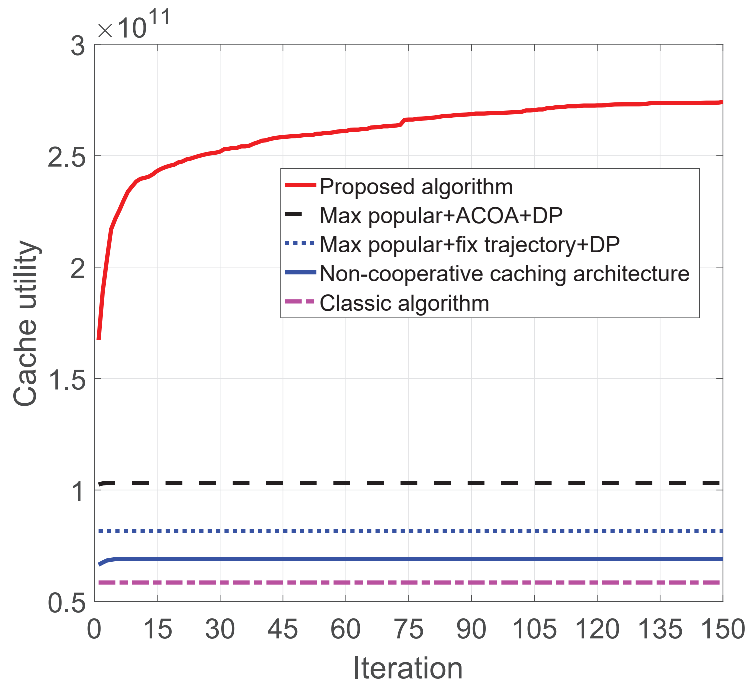

Fig. 3: Effectiveness of the proposed algorithm.

First, the convergence of the proposed algorithm is shown in Fig. 3. We set UT number $N = 20$, UAV cache space $S_V = 100$ Mbit and Zipf parameter $\gamma = 1$. The proposed algorithm can reach the convergence within 150 iterations as shown in Fig. 3. We also compare the proposed algorithm with the classic algorithm and show the improvement of algorithms of three sub-problems, namely many-to-many swap matching algorithm, approximate convex optimization algorithm (ACOA) and dynamic programming (DP). From Fig. 3, all the three sub-problems can improve the cache utility of the system. The improvement brought by many-to-many swap matching algorithm is larger than ACOA and DP, which reveals the importance of UT caching and D2D communication. Meanwhile, the convergence of the proposed algorithm demonstrated in Fig. 3 is consistent with **Remark 2**.

Furthermore, Fig. 3 also shows the advantage of the cooperative caching architecture of UTs and UAV compared with non-cooperative caching architecture, that is, cache-enabling UAV-assisted cellular networks. In non-cooperative caching architecture, it is assumed that UTs do not carry any storage units and cannot share contents by D2D communications. The UAV caches files and provide services for UTs. The UAV flight trajectory and UAV caching placement are obtained by approximate convex optimization and dynamic programming proposed by this article, respectively. The simulation provides equivalent cache storage capacity and transmission bandwidth for non-cooperative caching architecture to ensure fair comparison. It is set UAV cache space $S_V = 500$ Mbit and downlink bandwidth between the UAV and UTs $B_V = 20$ MHz. However, Fig. 3 shows that cooperative caching framework obtain higher cache utility, since cooperative caching framework of UTs and the UAV can make full use of the advantages of UT caching.

