UAV Assisted Video Multicasting in 6G Networks: A Joint Caching and Trajectory Optimisation

Nguyen-Son Vo[1,#], Thuong C. Lam[2], Minh-Phung Bui[3], Thanh-Minh Phan[4] and Quang-Nhat Tran[3]

1 Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh City, Vietnam
2 HUTECH University, Ho Chi Minh City, Vietnam
3 Faculty of Information Technology, School of Engineering and Technology, Van Lang University, Ho Chi Minh City, Vietnam
4 Vietnam Aviation Academy, Ho Chi Minh City, Vietnam
Corresponding Author / E-mail: vonguyenson@duytan.edu.vn

KEYWORDS: 6G networks, caching and trajectory design, UAV communications, video multicasting applications and services

Abstract — In this paper, we design a joint caching and trajectory optimisation (CTO) solution for unmanned aerial vehicles (UAVs) assisted video multicasting (UVM) in 6G networks. To do so, the mobile users (MUs), who have the same interest of videos, are grouped into different clusters. We then formulate a CTO problem and solve it for the optimal number of videos to cache and the optimal set of UAV's stops to fly throughout, to serve the MUs in different clusters at maximum video delivery capacity. Simultaneously, a next shortest hop problem and the travelling salesman problem are solved for optimal trajectories, and thus reducing the flying energy consumption of the UAV. Simulation results are presented to analyse and demonstrate the benefits of the proposed CTO solution for the UVM in 6G networks.

Manuscript received: September, 2021 / Revised: January 01, 2022 / Accepted: February 15, 2022

1. Introduction

Recently, we have witnessed the successful launch of 5G networks that provide significant benefits for all the involved entities including mobile service providers, Internet service providers, content providers, and mobile users (MUs). However, to follow up with the proliferation of a massive number of MUs and a various amount of high data rate services and applications, 6G networks and technologies settle on studying immediately. Some disruptive solutions that can be developed for 6G include frequency expand, digital twin, artificial intelligence, big data, and especially the integration of a dynamic air platform, i.e., with satellites and unmanned aerial vehicles (UAVs) assisted, into the conventional static ground one [1].

In this paper, the benefits of UAVs with caching are exploited to provide the MUs with high data rate of video applications and services (VASs). The UAVs by flying, as an important part of the dynamic air platform, have a higher opportunity to gain the line-of-sight (LoS) propagation for better communications. The UAVs can also flexibly catch up with the characteristics of MUs, i.e., mobility, density, common interest, and social relationship, to satisfy the MUs. In addition, the UAVs can play as a relay network to expand the coverage of 6G networks. Importantly, by caching, the UAVs can bring the videos closer to the MUs to provide a higher quality of service.

In fact, caching techniques have been done at the static ground stations such as macro base stations (MBSs), small-cell base stations, and even mobile devices [2], [3]. Mostly at the same time, UAV caching has also drawn a significant attention from both the academic and industrial communities [4], [5], but not completed yet [6]. Particularly, the authors in [7] proposed a UAV caching scheme to find an optimal number of UAVs for caching a content. And then, an optimal fixed number of positions of UAVs are chosen for flying to serve the MUs in the hotspot areas at maximum quality of experience. More efficiently, the work in [8] further considered minimising the transmission power of the UAVs. Other studies allow the UAVs flying randomly, while optimising the caching placements to enhance the system capacity and energy efficiency [9], [10]. Interestingly, after identifying the probabilistic cache placements, the UAV flies over a minimum number of stops to maximise the system capacity while satisfying a given coverage area [11]. However, these studies have not considered all the aspects of VASs including video popularity, UAV-MU association, storage and energy resources of UAVs, to find the optimal stops and trajectory for serving the MUs the highest system capacity.

