

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC DUY TÂN

LÂM CHI THƯƠNG

**TỔNG HỢP CÁC CÔNG BỐ LIÊN QUAN
LUẬN ÁN TIẾN SĨ KỸ THUẬT**

**CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH
MÃ SỐ CHUYÊN NGÀNH: 9.48.01.01**

**NGHIÊN CỨU KỸ THUẬT LƯU TRỮ
VÀ PHÂN PHỐI DỮ LIỆU TỐI ƯU THỜI GIAN THỰC
DÙNG UAV TRONG MẠNG 6G IoT**

ĐÀ NẴNG, 2026

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC DUY TÂN

LÂM CHI THƯƠNG

**TỔNG HỢP CÁC CÔNG BỐ LIÊN QUAN
LUẬN ÁN TIẾN SĨ KỸ THUẬT**

**CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH
MÃ SỐ CHUYÊN NGÀNH: 9.48.01.01**

**NGHIÊN CỨU KỸ THUẬT LƯU TRỮ
VÀ PHÂN PHỐI DỮ LIỆU TỐI ƯU THỜI GIAN THỰC
DÙNG UAV TRONG MẠNG 6G IoT**

ĐÀ NẴNG, 2026

DANH MỤC CÁC BÀI BÁO LIÊN QUAN ĐẾN LUẬN ÁN TIẾN SĨ

- [1] Nguyen-Son Vo, **Thuong C Lam**, Minh-Phung Bui, Thanh-Minh Phan, Quang-Nhat Tran, " UAV Assisted Video Multicasting in 6G Networks: A Joint Caching and Trajectory Optimisation," *Journal of Aviation Science & Technology*, vol. 1, no. 1, pp. 37-42, Mar. 2022;
- [2] **Thuong C. Lam**, Nguyen-Son Vo, Thanh-Hieu Nguyen, Thanh-Minh Phan, and De-Thu Huynh, "Genetic Algorithms for Storage- and Energy-Aware Caching and Trajectory Optimisation Problem in UAV-Assisted Content Delivery Networks", in *Proc. of EAI International Conference on Industrial Networks and Intelligent Systems (INISCOM '23)*, Ho Chi Minh City, Vietnam, Aug. 2023, pp. 34-50
- [3] **Thuong C. Lam**, Nguyen-Son Vo, Minh-Phung Bui, Chan Dai Truyen Thai, Haejoon Jung, Van-Ca Phan, "Service Time-aware Caching, Power Allocation, and 3D Trajectory Optimised Multimedia Content Delivery in UAV-assisted IoT Networks", *IEEE Transactions on Vehicular Technology*, Dec. 2024.
- [4] **Thuong C. Lam**, Nguyen-Son Vo, Minh-Phung Bui, De-Thu Huynh, Thanh-Dung Tran, and Ha Duy Hung, "Genetic Algorithms for Swarm UAV-Assisted Edge Content Delivery Optimisation Problem", *2024 RIVF International Conference on Computing and Communication Technologies (RIVF)*, Da Nang, Vietnam, Dec. 2024.
- [5] **Thuong C. Lam**, Nguyen-Son Vo, Tan Do-Duy, Long D. Nguyen, Thanh-Minh Phan, Hung T. Nguyen, and De-Thu Huynh "Transmission Mode Selection-based Content Caching for Real-time Swarm UAV-assisted 6G IoT Networks", *EAI INISCOM 2026, 12th EAI International Conference on Industrial Networks and Intelligent Systems*, Da Nang, Vietnam, Feb. 2026 (*accepted*).

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

Nguyen-Son Vo^{1,#}, Thuong C. Lam², Minh-Phung Bui³, Thanh-Minh Phan⁴ and Quang-Nhat Tran³

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: vonguyensson@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.

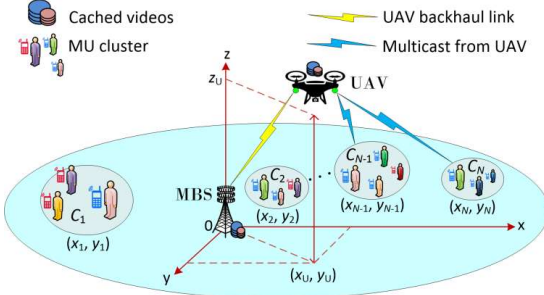


Fig. 1 A UVM system in 6G networks.

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 $z_U = h_U$ over the horizontal coordinate of $o_U = (x_U, y_U)$. It has a limit energy of E^* (Joules) for flying, transmitting, and other related tasks, and a limit storage of S^* (Mbits) for caching the videos. In the cluster n , $n = 1, 2, \dots, N$, the MUs in close proximity, which have the same interest of video i , $i = 1, 2, \dots, I$, 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

$$N_i = p_i N, \quad (1)$$

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

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

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 $N_i = \lceil p_i N \rceil$, where the round operator $\lceil \cdot \rceil$ 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

$$\sum_{i=1}^I q_i s_i \leq S^*, \quad (3)$$

$r_{i,n}=1, \forall n$

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 $p_1 \geq p_2 \geq \dots \geq p_i \geq \dots \geq p_I$, 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 $o_U = o_{l,w}$ to the MUs in the cluster n located at o_n is given by

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

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

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

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 $d_0 = 1\text{m}$, 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

$$d_{l,w}^n = \sqrt{h_0^2 + \|o_{l,w} - o_n\|^2}, \quad (6)$$

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

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

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

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 $\bar{C}$ 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.

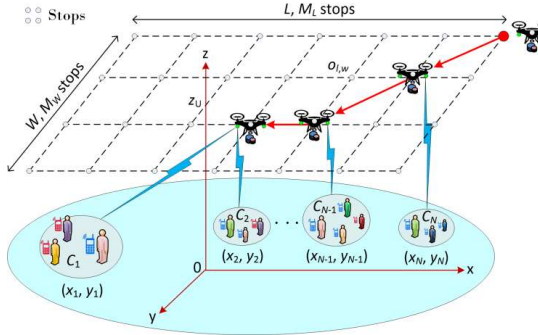


Fig. 2 UAV trajectory.

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: $\mathcal{M} = \{f(o_{l,w}) = 1, l = 1, 2, \dots, M_L, w = 1, 2, \dots, M_W\}$
- 2: $current_stop \leftarrow (x_o, y_o)$
- 3: $\mathcal{T} = \{current_stop\}$
- 4: $d_{min} = 0$
- 5: **while** $\mathcal{M} \neq \emptyset$ **do**
- 6: $next_stop = \text{argmin}\{d(current_stop, \mathcal{M})\}$
- 7: $\mathcal{T} \leftarrow \mathcal{T} \cup next_stop$
- 8: $d_{min} = d_{min} + d(current_stop, next_stop)$
- 9: $\mathcal{M} = \mathcal{M} \setminus next_stop$
- 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 $o_{l,w} = (x_l, y_w)$, $l = 1, 2, \dots, M_L$, $w = 1, 2, \dots, M_W$. 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

$$t_F = d_{min}/V. \quad (8)$$

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

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

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

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

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^* , is expressed as

$$E = P_F t_F + (P_S + P_T) t_T \leq E^*. \quad (11)$$

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

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

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

$$(11)$$

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

Algorithm 2 Exhaustive Search

Input: Initial system parameters

Output: $C^*, Q^*, F_{M_L \times M_W}^*$

```

1:  $\mathcal{C} \leftarrow \emptyset$ 
2: for each vector  $Q$  in  $\mathcal{Q}$  do
3:   for each matrix  $F_{M_L \times M_W}$  in  $\mathcal{F}$  do
4:     if (12b) holds then
5:        $C(Q, F_{M_L \times M_W}) = \bar{C}$ 
6:        $\mathcal{C} \leftarrow \mathcal{C} \cup C(Q, F_{M_L \times M_W})$ 
7:     end if
8:   end for
9: end for
10:  $C^* = \max \mathcal{C}$ 
11:  $\{Q^*, F_{M_L \times M_W}^*\} = \operatorname{argmax} \mathcal{C}$ 

```

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^*	2×10^3 Mbits (2Gbits)
α	1
V	15 m/s
P_T	0.5 W
P_F	50 W
P_S	$0.25 \times P_F$
E^*	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^* of length I bits and the binary matrix $F_{M_L \times M_W}^*$ 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 $\mathcal{F} = \{F_{M_L \times M_W}^1, F_{M_L \times M_W}^2, \dots, F_{M_L \times M_W}^{2^{M_L \times M_W}}\}$. To solve (12), we simply apply exhaustive search (ES) presented in Algorithm 2 for the exact solutions of Q^* and $F_{M_L \times M_W}^*$.

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.

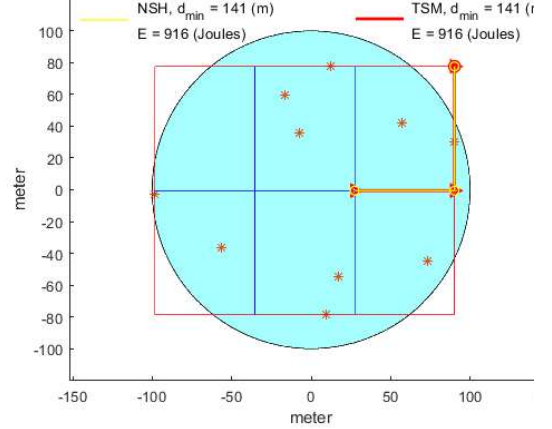


Fig. 3 UAV trajectory versus $E^* = 1000$ Joules.

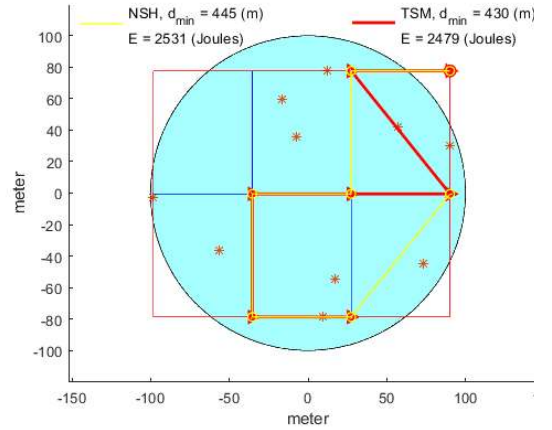


Fig. 4 UAV trajectory versus $E^* = 2500$ Joules.

We first evaluate the TSM and the NSH by setting the energy constraint E^* 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 $E^* = 1000$ Joules, the trajectories of TSM and NSH are the same. However, when relaxing E^* , 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^* , 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.

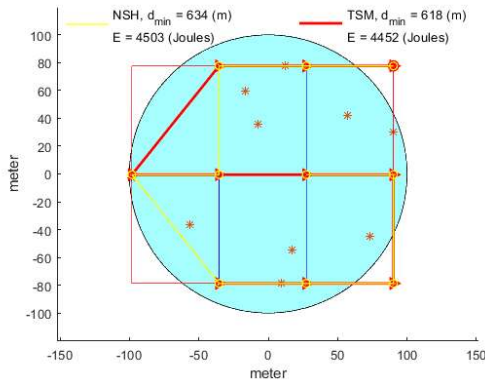


Fig. 5 UAV trajectory versus $E^* = 5000$ Joules.

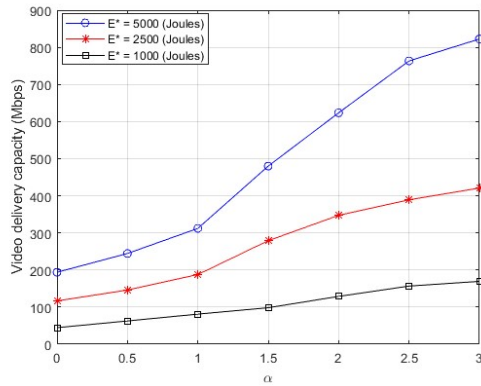


Fig. 6 Average video delivery capacity versus α and E^* .

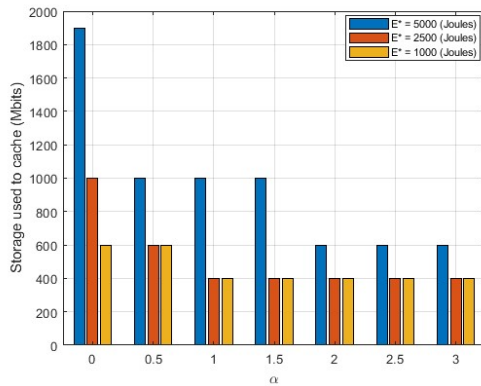


Fig. 7 Storage used to cache versus α and E^* .

The performance of the proposed solution is then evaluated by considering the average video delivery capacity versus α and E^* as shown in Fig. 6. The increase of E^* 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 applications," *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.



Genetic Algorithms for Storage- and Energy-Aware Caching and Trajectory Optimisation Problem in UAV-Assisted Content Delivery Networks

Thuong C. Lam¹, Nguyen-Son Vo^{2(✉)}, Thanh-Hieu Nguyen³,
Thanh-Minh Phan⁴, and De-Thu Huynh⁵

¹ HUTECH University, Ho Chi Minh City, Vietnam

lc.thuong@hutech.edu.vn

² Institute of Fundamental and Applied Sciences, Duy Tan University,
Ho Chi Minh City, Vietnam

vonguyenson@duytan.edu.vn

³ Ho Chi Minh City University of Transport, Ho Chi Minh City, Vietnam

thanhhieunguyen@ut.edu.vn

⁴ Vietnam Aviation Academy, Ho Chi Minh City, Vietnam

minhpt@vaa.edu.vn

⁵ The Saigon International University, Ho Chi Minh City, Vietnam

huynhdethu@siu.edu.vn

Abstract. Trajectory and caching optimisation design is a promising joint solution for enhancing the quality of services in unmanned aerial vehicle (UAV) assisted content delivery networks (CDNs). In this paper, we review the problem of which contents to cache in the UAV and which trajectory to fly, i.e., where to stop and how to gain the shortest path over the stops, under the constraints of caching storage and energy resources, namely storage- and energy-aware caching and trajectory optimisation (SECTO) problem. The SECTO problem in UAV-assisted CDNs is formulated and solved by applying genetic algorithms (GAs) to maximise the content delivery capacity while minimising the flying distance. The simulation results are shown to demonstrate the benefits of GAs in terms of accuracy and time complexity performance compared to other conventional solutions such as exhausted and greedy search algorithms.

Keywords: Content delivery network · Genetic algorithm · travelling salesman problem · UAV caching · UAV trajectory

1 Introduction

One of the most important missions of 6G networks is to take the non-terrestrial networks (NTNs) into account [1]. The NTNs enable a flying platform of satellites and unmanned aerial vehicles (UAVs) to connect more things in every remote

corner on the earth. And thus, the physical, cyber, and biological worlds can further flourish in the era of 6G thanks to the present of flexible access points, relays, contents, helps, and many advanced applications and services (AASs) from the sky [2–6]. In this scenario, the models, techniques, and AASs of UAV caching and trajectory in content delivery networks (CDNs) have drawn a numerous studies from both industrial and academic sectors [7]. These studies demonstrate that UAV caching networks are able to partially alternate the CDNs so as to not only mitigate the workload of the terrestrial stations but also provide the end users (EUs) with better AASs by exploiting shorter and line-of-sight (LoS) transmission.

In UAV caching networks, the UAVs have to select a set of contents to cache in advance. It then flies to deliver the contents to the EUs in the error-prone areas in which, for example, the workload of the terrestrial stations is extremely high, the wireless links are not good due to high and dense obstacles, and the emergency communications are required for rescue operations and safety public activities. To serve the EUs the highest quality of service (QoS) while utilising the resources of UAVs, the optimal set of contents and the optimal trajectory must be found simultaneously. This so-called resource-aware caching and trajectory optimisation problem in UAV-assisted CDNs has been studied in the literature by different approaches from objective functions, constraints, to algorithms and solutions [8–12].

Particularly, the work in [8] proposed a storage- and energy-aware caching and trajectory optimisation (SECTO) problem to maximise the content delivery capacity while minimising the flight distance under the caching storage and energy consumption constraints. The SECTO is solved by using exhausted and greedy search algorithms. In [9], the authors tried to minimise only the delay under the constraint of caching storage by solving a tractable semi-definite programming problem reformulated from an original intractable distributionally robust stochastic optimization problem. Besides caching in the UAVs, the authors in [10] considered caching in the EUs to establish a cache-enabling UAV-device-to-device (UAV-D2D) network. A joint caching placement in UAV-D2D network and UAV trajectory optimization (CTO) problem under the constraint of caching storage is formulated for maximising the cache utility. The CTO problem, which is a mixed integer nonlinear programming problem, is decomposed into three sub-problems for being solved by a low complexity iterative algorithm. Interestingly, reinforcement learning approach is applied to minimise the sum content acquisition delay of EUs by jointly optimising the multiple EUs' association, cache placement, UAV trajectory, and transmission power [11, 12].