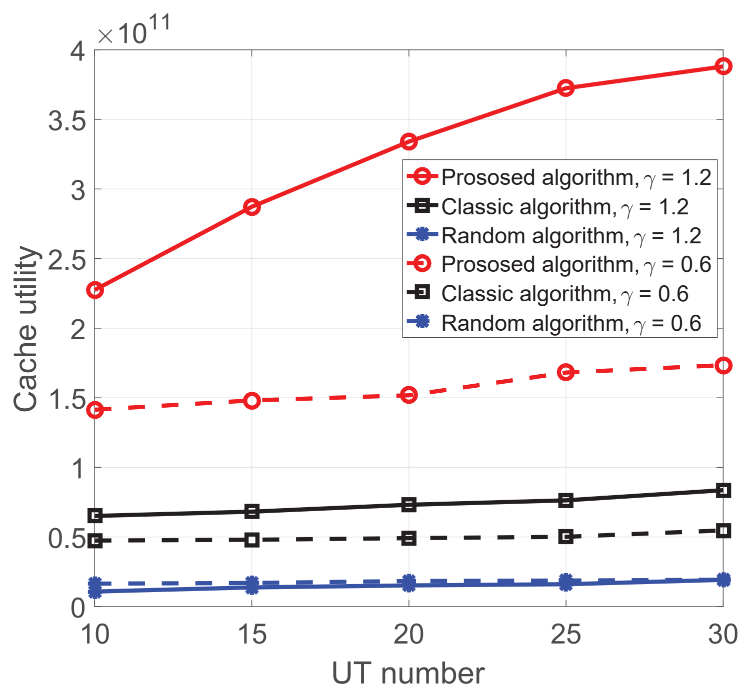

Fig. 4: Cache utility with varying UT number.

Then the effectiveness of the proposed algorithm is demonstrated with varying UT number. It is set that the UT number N is ranging from 10 to 30, $S_V = 100$ Mbit and zipf parameter $\gamma = \{0.6, 1.2\}$. Fig. 4 and Fig. 5 compare the cache utility and backhaul offloading ratio, respectively. With the proposed algorithm, the cache utility increases while the growth rate

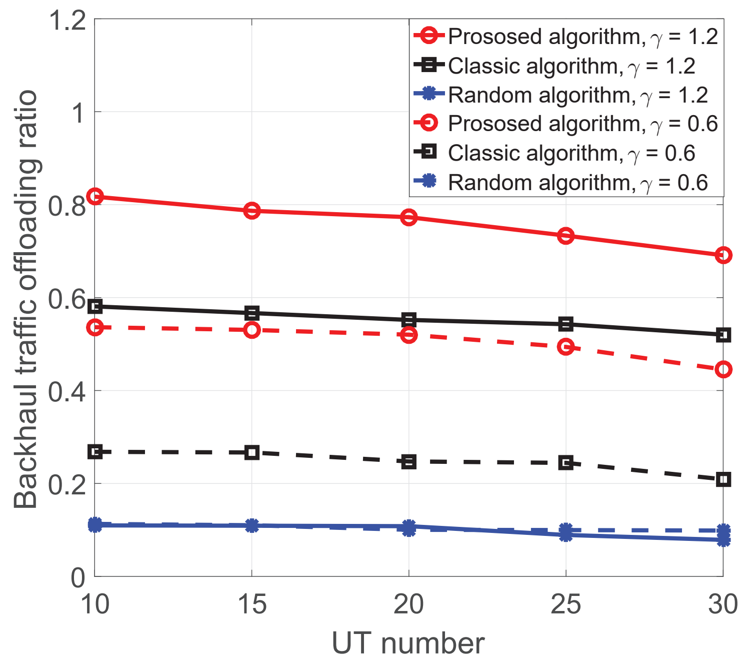

Fig. 5: Backhaul traffic offloading ratio with varying UT number.

gradually decreases with the increasing of the UT numbers. However, the cache utility of the classic algorithm decreases with the increasing of the UT numbers. This shows that UTs cache the same popular files with limited cache space and spectrum resources, which will lead to the gain of sharing content cannot catch up with the growth rate of caching cost. This simulation results also reflects the effectiveness and necessity of the importance of caching placement strategy. With the random algorithm, the UT number has little effect on the system performance. As can be seen from Fig. 5, with the increasing of the UT number, the backhaul traffic offloading ratio gradually decreases in our proposed algorithm and the classic algorithm. The backhaul traffic offloading ratio of the proposed algorithm decreases more slowly than that of the classic algorithm, which also reflects the effectiveness of the caching placement strategy in the proposed algorithm. Besides, the simulation results show that, compared with the cases of $\gamma = 0.6$, all the three algorithms with $\gamma = 1.2$ achieve better system performance. This is because UTs have more requests concentrating on the most popular contents with $\gamma = 1.2$ than $\gamma = 0.6$, since the popularity distribution is nore skewed with larger γ. It means that,in all the three algorithms with $\gamma = 1.2$, the cached files are required by more UTs.