Motivated by the aforementioned discussions, we propose a joint caching and trajectory optimisation (CTO) solution for UAVs assisted video multicasting (UVM) in 6G networks. In this solution, we first group the MUs into different clusters, each cluster has the same video interest. The number of clusters requesting a particular video depends on the video's popularity. Then, we formulate the CTO problem and solve it for optimally selecting which videos to cache and where to stop based on the popularity pattern of videos, the request number of clusters for each video, storage and energy resources of the UAV, and wireless channel characteristics between the UAV and the clusters. In addition, we consider two trajectory modes by solving a next shortest hop (NSH) problem and the travelling salesman (TSM) problem. As a result, the CTO provides the MUs in different clusters with the highest video delivery capacity while conserving the storage and energy resources of UAV.

The rest of this paper is organised as follows. In Section 2, we introduce the UVM system and describe how it works. The system is formulated in Section 3. The formulations enable us to derive the CTO problem and solution presented in Section 4. Section 5 is dedicated to providing the simulation results, analysis, and evaluation. Finally, Section 6 concludes the paper with future research directions.

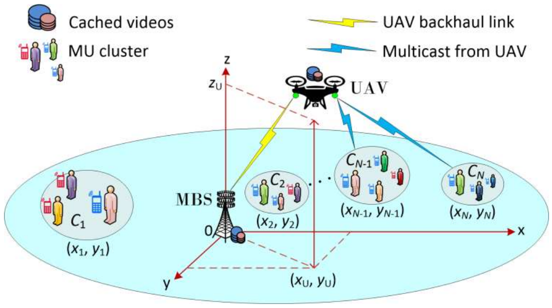

Fig. 1 A UVM system in 6G networks.

2. System Model

In this paper, we consider a UVM system in 6G networks as shown in Fig. 1. The system consists of one MBS, one UAV, N clusters, and I videos. The MBS is located at the origin of the horizontal coordinate. The UAV is assumed to fly at the same vertical coordinate of $z_{\mathrm{U}} = h_{\mathrm{U}}$ over the horizontal coordinate of $o_{\mathrm{U}} = (x_{\mathrm{U}}, y_{\mathrm{U}})$. It has a limit energy of $E^{\star}$ (Joules) for flying, transmitting, and other related tasks, and a limit storage of $S^{\star}$ (Mbits) for caching the videos. In the cluster n, $n = 1, 2, \ldots, N$, the MUs in close proximity, which have the same interest of video i, $i = 1, 2, \ldots, I$, are located around the horizontal coordinate center of $o_n = (x_n, y_n)$. In addition, the video i has a size of s_i (Mbits) and a popularity/access rate of p_i. The MBS always listens the demand of the MUs, collaborates with the UAVs to cache the potential requested videos, and then it performs the CTO solution to serve the MUs by performing the following three steps.

1. Clustering: In this step, the MBS groups the MUs into N clusters by using the device-to-device (D2D) clustering scheme proposed in [12]. This scheme enables the MUs, who gather and share their works and entertainment with each other by D2D communications, to request the same video from the MBS or the UAV.

2. Caching and flying for multicasting: In this step, depending on the storage capacity of the UAV, the video popularity pattern, and the request number of clusters for each video, the UAV selects a proper number of videos to cache in advance. Then, based on the energy resource of the UAV, the MBS formulates the CTO problem. Solving this problem, the UAV is able to fly throughout the optimal stops and follow the optimal trajectory to maximise the video delivery capacity, while satisfying the constraints on storage and energy of the UAV for resource savings.

3. Returning: In this step, at the last stop, the UAV keeps standstill in a short time for the next assignment from the MBS if required. Otherwise, it flies to the nearby charging station.

3. System Formulations

In this section, we formulate the system to derive the mathematical models of requesting and caching, trajectory, video delivery capacity and energy consumption. These formulations, which are used as the objective function and the constraints in the CTO problem, are presented below.