We can see that the aforementioned optimisation solutions are for whether multiple objective functions of very high time complexity but exact results [8] or single objective function of lower time complexity with approximate results by reformulating and decomposing the original problems into solvable sub-problems [9–12]. However, these solutions are not flexible in terms of balancing the trade-off between accuracy and time complexity in accordance with the

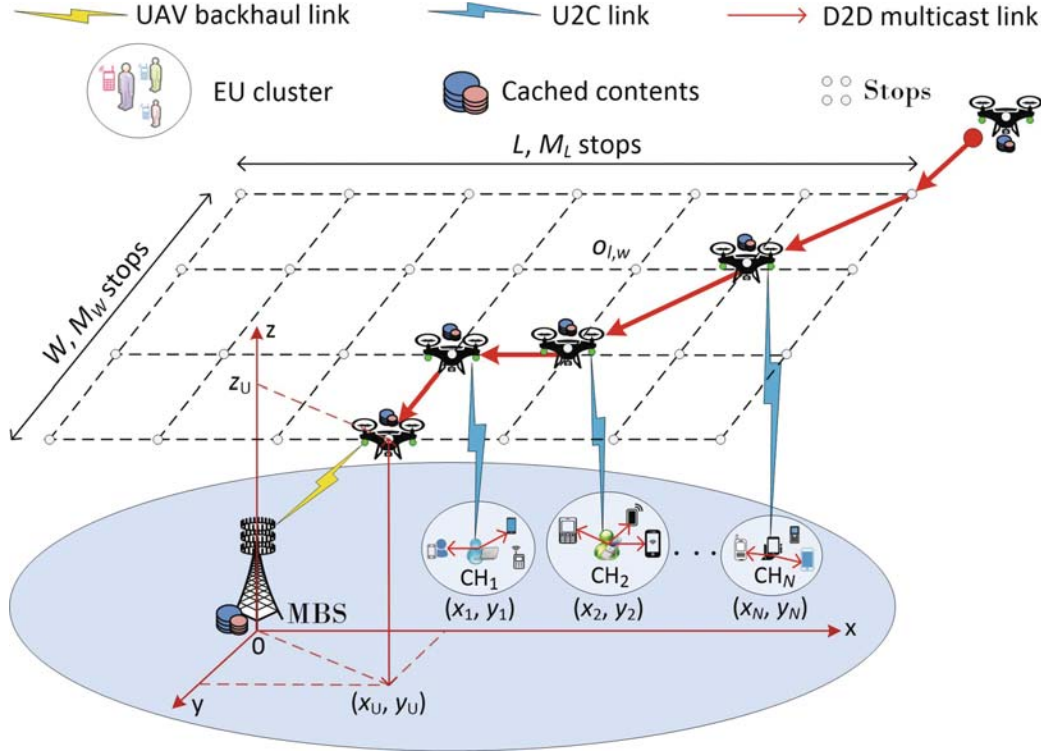


Fig. 1. UAV-assisted content delivery networks [8].

diverse requirements of AASs, especially in solving the complicated SECTO problem with multiple objective functions studied in [8].

In this paper, the typical SECTO problem is studied and then efficiently solved by applying genetic algorithms (GAs). The benefit of GAs is that they can capture the evolutionary principles of natural selection and genetic variation to find the exactly or approximately global optimal results without considering the fact that if the search space is unimodal or multimodal. To make the GAs more feasible to solve the SECTO problem with respect to two types of variables, i.e., 1) binary variables for caching decision and stop index and 2) integer variables for trajectory, we divide each original chromosome into two sub-chromosomes. The left sub-chromosome and the right sub-chromosome represent the binary variables and the integer variables, respectively. Each sub-chromosome is applied its own crossover and mutation operations with different probabilities. The performance of GAs is investigated in terms of flexible implementation, accuracy, and time complexity compared to other conventional exhausted and greedy search algorithms.

The rest of this paper is organised as follows. We introduce the UAV-assisted CDNs and describe how it works in Sect. 2. In Sect. 3, we review the SECTO problem formulations studied in [8] for the ease of following the GAs solution. Section 4 presents the GAs solution for SECTO optimisation problem. The GAs and system performance metrics are investigated in Sect. 5. Finally, Sect. 6 is dedicated to concluding the paper.

2 UAV-Assisted Content Delivery Networks

In this paper, we consider the model of UAV-assisted CDNs as shown in Fig. 1 [8]. The model consists of one MBS, one UAV, I contents, and N EU clusters. The clusters are established by applying the device-to-device (D2D) clustering scheme [13]. In each cluster, the EUs, who are interested in the same content, are represented by a cluster head (CH). The CHs, which are in charge of forwarding the contents to their EU members, are selected based on their powerful capacity of computing, storage, and energy. The MBS is able to acknowledge all the information of UEs for clustering, then formulating and solving the SECTO optimisation problem. After solving the SECTO optimisation, the UAV realises 1) the optimal set of contents for caching and 2) the optimal trajectory, i.e., optimal set of stops and optimal stop sequence, for flying and delivering. As a result, the content delivery capacity is maximised, meanwhile the flight distance is minimised. In addition, we further consider the constraints of storage and energy resources of the UAV to make the SECTO solution more efficient.

To fly, we assume that the UAV is placed at the horizontal coordinate of $o_U = (x_U, y_U)$ while the vertical coordinate is fixed at $z_U = h_U$. We then limit the flight area by defining a smallest rectangular, a length of L (m), and a width of W (m) that covers all the CHs. The rectangular is divided into a grid topology of $M_L \times M_W$ stops as shown in Fig. 1. The horizontal coordinates of the UAV's stops are $o_U = o_{l,w} = (x_l, y_w), l = 1, 2, \dots, M_L, w = 1, 2, \dots, M_W$. Furthermore, we define a $F_{M_L \times M_W}$ binary matrix in which each pair of row and column is represented by a binary stop index $f(o_{l,w})$. The UAV stops at $o_{w,l}$ if $f(o_{l,w}) = 1$, otherwise $f(o_{l,w}) = 0$. So, given $N_{\mathcal{T}}$ stops, we can find a proper stop sequence $\mathcal{T}$ for the UAV to fly throughout with minimum distance.

3 SECTO Problem Formulations

In this section, we review the system formulations studied in [8] for the ease of following. Based on these formulations, we then introduce the SECTO optimisation problem that can be solved by using GAs.

3.1 Content Popularity

It is certain that each content has its own access rate depending on the EUs' behavior, and thus a given set of I contents follows a popularity pattern. In this paper, we assume that the popularity pattern is modeled by Zipf-like distribution [14]. In this distribution, the content i is ranked in accordance with its popularity expressed as

$$p_i = \frac{i^{-\alpha}}{\sum_{i=1}^I i^{-\alpha}}, \quad (1)$$

where α is the coefficient reflecting the skewness of the content popularity preference, i.e., $\alpha = 0$ means that there is no popularity skewness among different contents, while the higher value of α makes the first contents much higher popularity than the last ones.

3.2 Caching, Requesting, and UAV-Cluster Association Index

Due to the storage limit, the UAV is capable of caching a proper number of contents which are frequently requested by the EUs in different clusters. So, the UAV has to decide which contents to be cached before flying to deliver. We define a binary variable c_i for cache decision in which $c_i = 1$ indicates that the content i is cached, otherwise $c_i = 0$. Meanwhile, the number of clusters requesting the content i , which is directly proportional to the popularity p_i is given by

$$N_i = \lceil p_i N \rceil, \quad (2)$$

where the operator $\lceil \cdot \rceil$ is used to round an arbitrary value the nearest integer.

However, this way may cause $\sum_{i=1}^I N_i \neq N$ due to the round operator. To make the equality held, without loss of generality, we do the following adjustment

- If $\sum_{i=1}^I N_i < N$, then the first smallest value in the array $\{N_i, i = 1, 2, \dots, I\}$ is increased by 1.
- If $\sum_{i=1}^I N_i > N$, then the last highest value in the array $\{N_i, i = 1, 2, \dots, I\}$ is decreased by 1.

To obtain the UAV-cluster association index, let $r_{i,n}$ be the requesting index to indicate that $r_{i,n} = 1$ if there is at least one EU in the cluster n requesting the content i ($r_{i,n} = 1$), otherwise $r_{i,n} = 0$. Obviously, we have

$$N_i = \sum_{n=1}^N r_{i,n}, \quad (3)$$

and

$$N = \sum_{n=1}^N \sum_{i=1}^I r_{i,n}. \quad (4)$$

So far, we define the UAV-cluster association index as below

$$a_{i,n} = c_i r_{i,n}. \quad (5)$$

The UAV-cluster association index $a_{i,n} \in \{0, 1\}$ is used to match the content i cached in the UAV and the cluster n . This association index enables us to establish the transmission channel, and thus derive the delivering capacity between the UAV and the cluster n presented in the following subsections.

3.3 Content Delivery Capacity

Following the UAV-cluster association, we further compute the average delivery capacity which is obtained based on the wireless channel model from the UAV to the CHs in different clusters. The UAV utilises the licensed frequency band for backhaul link, while stipulating the unlicensed Industrial, Scientific, Medical

(ISM) WiFi band, i.e., 2.4 GHz with 802.11 b/g/n devices and 5 GHz with 802.11 a/n/ac devices, for communicating with the CH in each cluster [15]. By dominating the LoS links [16, 17], the signal-to-noise ratio of the channel for delivering the content i from the UAV located at $o_{l,w}$ to the CH n in the cluster n located at o_n is given by

$$\gamma_{l,w}^{i,n} = \frac{a_{i,n} P_T \beta_0^U (d_{l,w}^n)^{-2}}{\sigma^2}, \quad (6)$$

where P_T is the transmission power of the UAV, β_0^U is the channel power gain at the reference distance of 1m, σ^2 is the additive white Gaussian noise power, and $d_{l,w}^n$, which is the distance between the UAV and the CH n , is computed as

$$d_{l,w}^n = \sqrt{h_U^2 + \|o_{l,w} - o_n\|^2}, \quad (7)$$

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

From Eq. (6), we have the corresponding delivery capacity expressed as

$$C_{l,w}^{i,n} = B \log_2(1 + \gamma_{l,w}^{i,n}), \quad (8)$$

where B is the system bandwidth.

Finally, the overall average delivery capacity from the UAV to the CHs in all clusters, which is the objective function of the SECTO optimisation problem, is given by

$$\overline{C} = \frac{1}{N} \sum_{n=1}^N \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^I p_i f(o_{l,w}) C_{l,w}^{i,n}, \quad (9)$$

where $f(o_{l,w}) \in \{0, 1\}$ is the stop index used to indicate that if $f(o_{l,w}) = 1$, the UAV stops at $o_{l,w}$, otherwise $f(o_{l,w}) = 0$, it does not stop. And thus, the number of stops is expressed as

$$N_{\mathcal{T}} = \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} f(o_{l,w}). \quad (10)$$

In Eq. (9) and Eq. (10), $\overline{C}$ is the function of caching index (c_i) and the flying strategy $f(o_{l,w})$. So, besides finding the optimal caching index, the SECTO finds the optimal flying strategy via $f(o_{l,w})$, which includes: 1) where to stop among $N_{\mathcal{T}}$ out of N stops ($N_{\mathcal{T}} \leq N$) to maximise $\overline{C}$ and 2) how to fly throughout $N_{\mathcal{T}}$ stops to minimise the distance for energy saving.

3.4 Storage and Energy Consumption

Storage Consumption. Given the caching index c_i and the size s_i of the content i , the total caching storage consumed is given by

$$S = \sum_{\substack{i=1 \\ r_{i,n}=1, \forall n}}^I c_i s_i. \quad (11)$$

Energy Consumption. The energy is consumed by transmission and flying. To compute the total energy consumption, we need to derive the transmission duration and flying duration. It is noted that when flying, the UAV must stop to transmit the contents, and thus the standstill duration is actually equal to the transmission duration. We first compute the transmission/standstill duration at the stop $o_{l,w}$ for transmitting the content i to the CH n which is given by

$$t_{l,w}^{i,n} = \frac{a_{i,n} f(o_{l,w}) s_i}{\max\{\epsilon, C_{l,w}^{i,n}\}}, \quad (12)$$

where the infinitesimal value ϵ is added to avoid the situation of dividing by zero if $C_{l,w}^{i,n} = 0$. And then, the total transmission/standstill duration is

$$t_T = \sum_{n=1}^N \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^I t_{l,w}^{i,n}. \quad (13)$$

For the flying duration, given a minimum distance $d_{\min}$ found by solving the SECTO optimisation problem (23) for minimum value of d (17) and a fixed flying velocity V , it is simply expressed by

$$t_F = d_{\min}/V. \quad (14)$$

Finally, the total energy consumption is given by

$$E = P_F t_F + (P_S + P_T) t_T, \quad (15)$$

where P_F , P_S , and P_T are the transmission powers used for flight, standstill, and transmission, respectively.

3.5 SECTO Optimisation Problem

As aforementioned, the SECTO solution includes 1) which contents to cache and where to stop (WCS) for maximising the average content delivery capacity and 2) how to fly over the stops for minimising the distance, i.e., the shortest path (STP) or travelling salesman problem. So, we first present the two STP and WCS optimisation subproblems separately and then we combine them to formulate the SECTO optimisation problem as below.

For the STP optimisation problem, given a set of $N_{\mathcal{T}}$ valid stops $\mathcal{T}_s = \{1, 2, \dots, N_{\mathcal{T}}\}$, $\mathcal{T}$ is the set of stop sequence (order) to fly, i.e., $\mathcal{T}$ is an arbitrary combination of $\mathcal{T}_s$. For example, if $\mathcal{T}_s = \{1, 2, 3\}$, $\mathcal{T} = \{2, 1, 3\}$, i.e., $\mathcal{T}_1 = 2, \mathcal{T}_2 = 1, \mathcal{T}_3 = 3$, is a valid stop sequence, meaning that the UAV flies to the stop 2 first, then to the stop 1, and finally to the stop 3. The distance to fly from the stop $n-1$ to the stop n is expressed as

$$d_n = \begin{cases} d(\mathcal{T}_{n-1}, \mathcal{T}_n), & \text{if } \mathcal{T}_n \in \mathcal{T}_s, \mathcal{T}_s = \mathcal{T}_s \setminus \mathcal{T}_n, \\ 0, & \text{if } \mathcal{T}_n \notin \mathcal{T}_s. \end{cases} \quad (16)$$

It is noted in (16) that if $\mathcal{T}_n \in \mathcal{T}_s$, after flying to the n -th stop with an actual distance added, and then this stop is removed from the set of valid stops. Otherwise, the UAV cannot fly to the n -th stop, and thus no distance is added but with violence. For example, considering an invalid stop sequence $\mathcal{T} = \{1, 1, 3\}$, it is easy to see that the violence occurs when $n = 2$ at $\mathcal{T}_2$. Consequently, the total distance for the UAV to fly over $N_{\mathcal{T}}$ stops is given by

$$d = \sum_{n=1}^{N_{\mathcal{T}}} d_n. \quad (17)$$

For the ease of grasping the violence degree of (16) if $\mathcal{T}_n \notin \mathcal{T}_s$, we further add an independent penalty function βd_p to (17); where $\beta > 0$ is the violent degree and d_p starts with 0, then increases by 1 if the violence occurs. So, the minimum value of d holds since $d_p = 0$. The STP integer combination problem is formulated as

$$\min_{\mathcal{T}} (d + \beta d_p). \quad (18)$$

For the WCS optimisation problem, we take into account the constraints of caching storage (S), energy consumption (E), and number of stops ($N_{\mathcal{T}}$), it is formulated as below

$$\max_{c_i, f(o_l, w)} \bar{C} \quad (19a)$$

$$\text{s.t.} \quad S \leq S^* \quad (19b)$$

$$E \leq E^* \quad (19c)$$

$$N_{\mathcal{T}} \leq N \quad (19d)$$

Aiming at maximising the average content delivery capacity (19) and minimising the distance (18) simultaneously, the SECTO optimisation problem is formulated as

$$\max_{c_i, f(o_l, w), \mathcal{T}} [\bar{C} - (\gamma d + \beta d_p)], \quad (20)$$

$$\text{s.t.} \quad (19b), (19c), (19d),$$

where $\gamma > 0$ is used to adjust the balance between the two objective functions $\bar{C}$ and d ; and S^* and E^* are to limit the caching storage and energy consumption.