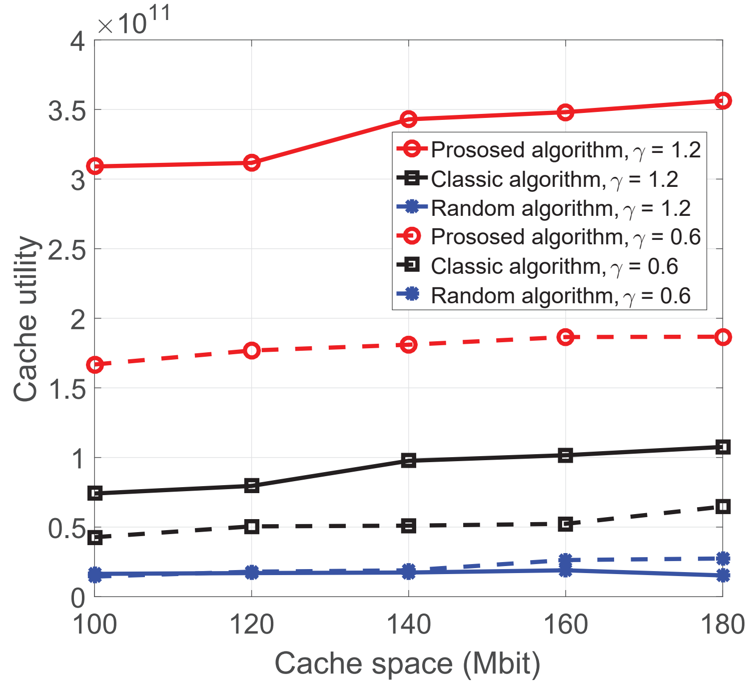

Fig. 6: Cache utility with varying UAV cache space.

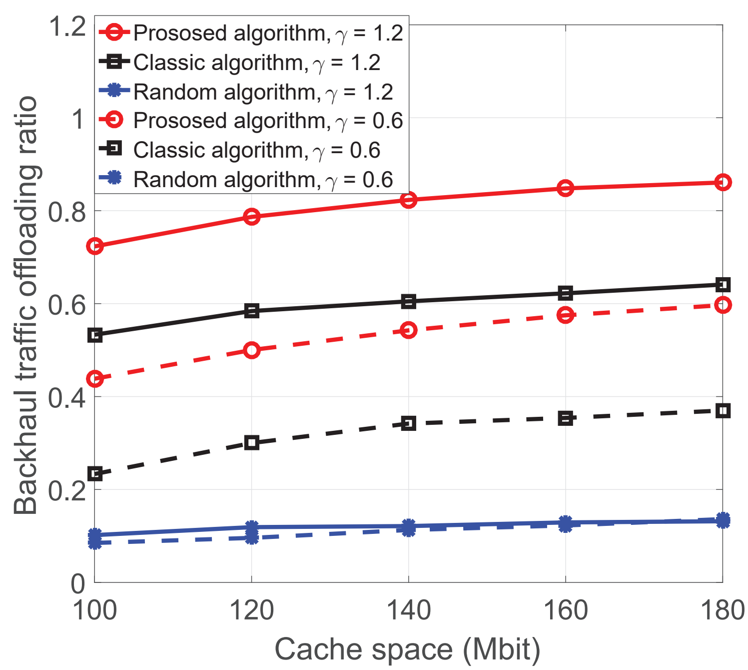

Fig. 7: Backhaul traffic offloading ratio with varying UAV cache space.

Next, the effectiveness of the proposed algorithm with varying UAV cache space is shown. In the simulation, the UAV cache space S_V is ranging from 100 to 180 Mbit, $N = 20$ and $\gamma = \{0.6, 1.2\}$. Fig. 6 and Fig. 7 show that the cache utility and backhaul traffic offloading ratio of the proposed algorithm are improved compared with two benchmark algorithms. The cache utility and backhaul traffic offloading ratio increase as S_V increases. However, the the growth speed of the system performance tends slow with S_V increasing. The reason is that the UAV can cache more layered files but the cache utility of new cached layered files decreases with S_V increasing. In the random algorithm, varying UAV cache space has few impact on system performance. The system performance in the cases of $\gamma = 1.2$ is much better than that of $\gamma = 0.6$. Meanwhile, the growth speed of the performance gain of the proposed algorithm with increasing S_V when $\gamma = 1.2$ is larger compared with $\gamma = 0.6$. The system performance of the proposed algorithm with $\gamma = 1.2$ and $\gamma = 0.6$ is larger than that of classic algorithm, which shows the effectiveness of caching placement optimization.