3.1. Requesting and Caching

Given the video popularity pattern, the number of clusters that requests the video i is proportional to the popularity of this video, given by

$$N_i = p_i N, \tag{1}$$

where p_i is the popularity of the video i, which follows the Zipf-like distribution [13] expressed as

$$p_i = \frac{i^{-\alpha}}{\sum_{i=1}^{I} i^{-\alpha}}, \tag{2}$$

here $\alpha \geq 0$ is the skewed coefficient used to reflect the skewed popularity pattern of a given video set. For example, $\alpha = 0$ means that all videos have the same popularity. Meanwhile, the higher value of α yields the higher skewed popularity between the first videos and the rear ones.

In (1), because N_i is an integer, without loss of generality, we further adjust it to $N_i = \lceil p_i N \rfloor$, where the round operator $\lceil . \rfloor$ is used to round a given value to the nearest integer. In addition, we define $r_{i,n}$ as a request index to identify that the cluster n requests the video i ($r_{i,n} = 1$) or not ($r_{i,n} = 0$) so that $N_i = \sum_{n=1}^{N} r_{i,n}$. However, this may cause $\sum_{n=1}^{N} \sum_{i=1}^{I} r_{i,n} < N$. To deal with this problem, the first smallest value of $\{N_i\}$ is increased by 1 until $\sum_{n=1}^{N} \sum_{i=1}^{I} r_{i,n} = N$.

The UAV selects a number of videos which are mostly requested by the MUs for caching, and thus we have

$$\sum_{\substack{i=1 \\ r_{i,n}=1, \forall n}}^{I} q_i s_i \leq S^{\star}, \tag{3}$$

where q_i is the caching index to identify that if the video i is cached in the UAV ($q_i = 1$) or not ($q_i = 0$) and it is noted in (3) that because Zipf-like distribution generates $p_1 \geq p_2 \geq \ldots \geq p_i \geq \ldots \geq p_I$, the UAV always takes priority on caching the videos with higher popularity. However, this is the pre-caching decision making because the final decision further depends on the remaining energy of the UAV. For example, the UAV does not decide to cache a video even with very high popularity if this video is requested by the MUs located too far from the UAV's location, meanwhile the remaining energy of the UAV is not enough to transmit this video.

3.2. Average Video Delivery Capacity

Video delivery capacity is considered as the objective function of the CTO problem. The UAV takes priority over the MBS to serve the MUs in the clusters. The average video capacity delivered to the MUs, which comes from the UAV based on the wireless channel model, is computed below.

Assuming that the UAV utilises the inband frequency (licensed band) for the backhaul link, but outband frequency (unlicensed band) for communicating with the MUs. Furthermore, all the MUs in a cluster have the same channel capacity, and thus the capacity to deliver the video i from the UAV located at $o_{\mathrm{U}} = o_{l,w}$ to the MUs in the cluster n located at o_n is given by

$$C_{l,w}^{i,n} = B \log_2(1 + \gamma_{l,w}^{i,n}), \tag{4}$$

where B is the system bandwidth and $\gamma_{l,w}^{i,n}$, which is the signal-to-noise ratio of the channel between the UAV at $o_{l,w}$ and the cluster at o_n, is dominated by the LoS link [14], [15], expressed as

$$\gamma_{l,w}^{i,n} = \frac{q_i r_{i,n} P_{\mathrm{T}} \beta_0^{\mathrm{U}} (d_{l,w}^{n})^{-2}}{\sigma^2}. \tag{5}$$

In (5), $q_i r_{i,n}$ is the UAV-cluster association index, P_{T} is the transmission power of the UAV, β_0^{U} is the UAV-to-cluster channel power gain at the reference distance of $d_0 = 1\mathrm{m}$, σ^2 is the additive white Gaussian noise power, and $d_{l,w}^{n}$ is the distance between the UAV at $o_{l,w}$ and the cluster at o_n computed as

$$d_{l,w}^{n} = \sqrt{h_{\mathrm{U}}^{2} + \| o_{l,w} - o_n \|^{2}}, \tag{6}$$

where $\|.\|$ is the Euclidean norm.