4 GAs for SECTO Optimisation Problem

In this paper, we apply GAs [18] to solve the SECTO optimisation problem. To do so, we apply penalty function [19] which requires to rewrite (19b), (19c), and

(19d) as below

$$\begin{cases} \Delta S = S^* - S \geq 0, \\ \Delta E = E^* - E \geq 0, \\ \Delta N = N - N_{\mathcal{T}} \geq 0. \end{cases} \quad (21)$$

As we have mentioned that βd_p in (18) is the penalty function, thus we extract it from (20), and then add it to the whole penalty function of (22) which is given by

$$F = \lambda_1 (\min\{0, \Delta S\})^2 + \lambda_2 (\min\{0, \Delta E\})^2 + \lambda_3 (\min\{0, \Delta N\})^2 + \beta d_p, \quad (22)$$

where λ_1 , λ_2 , and λ_3 are the constraint violation degrees which are properly selected to punish the individuals in the current generation if they violate the constraints.

Finally, the GAs can be used to solve the unconstrained SECTO optimisation problem given by

$$\max_{c_i, f(o_{l,w}), \mathcal{T}} \bar{C}_F = \bar{C} - (\gamma d + F). \quad (23)$$

In (23), if d is minimised and too small (~ 0), the UAV flies over small number of stops or even it stops at the original location to transmit the contents. This makes the capacity significantly decreased. Inversely, if d is minimised but too large, the capacity increases but limited by E^* . The balance of maximum value of $\bar{C}$ and minimum value of d yields maximum value of $\bar{C}_F$ when F approaches 0. So, together with λ_1 , λ_2 , and λ_3 , γ is used to balance the impact of F and d on the process of maximising the fitness function $\bar{C}_F$.

To implement GAs, it is noted that the SECTO optimisation problem has two different types of variables. The binary variables (c_i and $f(o_{l,w})$) for caching and stop indexes, and meanwhile the integer variables ($\mathcal{T}$) for stop sequence. These two types have different requirements of crossover and mutation, i.e., different operations and probabilities. So, we use two chromosome types, one for binary variables and the other for integer variables. Correspondingly, we have to create two separated binary and integer populations, each of the same number of individuals is performed by its own crossover and mutation operations.

In the SECTO optimisation problem, because the binary variables are c_i and $f(o_{l,w})$ of length $I + M_L \times M_W$ bits, each individual represented by a binary chromosome (binary string) has $N_B = I + M_L \times M_W$ bits. The maximum number of stops in the sequence $\mathcal{T}$ is $N_{\mathcal{T}} = N$, each individual represented by an integer chromosome (integer string) therefore has $N_I = N$ integers. When GAs terminate, the first $N_{\mathcal{T}}$ stops in the final $\mathcal{T}$ are derived for the optimal stop sequence. The detailed implementation of GAs is shown in Algorithm 1. In Algorithm 1, GAs terminate if one of the three following conditions of TC holds: 1) $F = 0$ in 3 successive generations, 2) $\bar{C}_F$ does not change while $F \leq 10^{-3}$ in 3 successive generations, and 3) $gen = N_G$.

Algorithm 1. GAs implementation**Input:** System parameters listed in Table 1 $N_P = 10,000$: Number of individuals in both the binary and integer populations $N_B = I + M_L \times M_W$: Number of binary variables (bits) for each binary individual $N_I = N$: Number of integer variables for each integer individual $N_G = 100$: Number of generations $P_G = 0.8$: Generation gap $P_{CB} = 0.6$: Crossover probability for binary individuals $P_{CI} = 0.9$: Crossover probability for integer individuals $P_{MB} = 10^{-6}$: Mutation probability for binary individuals $P_{MI} = 10^{-10}$: Mutation probability for integer individuals $\beta = 1$: Violence degree applied if $\mathcal{T}$ is invalid $\gamma = 0.1$: Coefficient used to balance the two objective functions $\overline{C}$ and d $\{\lambda_{1,2,3}\} = \{1, 10^3, 1\}$: The method to choose the proper λ s is presented in [20] TC : Termination conditions $Gen = 1$: Generation count**Output:** $\overline{C}_F^*$ and X^*

- 1: Randomly generating N_P binary strings, each of length N_B bits to represent the individual k ($\{X_B^k\}$), $k = 1, 2, \dots, N_P$
- 2: Randomly generating N_P integer strings, each of length N_I integers to represent the individual k ($\{X_I^k\}$)
- 3: Mapping $\{X_B^k\}$ into the binary variables c_i^k and $f(o_{l,w}^k)$ and mapping $\{X_I^k\}$ into the integer variables $\mathcal{T}^k$
- 4: Calculating N_P fitness values $\overline{C}_F(c_i^k, f(o_{l,w}^k), \mathcal{T}^k)$
- 5: **while** TC does not hold **do**
- 6: Putting $\{X_B^k\}$, $\{X_I^k\}$, and $\overline{C}_F(c_i^k, f(o_{l,w}^k), \mathcal{T}^k)$ in the mating pool for ranking
- 7: Selecting $N_G = N_P \times P_G$ better individuals with higher $\overline{C}_F(c_i^k, f(o_{l,w}^k), \mathcal{T}^k)$ for breeding the next generation by using stochastic universal sampling operator [18]
- 8: Dividing the selected individuals into a set of N_G binary individuals $\{X_B^{k,*}\}$ and a set of N_G integer individuals $\{X_I^{k,*}\}$
- 9: Selecting a pair of parents to make a pair of offsprings by using 1) multiple point crossover with probability P_{CB} for $\{X_B^{k,*}\}$ and 2) single point crossover with probability P_{CI} for $\{X_I^{k,*}\}$ [18]
- 10: Mutating the offsprings of $\{X_B^{k,*}\}$ with probability P_{MB} and the offsprings of $\{X_I^{k,*}\}$ with probability P_{MI} , to recover the good genetic materials likely lost in the previous operations
- 11: Merging the two sets of offsprings into one, evaluating the fitness values of the merged set, reinserting it into the present generation
- 12: $Gen = Gen + 1$
- 13: **end while**
- 14: Finding the best fitness value $\overline{C}_F^*$ with respect to the best individual X^* in the last generation

5 Performance Evaluation

5.1 GAs Performance

To evaluate the performance of GAs, we compare it to the other two search algorithms namely exhausted - exhausted (EE) search algorithm and exhausted - greedy (EG) search algorithm which are corresponding to TSM and NSH respectively studied in [8]. We deploy GAs, EE, and EG by a desktop computer with detailed information listed in Table 2. The CHs are randomly distributed within a circle of 100-meter radius. The UAV is originally located at $(x_U, y_U) = (100\text{m}, 100\text{m})$ and its vertical coordinate is fixed at $h_U = 50\text{m}$.

We first evaluate the convergence rate of GAs by considering 100 generations. Figure 2 plots the results of the best fitness value $\bar{C}_F^*$, the mean of all the fitness values with respect to all individuals in each generation, and the penalty value (F). The results show that the convergence situation of GAs starts from the 35-th generation when the mean value is closer to the best value and $F = 0$. It indicates that all the individuals become better after each generation while ensuring the given constraints for optimal solutions.

The performance of GAs is further evaluated by investigating the accuracy and time complexity in comparison with EG and EE. The Algorithm 1 is ex-

Table 1. Parameters setting.

Symbols	Specifications
$\{N, I\}$	{10 clusters, 5 videos}
s_i	[300, 100, 200, 400, 900] Mbits
S^*	1000 Mbits
α	1
V	15 m/s
$\{P_T, P_F, P_S\}$	{0.5, 50, $0.25 \times P_F$ } W
E^*	5000 Joules
B	10 MHz
β_0^U	10^{-5}
σ^2	10^{-13} W

Table 2. Computer information.

Processor	Intel(R) Core(TM) i7-7700 CPU @ 3.60 GHz, 3601 Mhz
Processor type	x64-based PC
Processor cores	4
Logical processors	8
Total PHY memory	15,9 GB
Operating System	Windows 10

cuted versus different population sizes N_P from 1000 to 20,000. For each population size, the Algorithm 1 is repeated 50 times, and then we calculate the average of 50 optimal results of $\bar{C}$ and d in accordance with the average of time complexities. As shown in Table 3, the accuracy of GAs increases if N_P increases. We select $N_P = 10,000$ for obtaining the optimal results because if $N_P > 10,000$, the accuracy does not increase significantly but time complexity becomes too high, i.e., up to 1,009.36 s. At $N_P = 10,000$, GAs provide a reasonable accuracy of 99.69% for $\bar{C}$ and 99.51% for d compared to EE. However, GAs have a much less time complexity of 356.01 s compared to that of 1,320.17 s introduced by EE. Related to EG, although it can provide the exact result of $\bar{C}$ thanks to using exhausted search algorithm, the accuracy of d is lower than GAs done at $N_P \geq 5,000$. We can see that the accuracy and time complexity of EG is equivalent to GAs done at $N_P = 5,000$. So, GAs are flexible to obtain the exact or approximate optimal results with reasonable accuracy and time complexity.

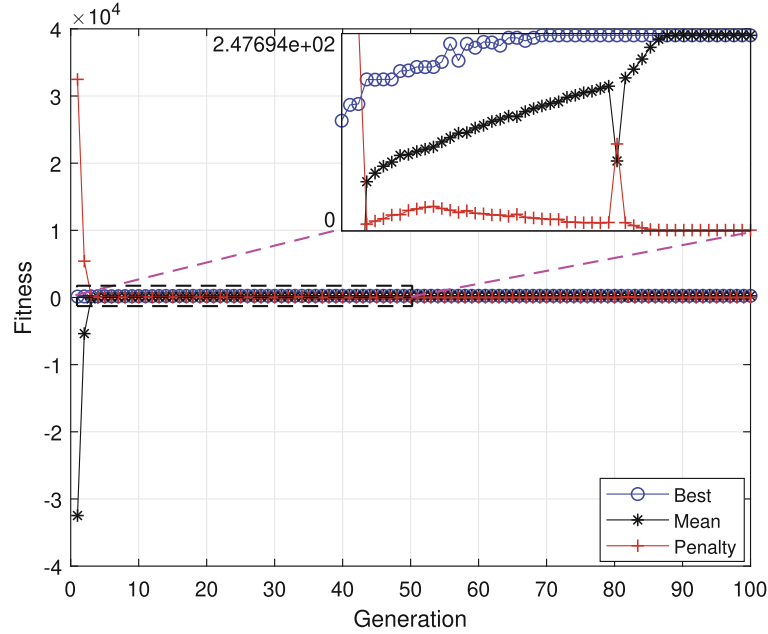


Fig. 2. Convergence rate of GAs.

Table 3. Accuracy and time complexity comparison

Metric	N_P (GAs)						
	1,000	5,000	10,000	15,000	20,000	EG	EE
$\bar{C}$ (Mbps)	310.55	310.91	310.97	311.14	311.15	311.95	311.95
Accuracy of $\bar{C}$ (%)	99.55	99.67	99.69	99.74	99.74	100.00	100.00
d (m)	696.26	653.93	645.64	641.25	641.94	657.93	642.53
Accuracy of d (%)	91.64	98.22	99.51	99.80	99.91	97.60	100.00
Time (s)	19.23	117.55	356.01	635.52	1,009.36	111.65	1,320.17

5.2 System Performance Metrics

Storage-Aware Results. To have the storage-aware results, we run the Algorithm 1 by changing the skewness coefficient of content popularity preference α with different values of caching storage constraint S^* (19b). As shown in Fig. 3, if S^* is large enough ($S^* \geq 400$ Mbits), $\bar{C}$ increases with respect to the increase of α commonly (Fig. 3(a)), meanwhile d is mostly kept the same to provide the highest value of $\bar{C}$ (Fig. 3(b)). It is noted that when α is high, the results become

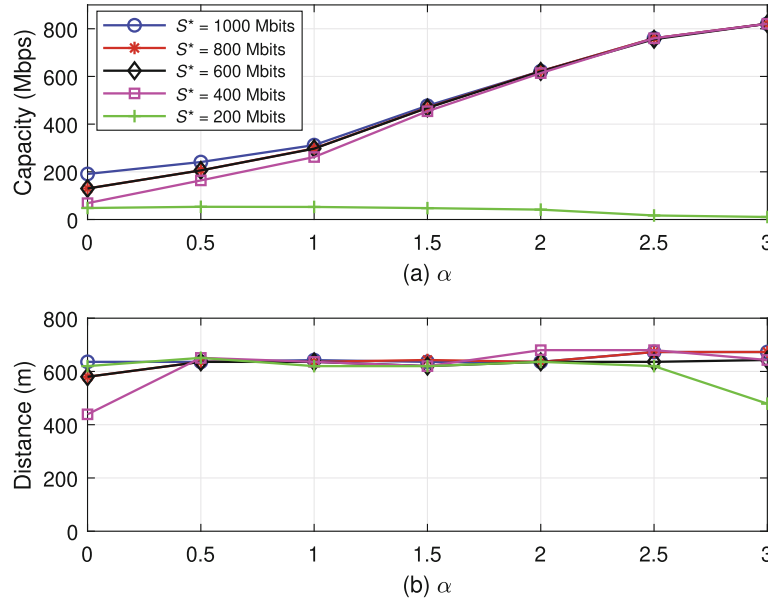


Fig. 3. Capacity and distance versus α with different S^* .

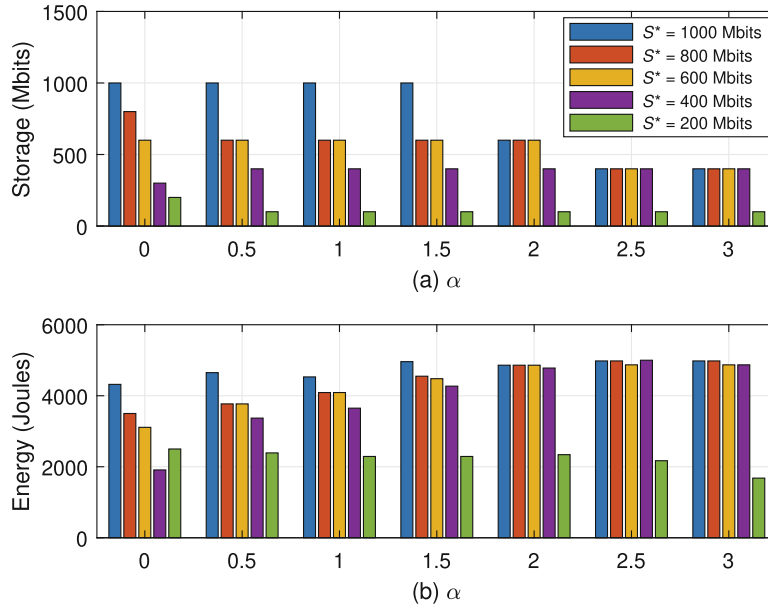


Fig. 4. Storage and energy consumption versus α with different S^* .

equally saturated because the system just focuses on less contents with higher popularity to serve the CHs for the highest capacity performance. However, if S^* is too small ($S^* = 200$ Mbits), it is clear that $\bar{C}$ significantly decreases since the UAV cannot cache the most popular contents with $s_i > 200$ Mbits, e.g., $s_1 = 300$ Mbits.

Figure 4 plots the storage and energy consumption which strictly satisfies the constraints (19b) and (19c). In Fig. 4(a), if α is high enough, i.e., $\alpha \geq 2.5$,

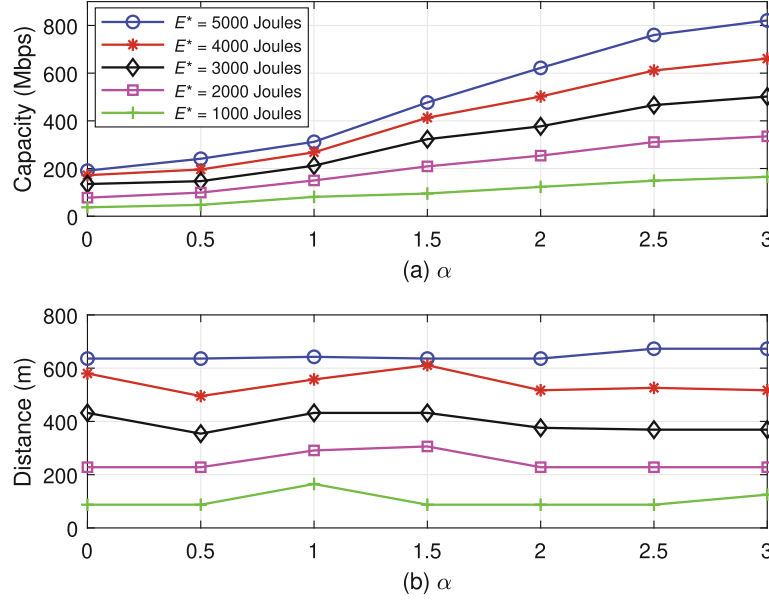


Fig. 5. Capacity and distance versus α with different E^* .