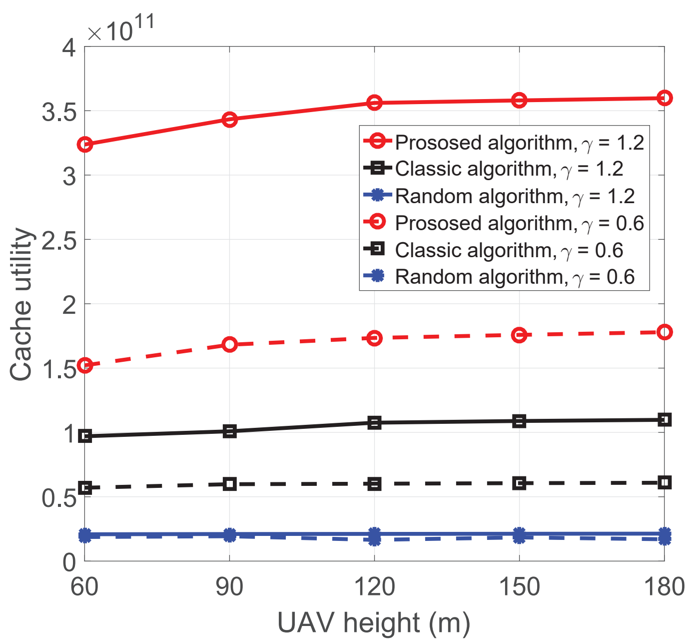

Fig. 8: Cache utility with varying UAV height.

Finally, the effectiveness of the proposed algorithm with varying UAV height is given. In the simulation, the UAV height is ranging from 60 m to 120 m, $N = 20$ and $S_V = 140$ Mbit. Fig. 8 shows that the cache utility grows slowly with the altitude of UAV increasing. Furthermore, the

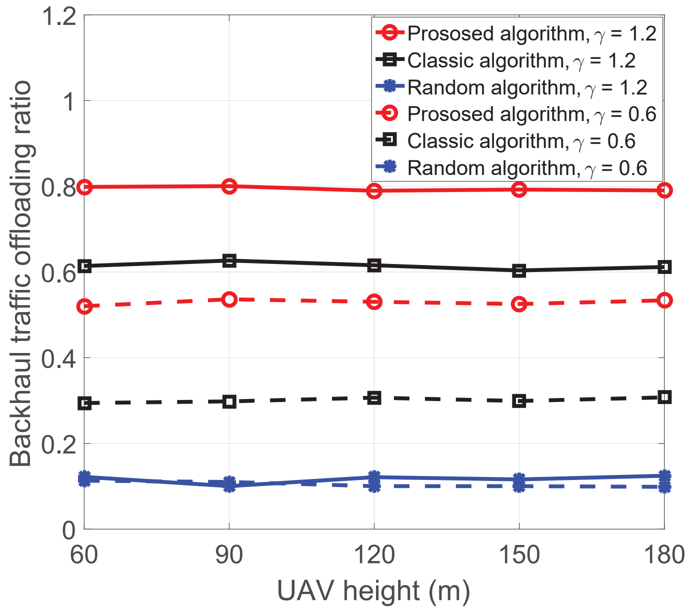

Fig. 9: Backhaul traffic offloading ratio with varying UAV height.

UAV height has little effect on backhaul traffic offloading ratio as shown in Fig. 9. When the UAV flight altitude is high, the air-to-ground communication channel is mainly controlled by the LoS link [45]. For example, when the flight altitude is 120m, the probability that the transmission link is the LoS link in the urban environment is more than 95% [45]. So when the altitude of the UAV is increasing, the LoS link makes the transmission rate between the UAV and UTs increase as well, but the logarithmic function of the transmission rate with the distance between the UAV and UTs makes the UAV flight altitude have little impact on system performance. We should note that, the system performance cannot be improved continuously with the increasing of UAV flight altitude. We obtain the conclusion from Fig. 8 and Fig. 9 since the UAV height is ranging from 60 m to 120 m in the simulation, which has more LoS link probability but smaller path loss fading.

V. Conclusion

In this paper, we have proposed a collaborative SVC video caching architecture in cache-enabling UAV-D2D cellular networks and investigated the joint optimization of UT caching placement, UAV trajectory, and UAV caching placement. We have formulated an optimization problem and proposed a low complexity suboptimal algorithm. The simulation results have verified that the proposed cooperative caching architecture of UAV and UTs can greatly improve the system performance, which also confirm the advantage of UT caching and D2D communications. The effectiveness of the proposed algorithm in terms of cache utility and backhaul offloading ratio has been demonstrated. This paper provides a reference architecture of edge caching and D2D communications with UAV-assisted cellular networks, which would be deployed for massive popular content delivery in the temporary hotspot coverage, emergency communication and rescue scenario. When designing UAV-assisted cellular networks, the joint UAV trajectory and content caching design are vital for the system performance, which has been studied and verified in this paper. In the future work, we would pay more attention to multiple UAVs networking, 3D trajectory optimization of UAV, and caching and computing combination, which are promising research direction to improve the system performance of UAV-assisted cellular networks.