Finally, the average video capacity delivered from the UAV to all the cluster is given by

$$\overline{C} = \frac{1}{N} \sum_{n=1}^{N} \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^{I} p_i f(o_{l,w}) C_{l,w}^{i,n}, \tag{7}$$

where, $f(o_{l,w})$ is the stop index defined as a binary variable to identify that the UAV decides to stop ($f(o_{l,w}) = 1$) or not to stop ($f(o_{l,w}) = 0$) at $o_{l,w}$ so that $\sum_{l=1}^{M_L} \sum_{w=1}^{M_W} f(o_{l,w}) \leq N$, $M_L \times M_W$ is the total number of stops located in a rectangular grid of length M_L stops and width M_W stops. The CTO problem is solved to maximise $\overline{C}$ by finding the optimal caching index q_i and stop index $f(o_{l,w})$ as well as how to fly throughout the stops with minimum energy consumption.

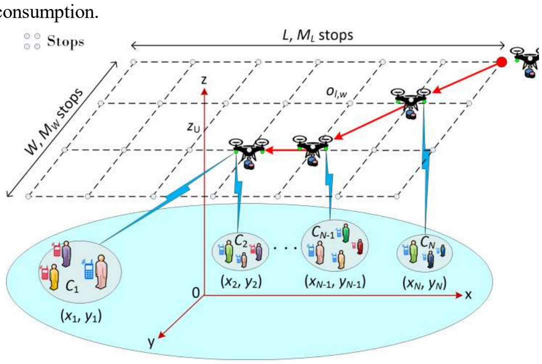

Fig. 2 UAV trajectory.

Algorithm 1 Next Shortest Hop Trajectory

Input: $o_{l,w}$, initial horizontal coordinate of UAV (x_o, y_o)

Output: $d_{\min}, \mathcal{T}$

1: $\mathcal{M} = \{f(o_{l,w}) = 1, l = 1, 2, \ldots, M_L, w = 1, 2, \ldots, M_W\}$

2: $current\_stop \leftarrow (x_o, y_o)$

3: $\mathcal{T} = \{current\_stop\}$

4: $d_{min} = 0$

5: **while** $\mathcal{M} \neq \varnothing$ **do**

6: $next\_stop = \mathrm{argmin}\{d(current\_stop, \mathcal{M})\}$

7: $\mathcal{T} \leftarrow \mathcal{T} \cup next\_stop$

8: $d_{min} = d_{min} + d(current\_stop, next\_stop)$

9: $\mathcal{M} = \mathcal{M} \backslash next\_stop$

10: $current\_stop \leftarrow next\_stop$

11: **end while**

3.3. Trajectory

For the ease of formulating the trajectory, the UAV plans to fly at a fixed velocity V (m/s) within a smallest rectangular that covers all the centers of the clusters. The rectangular is made up of a grid as shown in Fig. 2. The cross point of each pair of straight lines is a stop in which the UAV flies throughout or stays for a while if needed to multicast a video. The rectangular has a length of L (m) and a width of W (m) including M_L stops and M_W stops, respectively. The number of stops estimated depends on the density of clusters.

Because the stops have the same vertical coordinates with the UAV, we only consider the horizontal coordinates of the stops which are located at $o_{l,w} = (x_l, y_w)$, $l = 1, 2, \ldots, M_L$, $w = 1, 2, \ldots, M_W$. Therefore, to plan for flying from the initial coordinate (x_o, y_o) of the UAV, the UAV has $2^{M_L \times M_W}$ trajectories corresponding to the number of combinations of the $M_L \times M_W$ binary matrix. Here, each matrix includes a set of row stops and column stops valued by $f(o_{l,w})$. After caching, the best trajectory is selected to fly throughout the stops with $f(o_{l,w}) = 1$ to serve the MUs in all clusters at maximum video delivery capacity, while satisfying the constraint on energy consumption. In addition, the best trajectory enables the UAV to fly over the shortest path ($d_{\min}$) to save the energy by finding the set $\mathcal{T}$ of next stops in sequence by solving the well-known TSM problem using exhaustive search (ES). However, the ES generates a very high complexity. Therefore, we propose a simple next shortest hop (NSH) method as described in Algorithm 1.