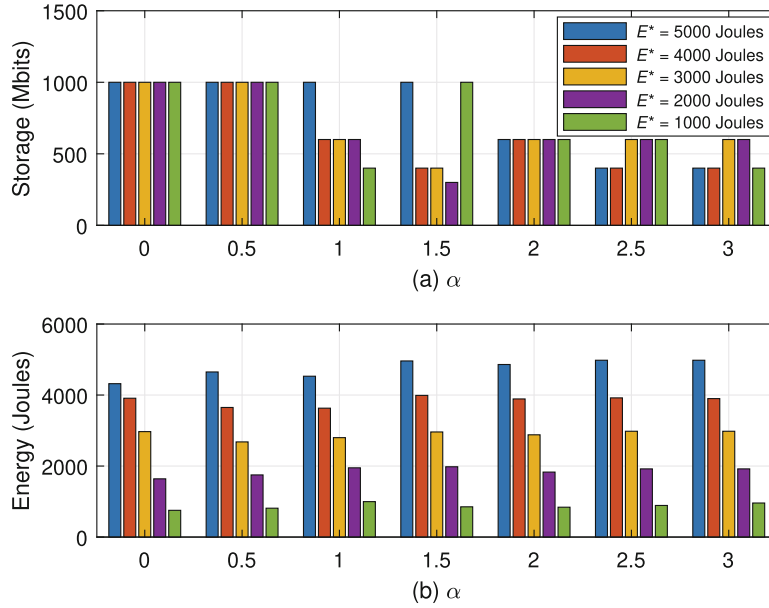


Fig. 6. Storage and energy consumption versus α with different E^* .

the system just focuses on less contents with higher popularity as we mentioned earlier, thus the caching storage consumption decreases. However, the energy consumption increases if α increases because more CHs request the most popular content with relatively large size, i.e., $s_1 = 300$ Mbits, leading to the fact that the transmission and standstill duration becomes longer requiring higher energy consumption (Fig. 4(b)), except for $S^* = 200$ Mbits making the distance decreased.

Energy-Aware Results. To have the energy-aware results, we run the Algorithm 1 by changing the skewness coefficient of content popularity preference α with different values of energy constraint E^* (19c). As shown in Fig. 5, the decrease of E^* significantly reduces both capacity and distance. The results of caching storage and energy consumption in Fig. 6 always satisfy the constraints (19b) and (19c). The behavior of all the system performance metrics is similar to that of the storage-aware results with respect to α .

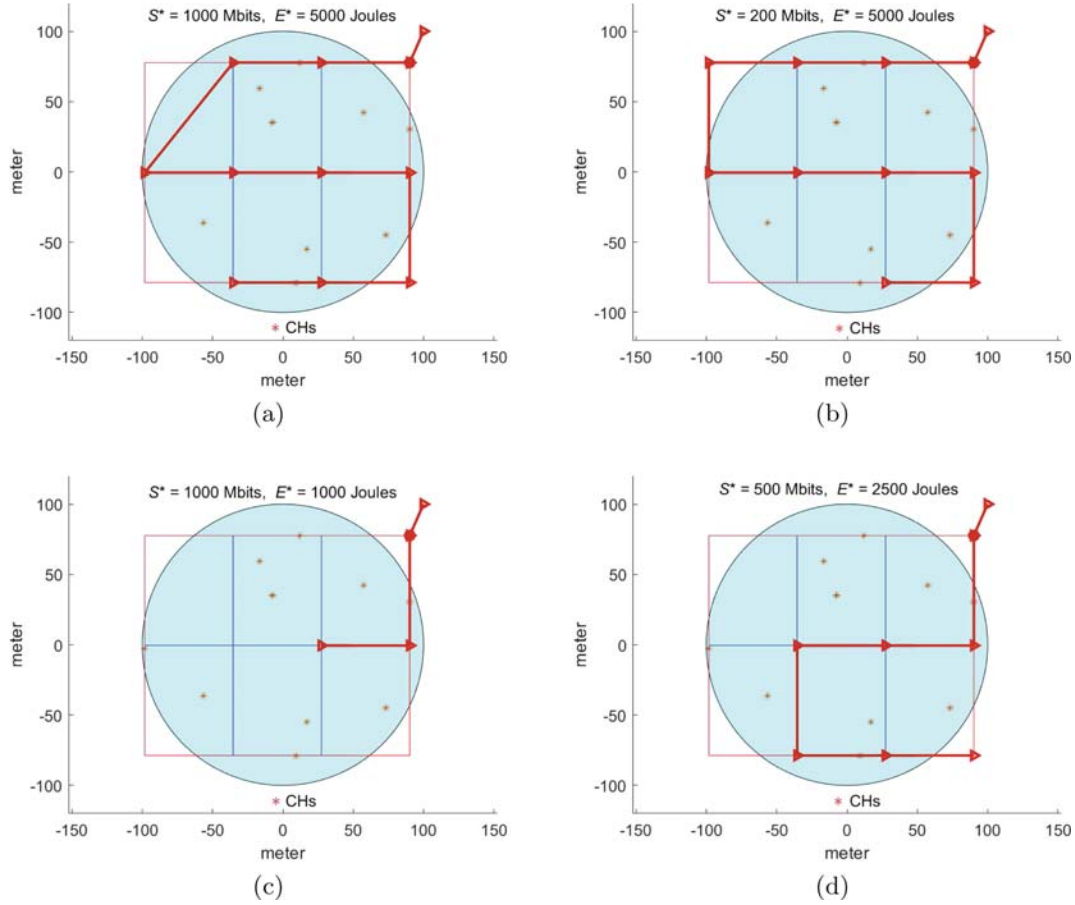


Fig. 7. Optimal trajectories with different constraints of S^* and E^* .

Optimal Trajectories. Figure 7 depicts the optimal trajectories of the UAV when $\alpha = 1$ versus different constraints of S^* and E^* . Figure 7(a) shows the optimal trajectory when S^* and E^* are relaxed to 1000 Mbits and 5000 Joules, respectively. In Fig. 7(b), if we strictly limit the caching storage constraint S^* to 200 Mbits, the UAV has to change the caching policy, and thus change the set of stops to serve the CHs with a different optimal trajectory. The trajectories significantly change to the short flight distances if we further limit the energy constraint E^* or/and the caching storage constraint S^* as plotted in Fig. 7(c) and Fig. 7(d).

6 Conclusion

In this paper, we have introduced GAs to solve the SECTO problem for UAV-assisted CDNs to maximise the content delivery capacity while minimising the flight distance. Unlike other common optimisation problem having only one objective function with respect to (w.r.t) only one type of variable, i.e., binary, real, or integer variable, the SECTO problem consists of two objective functions w.r.t both binary and integer variables. To solve the SECTO problem, we modify the GAs by applying penalty method and separating the whole string (chromosome) of each individual into binary sub-string and integer sub-string. These sub-strings are applied different crossover and mutation schemes (operations and probabilities). The results are shown to demonstrate the benefits of GAs in terms of flexible implementation, accuracy, and time complexity compared to other conventional exhausted and greedy search algorithms.

References

1. Strinati, E., et al.: 6G in the sky: on-demand intelligence at the edge of 3D networks. *ETRI J.* **42**, 643–657 (2020)
2. Chen, M., Mozaffari, M., Saad, W., Yin, C., Debbah, M., 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.* **35**, 1046–1061 (2017)
3. Zeng, Y., Wu, Q., Zhang, R.: Accessing from the sky: a tutorial on UAV communications for 5G and beyond. *Proc. IEEE* **107**, 2327–2375 (2019)
4. Lu, R., Zhang, R., Cheng, X., Yang, L.: Relay in the sky: a UAV-aided cooperative data dissemination scheduling strategy in VANETs. In: *Proceedings IEEE International Conference on Communications (Shanghai, China)*, pp. 1–6 (2019)
5. Erdelj, M., Natalizio, E., Chowdhury, K.R., Akyildiz, I.F.: Help from the sky: leveraging UAVs for disaster management. *IEEE Pervasive Comput.* **16**, 24–32 (2017)
6. Bilen, T., Canberk, B.: Content delivery from the sky: drone-aided load balancing for mobile-CDN. *EAI Endorsed Trans. Ind. Netw. Intell. Syst.* **9**, 1–7 (2022)
7. Duong, T.Q., Kim, K.J., Kaleem, Z., Bui, M.-P., Vo, N.-S.: UAV caching in 6G networks: a survey on models, techniques, and applications. *Phys. Commun.* **51**, 1–19 (2022)

8. Vo, N.-S., Lam, T.C., Bui, M.-P., Phan, T.-M., Tran, Q.-N.: UAV assisted video multicasting in 6G networks: a joint caching and trajectory optimisation. *J. Aviat. Sci. Technol.* **1**, 37–42 (2022)
9. Li, X., Liu, J., Zhao, N., Wang, X.: UAV-assisted edge caching under uncertain demand: a data-driven distributionally robust joint strategy. *IEEE Trans. Commun.* **70**, 3499–3511 (2022)
10. Zhang, T., Wang, Y., Yi, W., Liu, Y., Nallanathan, A.: Joint optimization of caching placement and trajectory for UAV-D2D networks. *IEEE Trans. Commun.* **7**, 5514–5527 (2022)
11. Xu, H., Ji, J., Zhu, K., Wang, R.: Reinforcement learning for trajectory design in cache-enabled UAV-assisted cellular networks. In: *Proceedings IEEE Wireless Communications and Networking Conference (Austin, TX)*, pp. 1–6 (2022)
12. Ji, J., Zhu, K., Cai, L.: Trajectory and communication design for cache-enabled UAVs in cellular networks: a deep reinforcement learning approach. *IEEE Trans. Mob. Comput.*, 1–15 (2022)
13. Gyawali, S., Xu, S., Ye, F., Hu, R.Q., Qian, Y.: A D2D based clustering scheme for public safety communications. In: *Proceedings of IEEE 87th Vehicular Technology Conference (Porto, Portugal)*, pp. 1–5 (2018)
14. Breslau, L., Cao, P., Fan, L., Phillips, G., Shenker, S.: Web caching and Zipf-like distributions: evidence and implications. In: *Proceedings of IEEE International Conference on Computer Communications (INFOCOM) (New York, NY)*, pp. 126–134 (1999)
15. Lin, D., Zuo, P., Peng, T., Qian, R., Wang, W.: Energy-efficient UAV-based IoT communications with WiFi suppression in 5 GHz ISM bands. *IEEE Trans. Veh. Technol.*, 1–16 (2022)
16. Xu, X., Zeng, Y., Guan, Y.L., Zhang, R.: Overcoming endurance issue: UAV-enabled communications with proactive caching. *IEEE J. Sel. Areas Commun.* **36**, 1231–1244 (2018)
17. Wu, H., Lyu, F., Zhou, C., Chen, J., Wang, L., Shen, X.: Optimal UAV caching and trajectory in aerial-assisted vehicular networks: a learning-based approach. *IEEE J. Sel. Areas Commun.*, 1–14 (2020)
18. Chipperfield, A., Fleming, P., Pohlheim, H., Fonseca, C.: Genetic algorithm TOOLBOX for using with Matlab Ver 1.2 users guide. University of Sheffield (1994)
19. Fang, T., Chau, L.P.: GOP-based channel rate allocation using genetic algorithm for scalable video streaming over error-prone networks. *IEEE Trans. Image Process.* **15**, 1323–1330 (2006)
20. Vo, N.-S., Duong, T.Q., Tuan, H.D., Kortun, A.: Optimal video streaming in dense 5G networks with D2D communications. *IEEE Access* **6**, 209–223 (2017)

Service Time-aware Caching, Power Allocation, and 3D Trajectory Optimised Multimedia Content Delivery in UAV-assisted IoT Networks

Thuong C. Lam, Nguyen-Son Vo*, Minh-Phung Bui, Chan Dai Truyen Thai, Haejoon Jung, and Van-Ca Phan

Abstract—In this paper, we consider an integration of software-defined edge techniques into unmanned aerial vehicles (UAV) to deliver multimedia contents in Internet of Things (IoT) networks. The UAV is integrated with caching, resource allocation, multicasting transmission, and 3D flight trajectory (3DF) techniques, to serve the IoT devices and users (IDU) in different clusters. By further utilising many aspects of multimedia contents, IDU features, UAV-to-clusters association, and diverse characteristics of wireless propagation environments, we formulate a service time-aware caching, power allocation, and 3DF (TCP-3DF) optimisation problem. The TCP-3DF problem is solved by genetic algorithms (GA) for optimally finding 1) the multimedia contents to cache, 2) the powers allocated to multicast each content and to fly over each hop, and 3) where to stop in 3D coordinates and stop sequence. The optimal results are found under the constraints not only on the storage and power resources but also on the service time required by different multimedia applications and services. Due to the complexity of various optimisation variables and constraints, the GA is modified to ensure the accurate and stable convergence by applying penalty function and truncated chromosome scheme. Simulation results are shown to demonstrate the feasibility of GA and the benefits of the proposed TCP-3DF method in terms of sum rate, service time, and flight distance while conserving the storage and power resources.

Index Terms—3D trajectory, edge caching, multimedia content delivery, power allocation, UAV-assisted 6G IoT networks.

I. INTRODUCTION

According to the data forecasts reported by Ericsson, it is anticipated in the sixth generation (6G) era that there will be about 39 billion Internet of Things (IoT) connections in 2029 [1]. This stimulates the development of nearly 300 million of applications and services (A&S) in the upcoming years [2].

Thuong C. Lam is with the HUTECH Institute of Engineering, HUTECH University, Ho Chi Minh City 70000, Vietnam (e-mail: lc.thuong@hutech.edu.vn).

*Corresponding author: Nguyen-Son Vo is with the Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh City 70000, Vietnam, and also with the Faculty of Electrical-Electronic Engineering, Duy Tan University, Da Nang 50000, Vietnam (e-mail: vonguyensong@duytan.edu.vn).

Minh-Phung Bui is with the Faculty of Information Technology, School of Technology, Van Lang University, Ho Chi Minh City 70000, Vietnam (e-mail: phung.bm@vlu.edu.vn).

C. D. T. Thai is with Vietnamese-German University (VGU), Vietnam (e-mail: chan.ttd@vgu.edu.vn).

H. Jung is with the Dept. of Electronics and Information Convergence Eng., Kyung Hee University, Yongin-si 17104, Korea (e-mail: haejoon-jung@khu.ac.kr).

Van-Ca Phan is with the Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology and Education, Ho Chi Minh City 70000, Vietnam (e-mail: capv@hcmute.edu.vn).

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.04-2023.25.

The ever-increasing demands of wide-area, cellular, and short-range IoT devices and users (IDU) for numerous A&Ss yield a proliferation of measurement, monitoring, automation, instruction, and entertainment communications, especially high data rate multimedia contents. Certainly, the global data traffic, which has exponentially increased [3], [4], causes not only an extreme congestion situation of all the backhaul links but also a degradation of resource utilisation in 6G networks. The Internet service providers (ISP) and content providers (CP) therefore face a major challenge of how to utilise the available system resources to provide the IDUs with high quality of service (QoS) while flexibly ensuring the service time¹ required by various multimedia A&Ss. In this context, although 6G networks are equipped with more powerful communication, sensing, computing, and intelligence capabilities, an urgent need for efficient groundbreaking solutions to address this challenge is strongly fostered.

To address the aforementioned challenge, the ISPs and CPs can develop disruptive network architectures and technologies for high-speed backhaul links, but it is a costly investment. Alternatively, a higher priority on software-defined edge (SDE) techniques can be applied to delivering the multimedia contents. One of the most efficient techniques, which can enable the ISPs and CPs to utilise the system resources and available infrastructure to satisfy the high demand of IDUs at a low cost, is terrestrial caching [5], [6]. However, in many cases, the IDUs are located in the crowded hotspots, rush hours, and important events surrounded by dense skyscrapers or in non-functional areas and emergency scenarios [2], [7], [8], thus the terrestrial caching techniques become ineffective. To resolve this limitation, the terrestrial caching techniques have rapidly cooperated with a dynamic air caching platform assisted by unmanned aerial vehicles (UAV) [2], [9]. UAV-caching techniques not only bring the multimedia contents closer to the IDUs but also 1) plan an optimal trajectory to benefit from the line-of-sight (LoS) propagation, 2) capture the behavior of IDUs (mobility, density, common interest, and social relationship), and 3) expand the functional coverage area, to provide more IDUs with a much better QoS.

In this paper, we propose an integration system of SDE techniques, namely a joint service time-aware caching, power allocation (TCP), and 3D flight trajectory (3DF) optimisation solution, for delivering multimedia contents in UAV-assisted IoT networks. For TCP, the UAV caching is integrated with

¹Service time is defined as the time for a UAV to fly and deliver (or transmit) the multimedia contents to the IDUs in different clusters.

power allocation technique. On the one hand, we apply transmission power allocation to each multimedia content associated with the specific aspects of the contents and the diverse characteristics of wireless propagation environments. For transmission, the benefit of the IDUs in a cluster having the same multimedia content interest is utilised by multicasting. On the other hand, the flight power (or flight speed) is allocated to each hop depending on the service time required by various multimedia A&Ss. For 3DF, the UAV-to-clusters (U2C) association is established so that the UAV can find the 3D coordinates of stops and the stop sequence based on the IDU features to serve them more efficiently.