References

[1] L. Gupta, R. Jain, and G. Vaszkun, “Survey of important issues in UAV communication networks,” *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, May. 2016.

[2] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, “A tutorial on UAVs for wireless networks: Applications, challenges, and open problems,” *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2334–2360, thirdquarter 2019.

[3] X. Li, H. Yao, J. Wang, X. Xu, C. Jiang, and L. Hanzo, “A near-optimal UAV-aided radio coverage strategy for dense urban areas,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 9098–9109, Sep. 2019.

[4] T. Zhang, J. Lei, Y. Liu, C. Feng, and A. Nallanathan, “Trajectory optimization for UAV emergency communication with limited user equipment energy: A safe-DQN approach,” *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 3, pp. 1236–1247, Sep. 2021.

[5] Y. Liang, L. Xiao, D. Yang, Y. Liu, and T. Zhang, “Joint trajectory and resource optimization for uav-aided two-way relay networks,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 639–652, Jan. 2022.

[6] M. Hua, L. Yang, Q. Wu, and A. L. Swindlehurst, “3D UAV trajectory and communication design for simultaneous uplink and downlink transmission,” *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5908–5923, Jun. 2020.

[7] “Cisco visual networking index: Global mobile data traffic forecast update, 2016 to 2021 white paper,” https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html, Mar. 2017.

[8] N. Zhao, X. Liu, F. R. Yu, M. Li, and V. C. M. Leung, “Communications, caching, and computing oriented small cell networks with interference alignment,” *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 29–35, Sep. 2016.

[9] M. A. Maddah-Ali and U. Niesen, “Fundamental limits of caching,” *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, Mar. 2014.

[10] T. Zhang, X. Fang, Z. Wang, Y. Liu, and A. Nallanathan, “Stochastic game based cooperative alternating Q-learning caching in dynamic D2D networks,” *IEEE Trans. Veh. Technol.*, pp. 1–1, Oct. 2021, doi:10.1109/TVT.2021.3120292.

[11] F. Cheng, G. Gui, N. Zhao, Y. Chen, J. Tang, and H. Sari, “UAV-relaying-assisted secure transmission with caching,” *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3140–3153, Jan. 2019.

[12] D. Liu, B. Chen, C. Yang, and A. F. Molisch, “Caching at the wireless edge: design aspects, challenges, and future directions,” *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 22–28, Sep. 2016.

[13] N. Zhao, X. Liu, Y. Chen, S. Zhang, Z. Li, B. Chen, and M.-S. Alouini, “Caching D2D connections in small-cell networks,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12 326–12 338, Oct. 2018.

[14] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, Feb. 2018.

[15] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, “Overview of SHVC: Scalable extensions of the high efficiency video coding standard,” *IEEE Trans. Circuits Syst. for Video Technol.*, vol. 26, no. 1, pp. 20–34, Jul. 2015.

[16] X. Zhang, T. Lv, Y. Ren, W. Ni, N. C. Beaulieu, and Y. J. Guo, “Economical caching for scalable videos in cache-enabled heterogeneous networks,” *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1608–1621, Jul. 2019.

[17] J. Ma, L. Liu, H. Song, R. Shafin, B. Shang, and P. Fan, “Scalable video transmission in cache-aided Device-to-Device networks,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 4247–4261, Mar. 2020.

[18] Y. Liu, Z. Qin, Y. Cai, Y. Gao, G. Y. Li, and A. Nallanathan, “UAV communications based on non-orthogonal multiple access,” *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 52–57, Feb. 2019.

[19] J. Cui, Y. Liu, and A. Nallanathan, “Multi-agent reinforcement learning-based resource allocation for UAV networks,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 729–743, Feb. 2020.

[20] S.-F. Chou, A.-C. Pang, and Y.-J. Yu, “Energy-aware 3D unmanned aerial vehicle deployment for network throughput optimization,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 563–578, Oct. 2019.

[21] 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 Trans. Wireless Commun.*, vol. 18, no. 12, pp. 5831–5844, Sep. 2019.

[22] Y. Guo, S. Yin, and J. Hao, “Joint placement and resources optimization for multi-user UAV-relaying systems with underlaid cellular networks,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12 374–12 377, Aug. 2020.