In Algorithm 1, the set of stops ($\mathcal{M}$) with $f(o_{l,w}) = 1$ is given in the line 1. The line 2, line 3, and line 4 identify that the initial set $\mathcal{T}$ is the current stop of the UAV (x_o, y_o) and the initial distance to fly is $d_{min} = 0$. From the current stop, the next stop is the one in the set $\mathcal{M}$ that has the minimum distance to the current stop (line 6). The next stop in the trajectory is added to the set $\mathcal{T}$ in line 7. The distance is prolonged to fly to the next stop (line 8). The line 9 and line 10 are to update the current stop and the set $\mathcal{M}$.

3.4. Energy Consumption

In this paper, we consider the three important types of energy consumed by flight, transmission, and standstill. To compute the energy consumption, we first derive the corresponding duration of flight and transmission/standstill, here the standstill duration is for transmission. Given the shortest path d_{min} found by using the Algorithm 1 and by solving the TSM problem, the flight duration is computed as

$$t_F = d_{min}/V. \tag{8}$$

Furthermore, the standstill (or transmission) duration is given by

$$t_T = \sum_{n=1}^{N} \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^{I} \frac{q_i r_{i,n} f(o_{l,w}) s_i}{C_{l,w}^{i,n}}. \tag{9}$$

It is noted that if $q_i r_{i,n} = 0$, the computation of (9) cannot be done due to the division by zero, i.e., $C_{l,w}^{i,n} = 0$. To avoid this problem, we further define an infinitesimal value ϵ and rewrite (9) as below

$$t_T = \sum_{n=1}^{N} \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^{I} \frac{q_i r_{i,n} f(o_{l,w}) s_i}{\max\{\epsilon, C_{l,w}^{i,n}\}}. \tag{10}$$

By taking into account the flying power (P_{F}), standstill power (P_{S}), and transmission power (P_{T}), the total energy consumption for a particular trajectory, which is less than a given threshold $E^{\star}$, is expressed as

$$E = P_{\mathrm{F}} t_F + (P_{\mathrm{S}} + P_{\mathrm{T}}) t_T \leq E^{\star}. \tag{11}$$

The objective of (11), considered as the constraint in the CTO problem, is to limit the trajectory of the UAV when serving the MUs in different clusters.

4. CTO Problem and Solution

So far, we have derived the objective function (7) and the constraints on caching storage (3) and energy consumption (11). Mathematically, the CTO problem is formulated as below

$$\max_{q_i, f(o_{l,w})} \overline{C} \tag{12a}$$

$$\text{s.t} \quad (3) \tag{12b}$$

$$(11)$$

$$\sum_{l=1}^{M_L} \sum_{w=1}^{M_W} f(o_{l,w}) \leq N$$

Algorithm 2 Exhaustive Search

Input: Initial system parameters
Output: $C^\star$, $Q^\star$, $F^\star_{M_L \times M_W}$

```
𝒞 ← ∅
for each vector Q in 𝒬 do
    for each matrix F_{M_L×M_W} in ℱ do
        if (12b) holds then
            C(Q, F_{M_L×M_W}) = C̄
            𝒞 ← 𝒞 ∪ C(Q, F_{M_L×M_W})
        end if
    end for
end for
C* = max 𝒞
{Q*, F*_{M_L×M_W}} = argmax 𝒞
```