The UAV caching, power allocation, multicasting transmission, and U2C association-based 3DF are then synthesised into a TCP-3DF optimisation problem, under the constraints on storage and power resources, and significantly handled by the requirement of service time. To make the TCP-3DF practically for a higher optimisation performance, we encompass the aspects of multimedia contents (size and popularity pattern), the IDU features (density and common interest), and the diverse characteristics of wireless propagation environments (suburban, urban, dense urban, and high-rise urban). By using genetic algorithms (GA), the TCP-3DF optimisation problem is solved for the optimal results of caching, transmission and flight powers, 3D coordinates of stops², and stop sequence. The objective is to provide more IDUs with maximum sum rate of multimedia content delivery (MCD) while conserving the storage and power resources, and satisfying the required service time with the approximate shortest path. Furthermore, due to the complexity of the TCP-3DF optimisation problem, the conventional GA is modified by applying penalty function and truncated chromosome (string) scheme, so as to ensure that the optimal results are accurate and stable. The main contributions of our paper are summarized as follows:

- We propose the TCP-3DF method for MCD in UAV-assisted IoT networks. The UAV flies over the approximate shortest path to satisfy more IDUs by maximum sum rate and proper service time. Simultaneously, it benefits the ISPs and the CPs by effectively utilising the storage and power resources and available infrastructure.
- The TCP-3DF method applies various SDE techniques including UAV caching, transmission and flight power allocations, multicasting transmission, and 3DF. The practical aspects of multimedia contents, IDU features, U2C association, characteristics of wireless propagation environments, storage and power resources, and required service time are considered for formulating a multi-objective constrained TCP-3DF optimisation problem. The TCP-3DF method achieves the highest optimisation performance in terms of sum rate, service time, storage and power resource utilisation, and flight distance. It is noted that by allocating the flight and transmission powers, we can control both the flight time and transmission time of the UAV to meet the required service time.

²3D coordinates of stops are defined as a horizontal 2D rectangular flight grid (RFG) and the altitudes of stops. A point to stop (or not to stop) in the 2D RFG is identified by a binary stop index 1 (or 0) with unique x and y coordinates.

- We modify the GA so that it works well with the complicated optimisation variables (OV) and constraints of the TCP-3DF optimisation problem. In particular, the TCP-3DF optimisation problem has three types of six OVs: 1) binary type for caching and stop indexes² in the horizontal 2D RFG, 2) real type for transmission and flight powers and altitudes, and 3) integer type for stop sequence. Because each type (or OV) may require different crossover and mutation schemes, we truncate the original string of an individual into sub-strings, each represents a type of OVs or an OV to make the GA accurately and stably converged to the best generation in the end. To address the complicated constraints, i.e., not simply in the form of lower bounds and upper bounds, we apply penalty function to convert the constrained optimisation problem to an unconstrained one which can be feasibly solved by GA.
- Simulation results demonstrate the feasibility of GA with truncated string (GTS) and the benefits of TCP-3DF method compared to the other conventional ones. The detailed analyses and discussions about the simulation results provide valuable insights into the design of SDE techniques based on UAV for MCD in IoT networks.

The rest of this paper is organised as follows. Related works are discussed in Section II. In Section III, we introduce the UAV-assisted MCD system in IoT networks with TCP-3DF and describe how it works. The system, which includes preliminary models, objective functions, service time, and resource consumption, is formulated in Section IV. Based on the system formulation, Section V proposes the multi-objective constrained TCP-3DF optimisation problem and solution with GTS. Section VI is dedicated to presenting the performance evaluation of TCP-3DF method. Finally, we conclude the paper in Section VII.

II. RELATED WORKS

It is certain that more benefits of UAV caching for content delivery in IoT networks can be achieved by integrating with other emerging techniques such as resource allocation [10]–[15], energy harvesting (wireless power transmission) [13], [15], [16], scheduling and jamming [14], [17], [18], multicasting [19], [20], non-orthogonal multiple access (NOMA) [12], [13], [15], relaying and cooperative communications [21]–[23], and sensing [24]. We can see that the majority of the integrated emerging techniques is based on resource allocation [10], [11] combined with energy harvesting [13], [15], jamming [14], and NOMA [12], [13], [15]. It is prerequisite to elaborate on the pros and cons of these main related works [10]–[15], [17], [22], to demonstrate the contributions of our work in the sequel.

One of the most important solutions for UAV caching with emerging techniques is the joint cache placement, flight trajectory and transmission power optimisation studied in [10]. The authors have deployed multiple UAVs to serve the IDUs by optimally caching the popular contents in advance, identifying the 2D coordinates with a fixed altitude, and finding the transmission power. The objective is to maximise the minimum

throughput among the UAV-served IDUs. Unlike [10], the authors in [11] have considered the mobile network operator and the CPs which can mutually allocate the computing resources for transcoding each video into different bitrate versions. The popular videos with the highest bitrate versions (highest qualities) are cached in the UAV. Then, the UAV flies to serve the subscribers of CPs in the hotspots, each at an optimal hover altitude of 3DF, i.e., optimal coverage radius of UAV, to maximise the number of videos offloaded.

Besides applying resource allocation, in [12], the authors have deployed NOMA in a flying ad hoc network (FANET) of multiple UAVs. The FANET cooperates with the macro base station (MBS) and cellular users (CU) in functional areas to cache, and then forward the instruction messages and contents to the IDUs in non-functional areas. The system further applies the distributed NOMA technique to enhance the sum rate and the access rate of the IDUs in UAV-assisted heterogeneous IoT networks. Deploying only one UAV with caching but assisted by power allocation, NOMA, and energy harvesting, the work [13] introduces a joint constrained optimisation problem for maximising the throughput. The impact of the UAV's position on the system performance is investigated to fully comprehend the optimisation design.

In most specific A&Ss like safety public management, military, and e-health [2], the contents transmitted to the IDUs need to be secured. In this regard, the authors in [14] have proposed a joint design of caching, power allocation, jamming, and 2D trajectory. There are two UAVs in which the first one is for data caching and transmission while the other is for jamming. The CUs are utilised as relay nodes to provide the IDUs with two-hop communications, i.e., from UAV to CUs and from CUs to IDUs over device-to-device communications. This way can maximise the secrecy rate in the presence of eavesdroppers. Using similar techniques with [13], the authors in [15] have further applied user pairing scheme to improve the performance of NOMA while reducing the delay.

By a simple but efficient design, the work in [17] has considered UAV caching with 2D trajectory for delivering the contents in intelligent transportation systems. Although this way can maximise the offloading gain, the problem is that without optimising the altitude of UAV, it cannot adapt to the diverse characteristics of wireless propagation environments and density of IDUs to obtain higher performance. For a better adaptation to the density of IDUs (but not the wireless propagation environments), the authors in [22] further considered optimising the altitude of UAV for 3DF to maximise the throughput. However, the caching, transmission, and flight of UAV cannot significantly enhance the system performance if the size of contents for caching, the power for transmitting each content, and the power for flying over each hop are fixed.

We can see from the aforementioned discussions on the related works that the authors have focused on various aspects of UAV caching such as system models, techniques, IDUs, contents, trajectories, resources, and related performance metrics. However, the problem of these works is that the size of contents is fixed and the UAVs are deployed in 2D trajectories (or even without trajectory in case of FANET deployment), except that the work in [13] considers multiple versions of

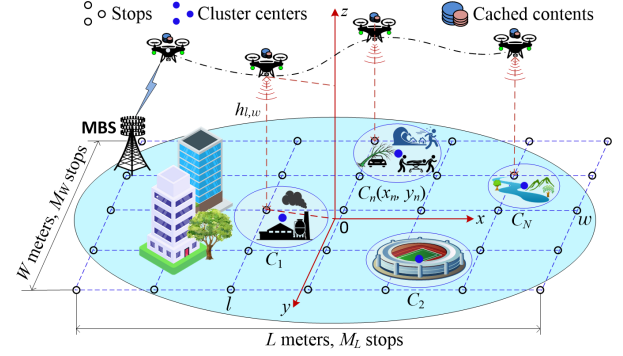


Fig. 1: A UAV-assisted MCD system in IoT networks with TCP-3DF.

videos with different sizes and 3DF. Furthermore, the authors have not applied 1) U2C association, 2) transmission power allocation for each content and flight power allocation for each hop, 3) multicasting technique, 4) diverse characteristics of both wireless propagation environments and density of IDUs, and 5) approximate shortest path, in order to maximise the delivery sum rate while not only conserving the storage and power resources but also ensuring the service time.

III. SYSTEM ARCHITECTURE

We propose a UAV-assisted MCD system in IoT networks with TCP-3DF as shown in Fig. 1. The system includes one MBS, one UAV, N clusters, and I contents, within a circular area of radius R . Cluster n , $n = 1, 2, \dots, N$, has a number of IDUs distributed in a density of ρ_n . Content i , $i = 1, 2, \dots, I$, has a size of s_i (Mbits) with popularity p_i . The channels between the MBS and the IDUs are not reliable enough for data transmission due to the obstacles in the middle, but they can be sufficiently utilised for collecting the necessary information from the IDUs to formulate and solve the TCP-3DF optimisation problem. The MBS controls the UAV to cache the requested contents, fly, and deliver them to the IDUs over much stronger U2C communications. The main notations used for the system formulation are listed in Table I.

For the ease of trajectory, we deploy an RFG of $M_W \times M_L$ stops with horizontal coordinates (x_l, y_w) at altitudes $h_{l,w}$, defined as $o_{l,w} = (x_l, y_w, h_{l,w})$, $l = 1, 2, \dots, M_L$ and $w = 1, 2, \dots, M_W$ [18]. The UAV flies to stop at $o_{l,w}$ and deliver the cached contents. Finding a set of 3D coordinates $o_{l,w}$ is to identify where to stop $f_{l,w}$ in the RFG at which altitude $h_{l,w}$. Here $f_{l,w} \in \{0, 1\}$ is the stop index to affirm that the UAV stops at (x_l, y_w) if $f_{l,w} = 1$, otherwise $f_{l,w} = 0$. When the MBS anticipates that there is a huge amount of contents requested by the IDUs in the rush hours, crowded hotspots, important events, non-functional areas, and emergency scenarios [2], [7], [8], it performs the three phases given as below.

- **Preparation phase:** The MBS collects the information of the IDUs and groups them into N clusters [20]. Cluster n has M_n IDUs distributed in a density of ρ_n within a circular area of radius R_n . The IDUs in a cluster, which are in close physical and social relationship, have the same content interest for sharing, instructing, executing, completing tasks, and even entertaining together.

TABLE I: Notations.

Symbols	Specifications
R	Radius of target circular area covered by UAV
N	Number of clusters
I	Number of contents
B	System bandwidth
σ^2	Additive white Gaussian noise power
N_i	Number of clusters requesting content i , $i = 1, 2, \dots, I$
ρ_n	Density of IDUs in cluster n , $n = 1, 2, \dots, N$
M_n	Number of IDUs in cluster n of radius R_n
M_L	Length of M_L stops in the RFG
M_W	Width of M_W stops in the RFG
$o_{l,w}$	Stop coordinate of UAV at (x_l, y_w) in the RFG, $l = 1, 2, \dots, M_L$, $w = 1, 2, \dots, M_W$, and at altitude $h_{l,w}$
$M_{l,w,n}$	Number of IDUs in cluster n covered by UAV at $o_{l,w}$
$u_n^{(k)}$	Coordinate $(x_n^{(k)}, y_n^{(k)})$ of IDU k in cluster n , $k = 1, 2, \dots, M_n$
N_T	Number of stops in a stop sequence T , i.e., $N_T \leq N$
s_i	Size of content i
p_i	Popularity of content i
α	Skewed popularity exponent coefficient among contents
$r_{i,n}$	Requesting index, $r_{i,n} \in \{0, 1\}$, to identify that the IDUs in cluster n request content i or not
c_i	UAV caching index, $c_i \in \{0, 1\}$, to cache content i or not
$f_{l,w}$	UAV stop index, $f_{l,w} \in \{0, 1\}$, to stop at (x_l, y_w) or not
$v_{n,T}$	Speed of UAV over hop n_T , $n_T = 1, 2, \dots, N_T$
$P_{n,T}$	Power of UAV used to fly over hop n_T
P_i	Power of UAV used to transmit content i
P_S	Standstill power of UAV
I_0	No-load current (A)
R_0	Motor resistance (Ω)
U_0	No-load voltage (V)
K_v	Nominal no-load motor constant (rpm/V)
K_E	Back-electromotive force constant, $K_E = (U_0 - I_0 R_0)/(K_v U_0)$
K_T	Torque constant, $K_T = 9.55 K_E$
m	UAV mass (kg)
g	Acceleration of gravity (m/s^2)
C_m	Torque coefficient ($\text{Nm/(rad/s}^2)$)
C_d	Fuselage drag coefficient ($\text{N/(m/s}^2)$)
C_t	Thrust coefficient ($\text{N/(rad/s}^2)$)

Then, the MBS formulates and solves the TCP-3DF optimisation problem for optimal results of caching c_i ; transmission power P_i for content i and flight power $P_{n,T}$ for hop n_T , $n_T = 1, 2, \dots, N_T$; and where to stop $f_{l,w}$ induced number of stops N_T , at which altitude $h_{l,w}$, and stop sequence T for 3DF.

- **Implementation phase:** Based on the optimal results, the UAV caches the proper contents, flies to the right stops at the optimal coordinates, and transmit the contents with optimal transmission powers. It also flies over the hops with optimal flight powers and optimal stop sequence.
- **Standby phase:** At the last stop, after transmitting the contents completely, the UAV keeps standstill for a short time to receive the next task from the MBS if any. Finally, it flies to the nearby charging station and puts on standby.

It is noted that in our system, to enhance the performance of a service time-aware regime, we consider the cruising speed with respect to the flight power consumption computed based on the voltage dropping across and the current flowing through the UAV's electric motors [25]. It means that we can control the speed of UAV over different hops to meet the service time requirement by changing the flight power. More practically, we utilise the diverse characteristics of radio propagations for different U2C communications. This way stimulates the UAV to fly in an optimal 3DF and serve the IDUs more efficiently.

IV. SYSTEM FORMULATION

A. Preliminary Models

1) *Multimedia Content Popularity Pattern:* To derive the content requesting and caching models, we take the content popularity into account. It is certain that each content has a popularity (or access rate) resulting in different delivery frequency which highly impacts the system performance. In this paper, we apply the well-known Zipf-like distribution [26] to represent the popularity pattern of I contents in which the popularity of content i is given by

$$p_i = \frac{i^{-\alpha}}{\sum_{i=1}^I i^{-\alpha}}, \quad (1)$$

where $\alpha \geq 0$ is the skewed coefficient representing a particular popularity pattern of I contents. Higher value of α yields higher skewed popularity amongst the contents.

2) *Requesting, Caching, and U2C Association Indexes:* The number of clusters, which has the IDUs requesting content i , is proportional to p_i [27], expressed as

$$N_i = \lfloor p_i N \rfloor, \quad (2)$$

where the operator $\lfloor \cdot \rfloor$ is used to round a real value to the nearest integer. Furthermore, to avoid resulting in $\sum_{i=1}^I N_i \neq N$ caused by the round operator, we make an adjustment in the array $\{N_i\}$ as follows:

- If $\sum_{i=1}^I N_i < N$, the first smallest number in the array is increased by 1, e.g., if $\{N_i\} = \{3, 3, 2, 1, 1\}$ and $N = 11$, then $\{N_i\} = \{3, 3, 2, 2, 1\}$.
- If $\sum_{i=1}^I N_i > N$, the last largest number in the array is decreased by 1, e.g., if $\{N_i\} = \{3, 3, 2, 1, 1\}$ and $N = 9$, then $\{N_i\} = \{3, 2, 2, 1, 1\}$.

Let $r_{i,n} \in \{0, 1\}$ be the requesting index to indicate that there is at least one of the IDUs in cluster n requesting content i ($r_{i,n} = 1$) or there is none ($r_{i,n} = 0$), the relationship between N and $r_{i,n}$ is given by

$$N = \sum_{i=1}^I N_i = \sum_{i=1}^I \sum_{n=1}^N r_{i,n}. \quad (3)$$

We further define $c_i \in \{0, 1\}$ as the caching index to indicate that the UAV decides to cache content i ($c_i = 1$) or does not ($c_i = 0$). So, the U2C association index ($a_{i,n} \in \{0, 1\}$), which is used to match the caching index of UAV and the requesting index of IDUs in clusters, is expressed as

$$a_{i,n} = c_i r_{i,n}. \quad (4)$$

Based on (4), the transmission channel between content i cached in the UAV and the IDUs in cluster n is properly established for optimal content caching and delivery.