[23] F. Zeng, Z. Hu, Z. Xiao, H. Jiang, S. Zhou, W. Liu, and D. Liu, “Resource allocation and trajectory optimization for QoE provisioning in energy-efficient UAV-enabled wireless networks,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7634–7647, Apr. 2020.

[24] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, “Trajectory design and power control for multi-UAV assisted wireless networks: A machine learning approach,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7957–7969, Aug. 2019.

[25] B. Khamidehi and E. S. Sousa, “Trajectory design for the aerial base stations to improve cellular network performance,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 945–956, Jan. 2021.

[26] J. Ji, K. Zhu, D. Niyato, and R. Wang, “Joint trajectory design and resource allocation for secure transmission in cache-enabled UAV-relaying networks with D2D communications,” *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1557–1571, Aug. 2021.

[27] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, “Wireless networks with cache-enabled and backhaul-limited aerial base stations,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7363–7376, Nov. 2020.

[28] M. Chen, W. Saad, and C. Yin, “Liquid state machine learning for resource and cache management in LTE-U unmanned aerial vehicle (UAV) networks,” *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1504–1517, Mar. 2019.

[29] J. Ji, K. Zhu, D. Niyato, and R. Wang, "Probabilistic cache placement in UAV-assisted networks with D2D connections: Performance analysis and trajectory optimization," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6331–6345, Jul. 2020.

[30] N. Zhao, F. Cheng, F. R. Yu, J. Tang, Y. Chen, G. Gui, and H. Sari, "Caching UAV assisted secure transmission in hyper-dense networks based on interference alignment," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2281–2294, May. 2018.

[31] T. Zhang, Z. Wang, Y. Liu, W. Xu, and A. Nallanathan, "Caching placement and resource allocation for cache-enabling UAV NOMA networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12 897–12 911, Nov. 2020.

[32] T. Zhang, Y. Wang, Y. Liu, W. Xu, and A. Nallanathan, "Cache-enabling UAV communications: Network deployment and resource allocation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7470–7483, Nov. 2020.

[33] T. Zhang, Z. Wang, Y. Liu, W. Xu, and A. Nallanathan, "Joint resource, deployment and caching optimization for ar applications in dynamic UAV NOMA networks," *IEEE Trans. Wireless Commun.*, pp. 1–1, Oct. 2021, doi:10.1109/TWC.2021.3121584.

[34] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, G. Feng, and S. Li, "Device-to-Device communications underlaying cellular networks," *IEEE Trans. Commun.*, vol. 61, no. 8, pp. 3541–3551, Jul. 2013.

[35] Q. Wu and R. Zhang, "Common throughput maximization in UAV-enabled OFDMA systems with delay consideration," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6614–6627, Aug. 2018.

[36] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: evidence and implications," in *Proc. IEEE INFOCOM*, vol. 1, Mar. 1999, pp. 126–134 vol.1.

[37] X. Zhang, T. Lv, Y. Ren, W. Ni, N. C. Beaulieu, and Y. J. Guo, "Economical caching for scalable videos in cache-enabled heterogeneous networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1608–1621, May. 2019.

[38] D. Gale and L. S. Shapley, "College admissions and stability of marriage," *American Mathematical Monthly*, vol. 69, pp. 9–15, May. 2013.

[39] J. Zhao, Y. Liu, K. K. Chai, Y. Chen, and M. Elkashlan, "Many-to-many matching with externalities for device-to-device communications," *IEEE Wireless Commun. Lett.*, vol. 6, no. 1, pp. 138–141, Dec. 2017.

[40] L. V. Stephen Boyd, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[41] S. B. Michael Gran, "Cvx: MATLAB software for disciplined convex programming," http://cvxr,com/cvx, 2016.

[42] Q. Li, A. M.-C. So, and W.-K. Ma, "Distributionally robust chance-constrained transmit beamforming for multiuser miso downlink," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May. 2014, pp. 3479–3483.

[43] T. Zhang, X. Fang, Y. Liu, and A. Nallanathan, "Content-centric mobile edge caching," *IEEE Access*, vol. 8, pp. 11 722–11 731, Dec. 2020.

[44] 3GPP, "Study on enhanced LTE support for aerial vehicles," *TR 36.777, Release 15*, pp. 33–46, Dec. 2017.

[45] X. Lin, V. Yajnanarayana, S. D. Muruganathan, S. Gao, H. Asplund, H.-L. Maattanen, M. Bergstrom, S. Euler, and Y.-P. E. Wang, "The sky is not the limit: Lte for unmanned aerial vehicles," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 204–210, Apr. 2018.