Table 1 Parameters Setting

Symbols	Specifications
N	10 clusters
I	5 videos
s_i	[300, 100, 200, 400, 900, 500, 700, 600, 750, 950] Mbits
$S^\star$	2×10³ Mbits (2Gbits)
α	1
V	15 m/s
P_T	0.5 W
P_F	50 W
P_S	0.25× P_F
$E^\star$	5000 Joules
B	10 MHz
β_0^U	10^{-5}
σ^2	10^{-13} W

Solving (12) for optimal caching index q_i and stop index $f(o_{l,w})$, the UAV knows which videos to cache, where to stop, and how to fly throughout the stops, so as to serve the MUs in all clusters the maximum average video delivery capacity, while satisfying the constraints on caching storage and energy consumption. It is easy to see that finding the optimal values of q_i and $f(o_{l,w})$ is equivalent to finding the binary vector $Q^\star$ of length I bits and the binary matrix $F^\star_{M_L \times M_W}$ of M_L rows and M_W columns. Here, the searching space respectively includes $\mathcal{Q} = \{Q^1, Q^2, \dots, Q^{2^I}\}$ and $\mathcal{F} = \{F^1_{M_L \times M_W}, F^2_{M_L \times M_W}, \dots, F^{2^{M_L \times M_W}}_{M_L \times M_W}\}$. To solve (12), we simply apply exhaustive search (ES) presented in Algorithm 2 for the exact solutions of $Q^\star$ and $F^\star_{M_L \times M_W}$.

5. Performance Evaluation

To analyse and evaluate the performance of the proposed CTO solution, we deploy the system using the parameters listed in Table 1. Furthermore, the clusters are uniformly randomly distributed within a 100-meter radius circle covered by the MBS. The UAV flies at the vertical coordinate of $h_U = 50$m starting at the upper right corner of the rectangular plane.

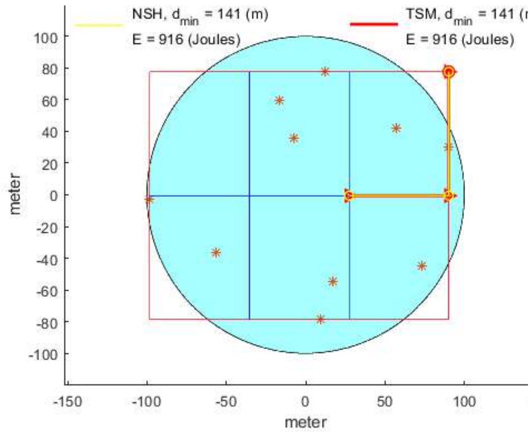

Fig. 3 UAV trajectory versus $E^\star = 1000$ Joules.

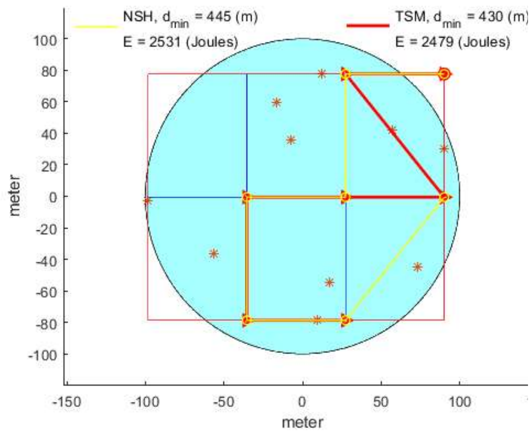

Fig. 4 UAV trajectory versus $E^\star = 2500$ Joules.

We first evaluate the TSM and the NSH by setting the energy constraint $E^\star$ to 1000 Joules, 2500 Joules, and 5000 Joules, respectively shown in Fig. 3, Fig. 4, and Fig. 5. In Fig. 3, because the shortest trajectory is too simple limited by the low value of energy $E^\star = 1000$ Joules, the trajectories of TSM and NSH are the same. However, when relaxing $E^\star$, the trajectories become more complicated, it is clear that the TSM using ES yields the exact shortest trajectory, and it consumes lower energy than the NSH does (Fig. 4 and Fig. 5). Obviously, the energy consumption of TSM is always less than or equal to $E^\star$, but it is not guaranteed by the NSH (Fig. 4). We also observe that although the NSH provides longer trajectory, the difference compared to the TSM is not significant. In a large-scale system, the NSH can be applied to finding the shortest trajectory thanks to its less time and memory complexity.