3) *UAV Coverage:* In cluster n of radius R_n at center $C_n = (x_n, y_n)$, the number of IDUs with the same content interest is $M_n = \lfloor \rho_n \pi R_n^2 \rfloor$. As shown in Fig. 2, the UAV stops at (x_l, y_w) of distance $\sqrt{(x_n - x_l)^2 + (y_n - y_w)^2}$ far from C_n . The number of IDUs covered by the UAV is indeed given by

$$M_{l,w,n} = \lfloor \rho_n A_{l,w,n} \rfloor, \quad (5)$$

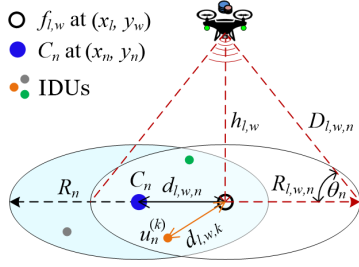


Fig. 2: The coverage model of UAV at a stop.

where $A_{l,w,n}$ is the intersection area of the two circles with radiuses R_n and $R_{l,w,n}$. In fact, IDU k , for $k = 1, 2, \dots, M_n$, is in the intersection area if the distance conditions $d_{l,w,n} \leq R_n + R_{l,w,n}$ and $d_{l,w,k} \leq R_{l,w,n}$ are held, here $d_{l,w,k}$ is the distance between $f_{l,w}$ and IDU k located at $u_n^{(k)} = (x_n^{(k)}, y_n^{(k)})$. So, (5) is computed by checking the distance conditions of all M_n IDUs in cluster n . To guarantee the QoS of the U2C links from the UAV to the IDUs in $A_{l,w,n}$, we consider the region covered by the UAV which is defined as a circular area of radius $R_{l,w,n}$ associated with altitude $h_{l,w}$ and elevation angle θ_n [28], expressed as

$$R_{l,w,n} = \frac{h_{l,w}}{\tan(\theta_n)}. \quad (6)$$

4) *Air-to-ground Channel*: In this system, the A2G channel from the UAV located at $o_{l,w}$ to the IDUs for delivering the contents is considered. We apply the widely used A2G channel defined by probabilistic path loss model [29]–[32]. This model consists of LoS and non-LoS (NLoS) links. The path loss of LoS and NLoS links between the UAV and IDU k in cluster n is given by

$$\psi_{l,w,n}^{\mathcal{A},(k)} = \left(\frac{4\pi f_c d(o_{l,w}, u_n^{(k)})}{c} \right)^2 \eta_n^{\mathcal{A}}, \mathcal{A} \in \{\text{LoS}, \text{NLoS}\}, \quad (7)$$

where f_c is the carrier frequency, c is the speed of light, $\eta_n^{\mathcal{A}}$ is the attenuation factor of LoS link or NLoS link, and the distance from the UAV to IDU k in cluster n is given by

$$d(o_{l,w}, u_n^{(k)}) = \sqrt{(x_n^{(k)} - x_l)^2 + (y_n^{(k)} - y_w)^2 + (h_{l,w})^2}. \quad (8)$$

By further considering the occurrence probabilities of the LoS link ($P_{l,w,n}^{\text{LoS},(k)}$) and of the NLoS link ($P_{l,w,n}^{\text{NLoS},(k)} = 1 - P_{l,w,n}^{\text{LoS},(k)}$), we can derive the following average path loss

$$\begin{aligned} \bar{\psi}_{l,w,n}^{(k)} &= \psi_{l,w,n}^{\text{LoS},(k)} P_{l,w,n}^{\text{LoS},(k)} + \psi_{l,w,n}^{\text{NLoS},(k)} P_{l,w,n}^{\text{NLoS},(k)} \\ &= L_{l,w,n}^{(k)} [\eta_n^{\text{LoS}} P_{l,w,n}^{\text{LoS},(k)} + \eta_n^{\text{NLoS}} P_{l,w,n}^{\text{NLoS},(k)}], \end{aligned} \quad (9)$$

where $L_{l,w,n}^{(k)} = \left(\frac{4\pi f_c d(o_{l,w}, u_n^{(k)})}{c} \right)^2$ and the occurrence probability of the LoS link is expressed as

$$P_{l,w,n}^{\text{LoS},(k)} = \frac{1}{1 + \kappa_n \exp[-\nu_n (\arctan(\zeta_{l,w,n}^{(k)}) - \kappa_n)]}, \quad (10)$$

where $\zeta_{l,w,n}^{(k)} = \frac{h_{l,w}}{\sqrt{(x_n^{(k)} - x_l)^2 + (y_n^{(k)} - y_w)^2}}$, κ_n and ν_n are the constants related to the environment of cluster n [33].

B. Objective Functions

1) *Average Sum Rate*: We assume that the UAV, which is considered as a common cellular user, is allocated itself a downlink spectrum resource for transmission. Without interference, the signal-to-noise ratio of the channel for delivering content i from the UAV at $o_{l,w}$ to IDU k in cluster n at $u_n^{(k)}$ is given by [29]

$$\gamma_{l,w,n}^{(i),(k)} = \frac{a_{i,n} (\bar{\psi}_{l,w,n}^{(k)})^{-1} P_i}{\sigma^2}, \quad (11)$$

where P_i is the transmission power of the UAV consumed to deliver content i . From (11), we have the corresponding content delivery rate expressed as

$$C_{l,w,n}^{(i),(k)} = B \log_2(1 + \gamma_{l,w,n}^{(i),(k)}), \quad (12)$$

where B is the system bandwidth.

Finally, the average delivery sum rate multicasted from the UAV to all the IDUs per cluster, which is the first objective function of the TCP-3DF optimisation problem, is given by

$$\bar{C} = \frac{1}{N} \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{n=1}^N \sum_{k=1}^{M_{l,w,n}} \sum_{i=1}^I p_i f_{l,w} C_{l,w,n}^{(i),(k)}, \quad (13)$$

where $f_{l,w} \in \{0, 1\}$ is the stop index used to indicate that the UAV stops at (x_l, y_w) if $f_{l,w} = 1$, otherwise $f_{l,w} = 0$. So, the number of stops is expressed as

$$N_{\mathcal{T}} = \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} f_{l,w}. \quad (14)$$

Aiming at reducing the complexity of the objective function (13) but without loss of generality, we consider the worst case of sum rate received at all IDUs in $A_{l,w,n}$ over the longest transmission distance of $d(o_{l,w}, u_n^{(k)}) = D_{l,w,n} = \sqrt{(R_{l,w,n})^2 + (h_{l,w})^2}$, which is shown in Fig. 2. The content delivery rate in the worst case is given by

$$C_{l,w,n}^{(i)} = C_{l,w,n}^{(i),(k)}|_{d(o_{l,w}, u_n^{(k)})=D_{l,w,n}}. \quad (15)$$

Therefore, (13) is reduced to

$$\bar{C} = \frac{1}{N} \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{n=1}^N M_{l,w,n} \sum_{i=1}^I p_i f_{l,w} C_{l,w,n}^{(i)}. \quad (16)$$

2) *Flight Distance*: The flight distance of the UAV, which is the second objective function of the TCP-3DF optimisation problem, is simply computed as

$$d_{\mathcal{T}} = \sum_{n_{\mathcal{T}}=1}^{N_{\mathcal{T}}} d_{n_{\mathcal{T}}}, \quad (17)$$

where $d_{n_{\mathcal{T}}}$ is the distance from stop $(n_{\mathcal{T}} - 1)$ to stop $n_{\mathcal{T}}$ in $\mathcal{T}$, namely $d(\mathcal{T}_{n_{\mathcal{T}}-1}, \mathcal{T}_{n_{\mathcal{T}}})$, here $\mathcal{T}_0$ represents the original position of the UAV.

C. Service Time and Resource Consumption

1) *Average Service Time*: The service time is defined as the total waiting time for each cluster to be completely served by the UAV. It consists of flight time and transmission time. First, the flight time over hop $n_{\mathcal{T}}$, is given by

$$t_{n_{\mathcal{T}}} = \frac{d_{n_{\mathcal{T}}}}{v_{n_{\mathcal{T}}}}, \quad (18)$$

where $v_{n_{\mathcal{T}}}$ is the speed of the UAV over hop $n_{\mathcal{T}}$ in $\mathcal{T}$.

Second, the transmission (or standstill) time at $o_{l,w}$ to transmit content i to the IDUs in cluster n is expressed as

$$t_{l,w,n}^{(i)} = \begin{cases} \frac{f_{l,w} s_i}{C_{l,w,n}^{(i)}}, & \text{if } a_{i,n} = 1, \\ 0, & \text{if } a_{i,n} = 0. \end{cases} \quad (19)$$

So, the total transmission time (T_S) and the average service time per cluster ($\bar{T}$) are respectively given by

$$T_S = \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{n=1}^N \sum_{i=1}^I t_{l,w,n}^{(i)} \quad (20)$$

and

$$\bar{T} = \frac{1}{N} \left[\sum_{n_{\mathcal{T}}=1}^{N_{\mathcal{T}}} (N_{\mathcal{T}} - n_{\mathcal{T}} + 1) t_{n_{\mathcal{T}}} + \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{n=1}^{N-1} \sum_{i=1}^I (N - n) t_{l,w,n}^{(i)} \right]. \quad (21)$$

2) *Storage Consumption*: To ensure the caching efficiency, the caching storage consumption, which depends on not only the content size (s_i) but also the caching index (c_i) and the requesting index ($r_{i,n}$) represented by the U2C association (4), is computed as

$$S = \sum_{n=1}^N \sum_{i=1}^I a_{i,n} s_i. \quad (22)$$

3) *Power Consumption*: The power consumption of the UAV is used for transmission, standstill, and flight. Because the standstill power is mostly the same, namely P_S , without optimisation at all the stops, we consider the transmission power P_i and the flight power $P_{n_{\mathcal{T}}}$ allocated to content i and hop $n_{\mathcal{T}}$, respectively. Given I contents and $N_{\mathcal{T}}$ stops, the total transmission power (P_T) and the total flight power (P_F) are given by

$$P_T = \sum_{i=1}^I P_i, \quad (23)$$

and

$$P_F = \sum_{n_{\mathcal{T}}=1}^{N_{\mathcal{T}}} P_{n_{\mathcal{T}}}, \quad (24)$$

where $P_{n_{\mathcal{T}}}$ is the flight power of the quad-rotor UAV at speed $v_{n_{\mathcal{T}}}$ computed as [25]

$$P_{n_{\mathcal{T}}} = \frac{c_4 V_{n_{\mathcal{T}}}}{4} + \frac{c_3 V_{n_{\mathcal{T}}}^{\frac{3}{4}}}{2} + c_2 V_{n_{\mathcal{T}}}^{\frac{1}{2}} + 2c_1 V_{n_{\mathcal{T}}}^{\frac{1}{4}} + 4c_0, \quad (25)$$

where $c_0 = I_0^2 R_0$, $c_1 = 30 I_0 K_E / \pi$, $c_2 = 2 I_0 R_0 C_m / K_T$, $c_3 = 30 K_E C_m / (\pi K_T)$, $c_4 = R_0 C_m^2 / K_T^2$, and

$$V_{n_{\mathcal{T}}} = (m^2 g^2 + C_d^2 v_{n_{\mathcal{T}}}^4) / C_t^2, \quad (26)$$

and the notations in (25) and (26) are defined in Table I.

V. TCP-3DF DESIGN

A. TCP-3DF Problem

The goals of the TCP-3DF method are to optimally find 1) a set of contents to be cached in the UAV (c_i); 2) the transmission power for content i (P_i) and the flight power over hop $n_{\mathcal{T}}$ ($P_{n_{\mathcal{T}}}$); 3) the 3DF including where to stop ($f_{l,w}$) at which altitude ($h_{l,w}$) and stop sequence $\mathcal{T}$. The objective is to maximise the average delivery sum rate $\bar{C}$, fly over $N_{\mathcal{T}}$ stops with the shortest path (STP), while satisfying the constraints on average service time (T_C), caching storage (S_C), and total transmission power consumption (P_{TC}) and flight power consumption (P_{FC}). Generally, the TCP-3DF method has two mutual optimisation problems: 1) caching, transmission and flight power allocations, and where to stop at which altitude (CPS) optimisation problem for maximum $\bar{C}$ and 2) stop sequence for the STP optimisation problem, detailed below.

1) *CPS Optimisation Problem*: The CPS optimisation problem is formulated as

$$\max_{c_i, f_{l,w}, P_i, P_{n_{\mathcal{T}}}, h_{l,w}} \bar{C} \quad (27a)$$

$$\text{s.t. } N_{\mathcal{T}} \leq N \quad (27b)$$

$$\bar{T} \leq T_C \quad (27c)$$

$$S \leq S_C \quad (27d)$$

$$P_T \leq P_{TC} \quad (27e)$$

$$P_F \leq P_{FC} \quad (27f)$$

$$P_{TL} \leq P_i \leq P_{TU} \quad (27g)$$

$$P_{FL} \leq P_{n_{\mathcal{T}}} \leq P_{FU} \quad (27h)$$

$$H_{FL} \leq h_{l,w} \leq H_{FU} \quad (27i)$$

where (27b) is to ensure that the number of stops cannot be greater than the number of clusters; P_{TL} and P_{TU} (27g), P_{FL} and P_{FU} (27h), and H_{FL} and H_{FU} (27i) are the lower bounds and upper bounds respectively of transmission power, flight power, and altitude of the UAV.

2) *STP Optimisation Problem*: Given $f_{l,w}$ and $N_{\mathcal{T}}$ found by solving (27), the STP optimisation problem, which is formulated for optimal stop sequence $\mathcal{T}$ to minimise the flight distance, is expressed as

$$\min_{\mathcal{T}} d_{\mathcal{T}}. \quad (28)$$

Solving (28) is equivalent to finding the best $\mathcal{T}$ in a searching space of $N_{\mathcal{T}}!$ permutations of $\mathcal{T}_s = \{1, 2, \dots, N_{\mathcal{T}}\}$ that minimises the flight distance $d_{\mathcal{T}}$. For example, if $N_{\mathcal{T}} = 3$ and $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\} = \{2, 1, 3\}$ is the best solution selected from 6 permutations of $\mathcal{T}_s = \{1, 2, 3\}$, the UAV flies to the stops 2, 1, and 3 in sequence.

B. TCP-3DF Solution with GA

GA has a powerful capability for solving such a complicated optimisation problem in various systems [34], e.g., from communications [35]–[37] to multimedia A&Ss [38]–[41]. One of the most important features of GA is that based on the evolutionary principles of nature selection and genetic variation, it flexibly provides exact or approximate

global optimal solutions despite the unimodal or multimodal searching spaces. In other words, GA is able to avoid local optimal solutions by considering many optimal peaks at the same time [38], [39]. In this paper, we apply GA [42] to solve the TCP-3DF optimisation problem. However, for the CPS optimisation problem (27), GA is designed for simple constraints like (27g), (27h), and (27i). To address the other complicated constraints, i.e., (27b), (27c), (27d), (27e), and (27f), we further apply penalty function [38] which requires to rewrite these complicated constraints as below

$$\begin{cases} \Delta_N = N - N_T \geq 0, \\ \Delta_T = T_C - \bar{T} \geq 0, \\ \Delta_S = S_C - S \geq 0, \\ \Delta_{P_T} = P_{TC} - P_T \geq 0, \\ \Delta_{P_F} = P_{FC} - P_F \geq 0. \end{cases} \quad (29)$$

And then, the penalty function is given by

$$F = \lambda_1(\min\{0, \Delta_N\})^2 + \lambda_2(\min\{0, \Delta_T\})^2 + \lambda_3(\min\{0, \Delta_S\})^2 + \lambda_4(\min\{0, \Delta_{P_T}\})^2 + \lambda_5(\min\{0, \Delta_{P_F}\})^2, \quad (30)$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, and λ_5 are the constraint violation degrees which are properly selected to punish the individuals in the current generation if they violate the constraints.

As a result, the GA can be used to solve the unconstrained CPS optimisation problem given by

$$\max_{c_i, f_{l,w}, P_i, P_{n_T}, h_{l,w}} \bar{C}_F = \bar{C} - F. \quad (31)$$

For the STP optimisation problem, to solve it by GA, we also apply penalty function to punish an individual by a distance of $d_{\max}$ if it makes the UAV fly to an invalid stop $\mathcal{T}_{n_T}$ when finding $\mathcal{T}$. The invalid stops occur because the GA generates a number of random individuals, i.e., random stop sequences, to create the initial population. In a random stop sequence, e.g., $\mathcal{T} = \{2, 2, 3\}$, the condition for $\mathcal{T}_{n_T} \neq \mathcal{T}_{m_T}, m_T \neq n_T$, cannot be held to be valid. The flight distance with punishment from $\mathcal{T}_{n_T-1}$ to $\mathcal{T}_{n_T}$ is given by

$$d_{n_T} = \begin{cases} d(\mathcal{T}_{n_T-1}, \mathcal{T}_{n_T}), & \text{if } \mathcal{T}_{n_T} \in \mathcal{T}_s, \mathcal{T}_s = \mathcal{T}_s \setminus \mathcal{T}_{n_T} \\ d_{\max}, & \text{if } \mathcal{T}_{n_T} \notin \mathcal{T}_s, \end{cases} \quad (32)$$

where $\mathcal{T}_s = \{1, 2, \dots, N_T\}$ is the set of valid stops and $\mathcal{T}_s = \mathcal{T}_s \setminus \mathcal{T}_{n_T}$ is applied to ensure that after flying to a valid stop, this stop is removed from $\mathcal{T}_s$. To acknowledge the process of punishment, we add a penalty amount βd_p to (28) so that the STP is rewritten as

$$\min_{\mathcal{T}} (d_{\mathcal{T}} + \beta d_p), \quad (33)$$

where $\beta > 0$ is the violent degree, and d_p , which is initially equal to 0, is then increased by 1 whenever an invalid stop occurs. This way enables us to observe that $d_{\mathcal{T}}$ is minimised if $d_p = 0$.

So far, the TCP-3DF optimisation problem is solved by combining CPS (31) and STP (33) optimisation problems into one which is given below

$$\max_{c_i, f_{l,w}, P_i, P_{n_T}, h_{l,w}, \mathcal{T}} \bar{C}_F = \bar{C} - \gamma(F + d_{\mathcal{T}} + \beta d_p). \quad (34)$$

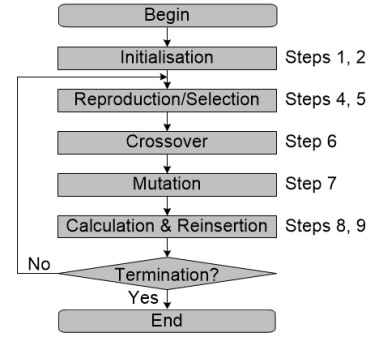


Fig. 3: Flowchart of GA.

In the fitness function (34), γ is used to balance between $\bar{C}$ and $(F + d_{\mathcal{T}} + \beta d_p)$ to maximise $\bar{C}_F$ while minimising $d_{\mathcal{T}}$. However, solving it by GA deals with the challenge of an extremely long chromosome (string) represented for each individual since there are various OV(s) ($c_i, f_{l,w}, P_i, P_{n_T}, h_{l,w}$, and $\mathcal{T}$). This in turn requires a huge population size to ensure the GA converged accurately and stably. Furthermore, each OV has its own characteristics associated with specific crossover and mutation schemes, i.e., operators and probabilities. So, instead of using long strings for common crossover and mutation schemes, we apply a divide-and-conquer policy to truncate them into sub-strings, each represents one or two OV(s), namely GA with truncated string (GTS). The general flowchart of GA is described in Fig. 3 corresponding to the detailed steps of GTS in Algorithm 1. The GTS terminates when it meets one of the following conditions (TC):

- 1) The error of $\bar{C}_F$ between two generations ($Gen - 1$ and Gen) and the total penalty ($F + \beta d_p$) are less than 10^{-3} in 10 consecutive generations.
- 2) $Gen = N_G$, i.e., 100 generations.

Similar to other adaptive heuristic searching algorithms, the complexity of Algorithm 1 depends directly on the searching space (N_P) and the number of iterations (N_G). Finding the proper values of N_P and N_G is further based on 1) the complexity of the TCP-3DF optimisation problem, i.e., the complexity of the fitness function (34) defined as the multi-objective functions (27) and (33), the penalty function (30), and the OV(s) needed to be encoded, and 2) the accuracy required by the multimedia A&Ss. Obviously, the complexity of Algorithm 1 is also associated with the selection, crossover, and mutation operators. In general, the computational complexity of Algorithm 1 is of the order $\mathcal{O}(N_P N_G)$ [35], [39].

VI. PERFORMANCE EVALUATION

The system's parameters are listed in Table II. In addition, the content size (s_i), the cluster radius (R_n), density of IDUs (ρ_n) are randomly generated in the ranges of [50, 250] Mbits, [10, 20] meters, and [0.01, 0.1] IDUs/m², respectively. The UAV originally starts at (100, 100, 75)m. To make the 3DF more effectively, a set of propagation features ($\kappa_n, \nu_n, \eta_n^{\text{LoS}}, \eta_n^{\text{NLoS}}, \theta_n$) in cluster n is also randomly selected from one of the four types of environments (suburban, urban, dense urban, high-rise urban) given in Table III. We evaluate the performance of the TCP-3DF method by comparing it to the other schemes including 1) service time-aware flying strategy

Algorithm 1 GTS for TCP-3DF.

Input: Parameters of system in Tables II and III

$Gen = 1$: Generation count

Output: $\bar{C}_F^*(c_i^*, f_{l,w}^*, P_i^*, P_{n_T}^*, h_{l,w}^*, T^*)$

1: Randomly generate

- 1) N_P sub-strings $\{X_1^z\}$, $z = 1, 2, \dots, N_P$, each of $(I + M_W \times M_L)$ bits to represent a potential solution for both c_i and $f_{l,w}$ in which the first I bits are in an array of $X_{1,I}^z$ and the last $M_W \times M_L$ bits are in an $M_W \times M_L$ matrix of $X_{1,M_W \times M_L}^z$.
- 2) N_P sub-strings $\{X_2^z\}$, each of $I \times PRECI$ bits to represent a potential solution for I real OV of P_i .
- 3) N_P sub-strings $\{X_3^z\}$, each of $N \times PRECI$ bits to represent a potential solution for N real OV of P_{n_T} .
- 4) N_P sub-strings $\{X_4^z\}$, each of $N \times PRECI$ bits to represent a potential solution for N real OV of $h_{l,w}$.
- 5) N_P sub-strings $\{X_5^z\}$, each of N base- N numerals to represent a potential solution for T .

where X_2^z , X_3^z , and X_4^z are respectively converted to real OV of $X_{2,R}^z$, $X_{3,R}^z$, and $X_{4,R}^z$ by using b2r operation [42]; in 3), 4), and 5), because we only need N_T valid stops, the $(N - N_T)$ rear ones are ignored; and the base- N operation generating the numerals from 0 to $N - 1$, is added by 1 to have $T_{n_T} \in T_s$.

- 2: Compute N_P fitness values based on (34) to have $\bar{C}_F^z$, i.e., $\bar{C}_F^z(X_{1,I}^z, X_{1,M_W \times M_L}^z, X_{2,R}^z, X_{3,R}^z, X_{4,R}^z, X_5^z)$
- 3: **while** TC does not hold **do**
- 4: Put $\{X_{ALL}^z\} = [\{X_1^z\}\{X_2^z\}\{X_3^z\}\{X_4^z\}\{X_5^z\}]$ associated with $\bar{C}_F^z$ into the mating pool for ranking
- 5: Select $N_{PG} = N_P \times P_G$ best individuals with higher fitness values for breeding by using stochastic universal sampling operator [42], so as to obtain $\{X_{ALL}^t\} = [\{X_1^t\}\{X_2^t\}\{X_3^t\}\{X_4^t\}\{X_5^t\}]$, $t = 1, 2, \dots, N_{PG}$
- 6: Choose a pair of parents to create the offsprings by using multiple point crossover with probabilities P_{C1} , P_{C4} , and P_{C5} for $\{X_1^t\}$, $\{X_4^t\}$, and $\{X_5^t\}$, while single point crossover with probabilities P_{C2} and P_{C3} for $\{X_2^t\}$ and $\{X_3^t\}$
- 7: Mutate the offsprings $\{X_1^t\}$, $\{X_2^t\}$, $\{X_3^t\}$, $\{X_4^t\}$, and $\{X_5^t\}$ with probabilities P_{M1} , P_{M2} , P_{M3} , P_{M4} , and P_{M5} by using complement operation. This way, the positive genetic features that have been probably lost in the previous steps can be recovered. As a result, we obtain $\{X_{ALL}^{t,*}\} = [\{X_1^{t,*}\}\{X_2^{t,*}\}\{X_3^{t,*}\}\{X_4^{t,*}\}\{X_5^{t,*}\}]$
- 8: Repeat step 2 to obtain $\bar{C}_F^{t,*}(X_{1,I}^{t,*}, X_{1,M_W \times M_L}^{t,*}, X_{2,R}^{t,*}, X_{3,R}^{t,*}, X_{4,R}^{t,*}, X_5^{t,*})$
- 9: Reinsert $\{X_{ALL}^{t,*}\}$ and $\bar{C}_F^{t,*}$ into the present generation to have the new sets of $\{X_{ALL}^z\}$ and $\bar{C}_F^z$
- 10: $Gen = Gen + 1$
- 11: **end while**
- 12: Find the best fitness $\bar{C}_F^*(c_i^*, f_{l,w}^*, P_i^*, P_{n_T}^*, h_{l,w}^*, T^*) \in \{\bar{C}_F^z(X_{1,I}^z, X_{1,M_W \times M_L}^z, X_{2,R}^z, X_{3,R}^z, X_{4,R}^z, X_5^z)\}$

(TFS) with joint optimisation of caching, where to stop, and altitude (TFS-CSA) [22], 2) TFS with joint optimisation of caching and where to stop (TFS-C&S) [17], 3) TFS with only optimal caching (TFS-OOC), and 4) TFS without any optimisations (TFS-NUL), as shown in Table IV.

A. Evaluation of GTS

We first evaluate the performance of GTS with a set of parameters presented in Table II. Fig. 4 shows the convergence rate of GTS over 100 generations. We can see that the GTS starts to converge after 70 generations and meets the TC from the 90-th generation. In the convergence situation, it is certain that the mean fitness value (Mean) and the best fitness value (Best) of all individuals are nearly the same. In other words,

TABLE II: System's parameters.

Symbols	Specifications
N, I	10 clusters, 10 contents
$M_W \times M_L$	3×4 , distributed in a circular area with $R = 100$ meters
B, f_c	10MHz, 2GHz
σ^2, c, α	$10^{-13}W$, $3 \times 10^8m/s$, 1
I_0, R_0, U_0	0.3A, 0.4Ω , 10V
K_v, m, g	380rpm/V, 3Kg, $9.8m/s^2$
C_m, C_d, C_t	$8.89 \times 10^{-7}Nm/(rad/s)^2$, $0.11N/(m/s)^2$, $4.85 \times 10^{-5}N/(rad/s)^2$
P_S	250W
P_{TL}, P_{TU}	100mW, 5000mW
P_{FL}, P_{FU}	300W, 2000W
H_{FL}, H_{FU}	50 meters, 150 meters
T_C, S_C	50s, 1Gbits
N_P	5000 individuals, i.e., population size
N_G	100 generations
P_G	0.9, generation gap
$PRECI$	8 bits representing the precision of each real OV
$P_{C1,2,3,4,5}$	0.6, 0.1, 0.1, 0.6, 0.9 for crossover probabilities
$P_{M1,2,3,4,5}$	10^{-9} , 10^{-12} , 10^{-12} , 10^{-9} , 10^{-9} for mutation probabilities
d_{max}, γ, β	200, 1.25, 10
$\lambda_{1,2,3,4,5}^a$	1, 1, 0.001, 0.2, 0.01

^aThe method to select the proper values of $\lambda_{1,2,3,4,5}$ is presented in [39].

TABLE III: Propagation features [28], [33].

Parameters	κ_n	ν_n	η_n^{LoS}	η_n^{NLoS}	θ_n
Environments					
Suburban	4.88	0.43	0.1	21	20.34°
Urban	9.61	0.16	1	20	42.44°
Dense urban	12.08	0.11	1.6	23	54.62°
High-rise urban	27.23	0.08	2.3	34	75.52°

TABLE IV: OV of different methods.

Methods	c_i	$f_{l,w}$	$h_{l,w}$	P_i and P_{n_T}
TCP-3DF	Yes	Yes	Yes	Yes
TFS-CSA [22]	Yes	Yes	Yes	No ¹
TFS-C&S [17]	Yes	Yes	No ²	No ¹
TFS-OOC	Yes	No ³	No ²	No ¹
TFS-NUL	No ⁴	No ³	No ²	No ¹

Note: STP is applied to all methods. TFS-CSA, TFS-C&S, and TFS-OOC optimisation problems are solved by using GA, but the results of TFS-NUL are computed on average.

¹ $P_i = 1000mW$, $P_{n_T} = 1000W$, $\forall i, n_T$, the transmission and flight powers of TFS-CSA are $P_{T_{CSA}}$ and $P_{F_{CSA}}$, computed using (23) and (24), respectively. Then, in (27e) and (27f), we set $P_{TC} = P_{T_{CSA}}$ and $P_{FC} = P_{F_{CSA}}$.

² $h_{l,w} = 125m$, $\forall l, w$.

³We derive all the permutations of selecting N_T out of N stops. Then, all the performance metrics, i.e., sum rate, distance, service time, storage, transmission and flight power consumption, and energy consumption, are computed with respect to each permutation. Finally, we compute the mean of each metric per permutation that satisfies the constraints.

⁴Given the number of contents optimally cached by TCP-3DF method, namely $V^* = \sum_{i=1}^I c_i$, we compute all the permutations of caching V^* out of I videos. Then, combining with the stop permutations, we compute the mean of each metric per pair of stop and caching permutations.

most of the individuals in the last generations own the best genetic features following the evolutionary principle. Fig. 4 also demonstrates that the penalty value $(F + \beta d_p)$ approaches zero, and thus all the constraints are satisfied.

Related to the accuracy and stability of GTS, the GTS is stable if it produces as many accurate global solutions as possible given a number of tries. The accuracy and stability can be improved by increasing the population size but not

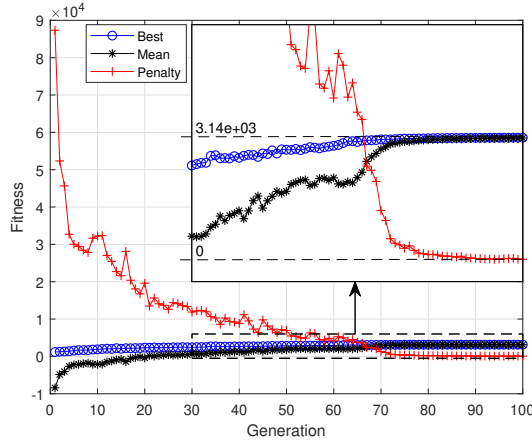


Fig. 4: Convergence rate of GTS.

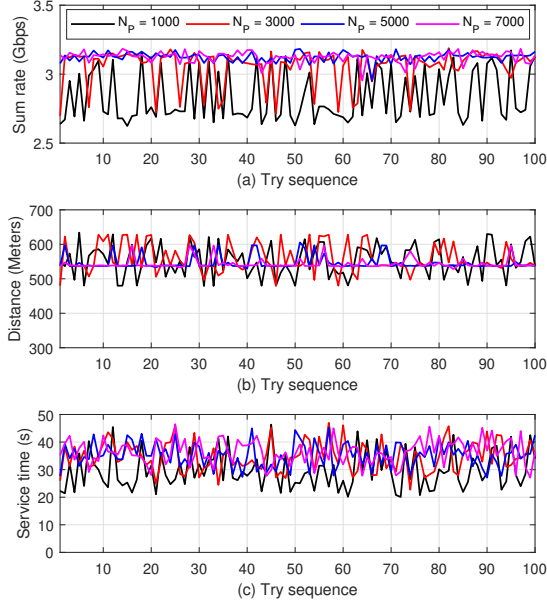


Fig. 5: Stability of GTS vs. different population sizes.

very large. The reason is that the large population size causes a higher computational complexity while not significantly improving the accuracy and stability anymore. Meanwhile, it is obvious that a small population size with low computational complexity cannot ensure the accuracy and stability. To ensure that the GTS is converged accurately and stably within a reasonable computational complexity, we carefully select the population size. Particularly, we execute the GTS 100 tries for each population size (N_P) in $\{1000, 2000, 3000, 4000, 5000, 6000, 7000\}$. We then plot the results of sum rate, distance, and service time of every try as shown in Fig. 5 and plot the minimum (Min), average (Ave), and maximum (Max) values of 100 tries as shown in Fig. 6. It is observed clearly in Fig. 5 and Fig. 6 that lower N_P (1000 and 3000) causes the results less accurate and less stable. Meanwhile, higher N_P (7000) does not significantly improve the accuracy and stability but certainly yielding a higher computational complexity. Especially, Fig. 6 indicates that the proper population size is 5000 (at the elbow) when the Ave closely reaches to the Max, i.e., numerous accurate global solutions and high stability of GTS are achieved within 100 tries.