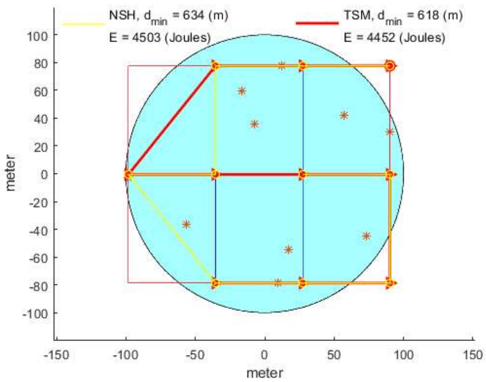

Fig. 5 UAV trajectory versus $E^{\star} = 5000$ *Joules.*

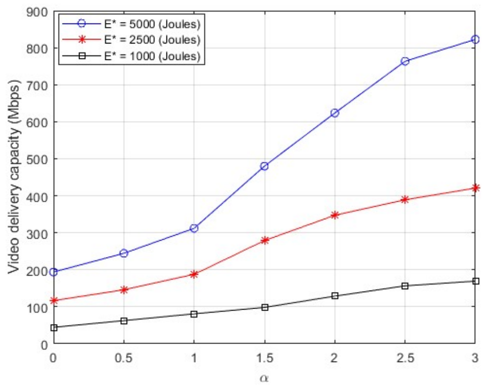

Fig. 6 Average video delivery capacity versus α *and* $E^{\star}$.

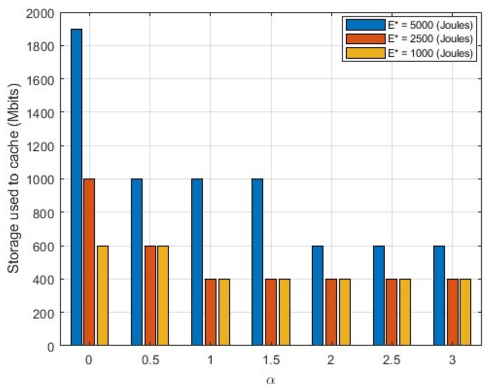

Fig. 7 Storage used to cache versus α *and* $E^{\star}$.

The performance of the proposed solution is then evaluated by considering the average video delivery capacity versus α and $E^{\star}$ as shown in Fig. 6. The increase of $E^{\star}$ enables the UAV to fly throughout more stops to serve the MUs better, with higher video delivery capacity. In addition, increasing α lets the UAV focus on caching the videos with much higher popularity to efficiently multicast them to the MUs in different clusters. This in turn provides the MUs with much higher video delivery capacity when α increases, but lower storage is used for caching as depicted in Fig. 7. The reason here is that the higher value of α causes the higher skewed popularity between the first videos and the rear ones. It results in the fact that more clusters request the first videos. Therefore, the UAV places a high priority on caching the first videos with less storage used, but ensuring to serve the MUs in all the clusters more efficiently. In addition, we can see that the higher energy utilised enables the UAV to cache more videos to enhance the video delivery capacity.

6. Conclusion

We have proposed the joint caching and trajectory optimisation (CTO) solution for unmanned aerial vehicles (UAVs) to multicast the videos in 6G networks. The CTO is able to provide the mobile users (MUs) in different clusters with the highest video delivery capacity, while conserving the storage and energy resources of the UAV. We have also analysed some initial results and demonstrated the benefits of the proposed CTO solution. In future works, we will design a network of UAVs for caching, flying, and delivering the videos to serve the MUs more efficiently.

REFERENCES

1. Strinati E. et al., “6G in the sky: On-demand intelligence at the edge of 3D networks,” *ETRI Journal*, vol. 42, no. 5, pp. 643–657, Oct. 2020.
2. Bui, M. P., Vo, N. S., Nguyen, S. Q. and Tran, Q. N., “Social-aware caching and resource sharing maximized video delivery capacity in 5G ultra-dense networks,” *Mobile Networks and Applications*, vol. 25, pp. 2037–2049, July 2019.
3. Vo, N. S., Bui, M. P., Truong, P. Q., Yin, C. and Masaracchia, A., “Multi- tier caching and resource sharing for video streaming in 5G ultra-dense networks,” *IEEE Commun. Lett.*, vol. 24, no. 7, pp. 1500–1504, July 2020.
4. Cheng, F., Zhang, S., Li, Z., Chen, Y., Zhao, N., Yu, F. R. and Leung, V. C. M., “UAV trajectory optimization for data offloading at the edge of multiple cells,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6732–6736, July 2018.
5. Yang, J., Xiao, S., Jiang, B., Song, H., Khan, S. and ul Islamo, S., “Cache-enabled unmanned aerial vehicles for cooperative cognitive radio networks,” *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 155–161, Apr. 2020.
6. Duong, T. Q., Kim, K. J., Kaleem, Z., Bui, M. P. and Vo, N. S., “UAV caching in 6G networks: A survey on models, techniques, and applica- tions,” *Physical Communication*, vol. 51, pp. 1-19, Apr. 2022.
7. Zhang, T., Wang, Y., Liu, Y., Xu, W. and Nallanathan, A., “Cache-enabling UAV communications: Network deployment and resource allocation*,” IEEE Trans. Wireless Commun*., vol. 19, no. 11, pp. 7470–7483, Nov. 2020.
8. Chen, M., Mozaffari, M., Saad, W., Yin, C., Debbah, M. and Hong, C. S., “Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience,” *IEEE J. Sel. Areas Commun*., vol. 35, no. 5, pp. 1046–1061, May 2017.
9. Kalinagac, O., Kafiloglu, S. S., Alagoz, F. and Gur, G., “Caching and D2D sharing for content delivery in software-defined UAV

networks," in Proc. IEEE Veh. Technol. Conf., Honolulu, HI, Sep. 2019, pp. 1–5.

10. Khoshkholgh, M. G., Navaie, K., Yanikomeroglu, H., Leung, V. C. M. and Shin, K. G., "Randomized caching in cooperative UAV-enabled fog-RAN," in *Proc. IEEE Wireless Commun. and Networking Conf.*, Marrakesh, Morocco, Apr. 2019, pp. 1–6.
11. Ji, J., Zhu, K., Niyato, D. and Wang, R., "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, Oct. 2020.
12. Gyawali, S., Xu, S., Ye, F., Hu, R. Q. and Qian, Y., "A D2D based clustering scheme for public safety communications," in *Proc. of IEEE 87th Vehicular Technology Conference*, Porto, Portugal, 2018, pp. 1–5.
13. Breslau, L., Cao, P., Fan, L., Phillips, G. and Shenker, S., "Web caching and zipf-like distributions: Evidence and implications," New York, NY, Mar. 1999, pp. 126–134.
14. Xu, X., Zeng, Y., Guan, Y. L. and Zhang, R., "Overcoming endurance issue: UAV-enabled communications with proactive caching," *IEEE J. Sel. Areas Commun.*, pp. 1231–1244, June 2018.
15. Wu, H., Lyu, F., Zhou, C., Chen, J., Wang, L. and Shen, X., "Optimal UAV caching and trajectory in aerial-assisted vehicular networks: A learning- based approach," *IEEE J. Sel. Areas Commun.*, pp. 1–14, June 2020.