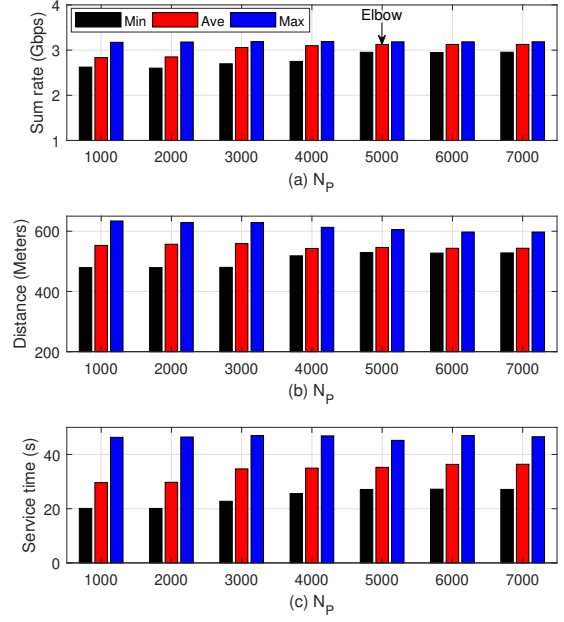


Fig. 6: Accuracy of GTS vs. different population sizes.

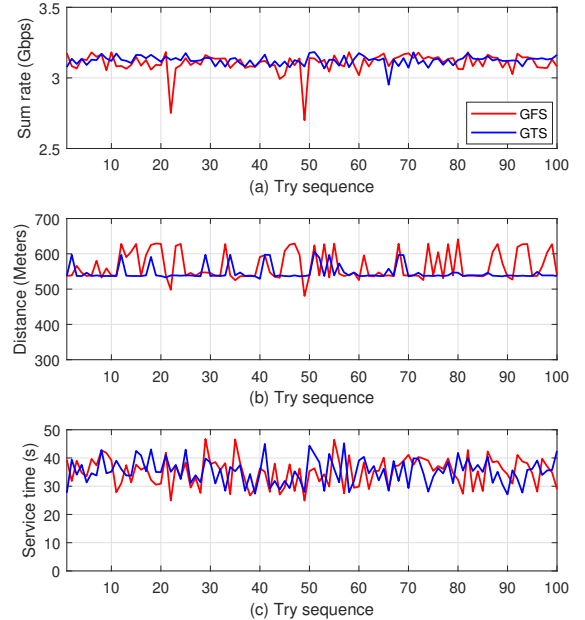


Fig. 7: Stability of GTS and GFS over 100 tries.

In the fulfillment of GTS evaluation, we further implement the GA with full long strings (GFS) and compare it to GTS in terms of accuracy and stability. Unlike GTS with five sub-strings, the GFS has two sub-strings. The first sub-string of GFS is a concatenation of the first four sub-strings of GTS $\{X_1^z\{X_2^z\{X_3^z\{X_4^z\}$ and the second sub-string of GFS is the last sub-string of GTS $\{X_5^z\}$. The parameters of GFS are the same with that of GTS except that the crossover and mutation probabilities are set to 0.9 ($\max\{P_{C_1}, P_{C_2}, P_{C_3}, P_{C_4}, P_{C_5}\}$) and 10^{-12} ($\min\{P_{M_1}, P_{M_2}, P_{M_3}, P_{M_4}, P_{M_5}\}$)³, respectively. In this way,

³Longer strings require higher crossover probability and lower mutation probability.

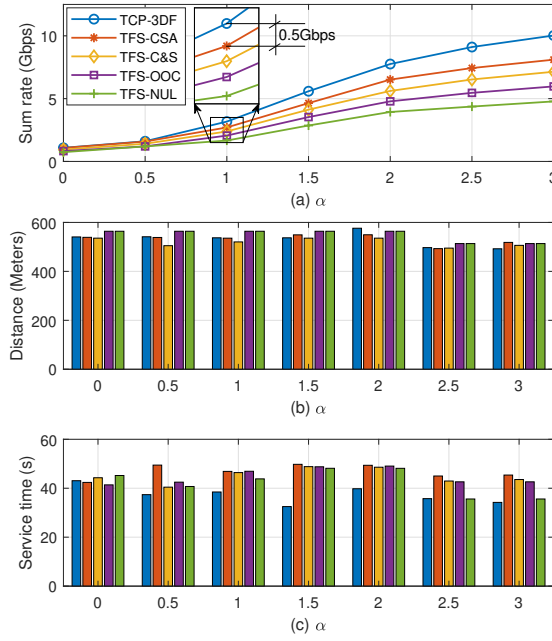


Fig. 8: Sum rate, distance, and service time vs. α .

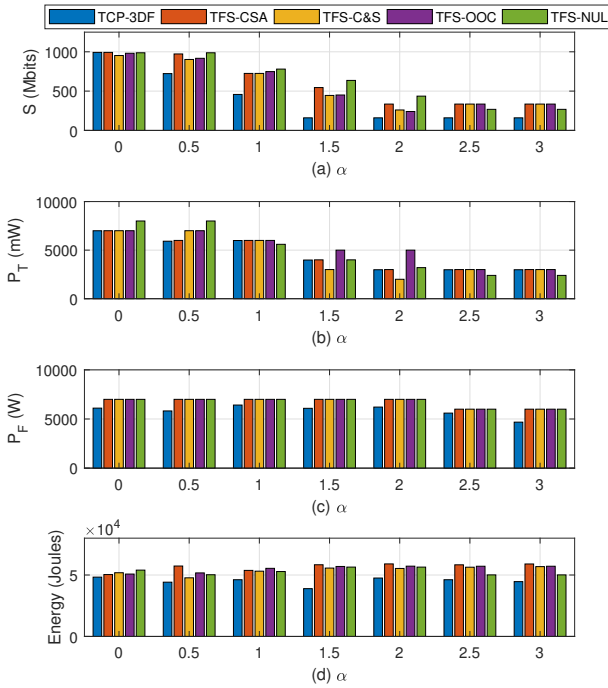


Fig. 9: Storage, power, and energy consumptions vs. α .

the memory and time complexities of GTS and GFS are equivalent. The results in Fig. 7 show that the GTS is more stable, and thus more likely to be accurate than the GFS is. In other words, it is not good to merge various types (or OV) of individuals and then apply the same crossover and mutation schemes to them. Finally, we can conclude that the GTS is feasible to solve the complicated problem of TCP-3DF in terms of accuracy, stability, and complexity.

B. Evaluation of System Performance Metrics

Next, we evaluate the system performance metrics of TCP-3DF, TFS-CSA, TFS-C&S, TFS-OOC, and TFS-NUL which include sum rate, distance, service time, storage, transmission and flight powers, and energy consumption. Fig. 8 and Fig. 9 plot the performance metrics of all methods with respect to α . In Fig. 8(a), it is worth observing that α plays an important role in evaluating the system performance. As α is low or equal to zero, all the contents have similar or the same popularity. This makes the caching technique ineffectively leading to the result that other optimisations of stop index, transmission and flight powers, and altitude do not work well. When α increases, the popularity values are more significantly skewed among the contents. The contents are selectively cached in the UAV and thus all the optimisation solutions can jointly work to make the proposed TCP-3DF more efficiently compared to others. Particularly, at $\alpha = 1$, from the TFS-NUL, if we gradually add up the OV) one by one to obtain different methods as shown in Table IV, the sum rate is correspondingly increased by the amounts of 0.4Gbps, 0.3Gbps, 0.3Gbps, and 0.5Gbps. The performance gain between TCP-3DF and TFS-NUL is totally up to 1.5Gbps. Importantly, the results in Fig. 8(b) and Fig. 8(c) demonstrate that the flight distance of TCP-3DF is approximately minimum compared to, while the service time of TCP-3DF is mostly lower than that of the others. The approximately minimum flight distance of TCP-3DF is obtained due to the nature of GTS which provides exact or approximate global optimal solutions given a reasonable population size ($N_P = 5000$).

Fig. 9 further shows the resource utilisation metrics of TCP-3DF, TFS-CSA, TFS-C&S, TFS-OOC, and TFS-NUL. Considering the storage resource in Fig. 9(a), the TCP-3DF method takes full advantage of the skewed popularity of contents to gain the highest storage efficiency. The higher the skewed popularity is, the more storage efficiency the TCP-3DF obtains, i.e., less storage is used for caching the most popular contents. In Fig. 9(b), Fig. 9(c), and Fig. 9(d), the TCP-3DF consumes a reasonable transmission power and the lowest flight power, and thus the lowest total energy compared to other methods, thanks to the optimal power allocation solution. It is noted that the total energy in Fig. 9(d) is determined by $E = \sum_{n=1}^{N_T} t_{n,T} P_{n,T} + T_S P_S + \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{n=1}^N \sum_{i=1}^I t_{l,w,n}^i P_i$, where $P_S = 250W$ is the standstill power of UAV [25] and T_S is the total standstill time (transmission time) given in (20).

In Table V, we evaluate the impact of service time constraint (T_C) on the system performance metrics of TCP-3DF, done at $\alpha = 1$. To do so, we relax the transmission and flight power constraints by setting $P_{TC} = 10,000mW$ and $P_{FC} = 10,000W$. It is certain that if the service time is required faster, most of the system performance metrics reduce accordingly to serve less. However, the transmission power consumption is not changed significantly. The reason the transmission power consumption keeps unchanged is that in any constraints of service time, the transmission power resource must be fully utilised to reduce the transmission time as much as possible. So, we can flexibly adjust T_C to meet the service time required by different multimedia A&Ss.

TABLE V: Performance of TCP-3DF vs. T_C ($\alpha = 1$, $P_{TC} = 10,000\text{mW}$, $P_{FC} = 10,000\text{W}$).

$T_C(\text{s})$	Sum rate (Mbps)	Distance (m)	Service time (s)	P_T (mW)	P_F (W)	Storage (Mbits)	Energy (Joules)
75	4251.6	709.9	72.9	9973	9353.3	707.5	66,321
50	3779.0	617.1	45.6	9988	8853.3	457.5	52,523
25	3010.5	518.3	24.9	9989	8773.3	457.5	45,912
10	2050.7	339.6	9.9	9989	8773.3	457.5	30,533

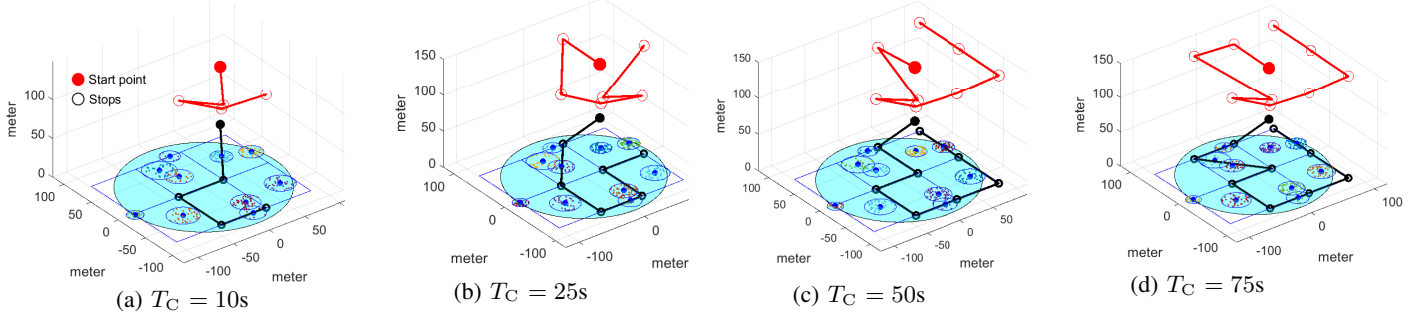


Fig. 10: Different trajectories vs. T_C .

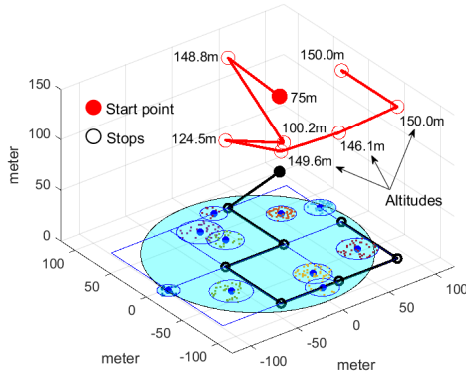


Fig. 11: Trajectory over an RFG of $M_W \times M_L = 3 \times 4$ ($\bar{C} = 3.2\text{Gbps}$, $d = 537.2\text{m}$, and $\bar{T} = 38.5\text{s}$).

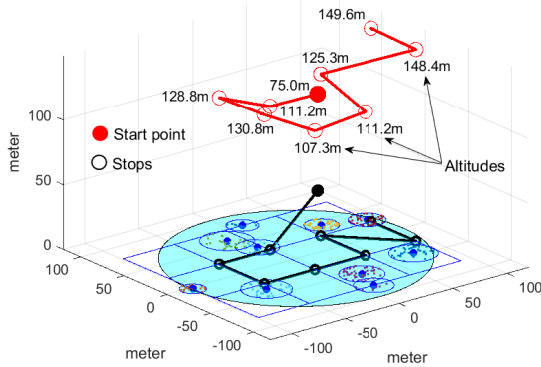


Fig. 12: Trajectory over an RFG of $M_W \times M_L = 4 \times 5$ ($\bar{C} = 3.6\text{Gbps}$, $d = 515.5\text{m}$, and $\bar{T} = 34.9\text{s}$).

C. Trajectory

Last but not least, we carefully investigate the trajectory of UAV versus different scenarios as shown in Fig. 10 and Fig. 11. Following the results in Table V, Fig. 10 elaborates on the trajectories of UAV versus different constraints on service time (T_C). It is clear that relaxing T_C from 10s to 75s stimulates the UAV to fly more stops over longer distances, i.e., from 4 stops over 339.6m to 9 stops over 709.9m. So, the UAV provides

more IDUs with higher sum rate when T_C increases, obviously satisfying the flight power constraint (P_{FC}). Fig. 11 shows that the altitudes of UAV are optimally found to make it stop at the right places in accordance with the diverse characteristics of propagation environments and various densities of IDUs in different clusters so as to enhance the sum rate compared to flying at fixed altitude of $h_{l,w} = 125\text{m}$. Here $h_{l,w} = 125\text{m}$ is the best altitude selected in the range of $[50, 150]\text{m}$ for the TFS-C&S, TFS-OOC, and TFS-NUL methods.

It is noted that due to the limit of computing resource, the aforementioned results are all derived from the flight trajectory over the RFG of $M_W \times M_L = 3 \times 4$ stops, e.g., see Fig. 11. Practically, the RFG is finely divided into more stops so that the UAV can fly to approach the clusters more exactly. So, we further provide an extension case when $M_W \times M_L = 4 \times 5$ as shown in Fig. 12. Compared to the result in Fig. 11, the fine-grid trajectory in Fig. 12 gains higher sum rate (3.6Gbps vs. 3.2Gbps), shorter distance (515.5m vs. 537.2m), and faster service time (34.9s vs. 38.5s), even it flies over one more stop (8 stops vs. 7 stops). And thus, the proposed TCP-3DF can be feasibly deployed by a powerful computing system in reality.

VII. CONCLUSION

We have proposed a joint optimisation method for the UAV-assisted MCD system in IoT networks. In technical perspective, the UAV is integrated with the emerging SDE techniques, i.e., caching, transmission and flight power allocation, multicasting, and U2C association-based 3DF. Regarding the aspects of multimedia contents, IDUs, and wireless propagation environments, the system considers 1) different size and popularity of contents, 2) density and common interest of IDUs, and 3) suburban, urban, dense urban, and high-rise urban environments. These techniques and aspects are formulated as a multi-objective TCP-3DF optimisation problem for the optimal results of binary caching and stop indexes, real transmission and flight powers and altitudes, and integer stop sequence, under the constraints on system resources and service time. GA is modified to solve the TCP-3DF problem

with respect to the complicated OV's and constraints. The results demonstrate that the TCP-3DF method outperforms the other conventional ones to provide the IDUs with maximum sum rate, proper service time required by various multimedia A&Ss over the approximate shortest path, while conserving the storage and power resources. The analyses and discussions contribute valuable insights to the design of SDE techniques using UAV for MCD in IoT networks. Future work will explore more powerful computing and optimisation methods, such as distributed computing and machine learning, for cooperative multi-UAV deployment over larger areas.

REFERENCES

- [1] Ericsson, "IoT connections outlook," Jan. 31, 2024 [Online]. Available: <https://www.ericsson.com/en/reports-and-papers/mobility-report>.
- [2] T. Q. Duong, K. J. Kim, Z. Kaleem, M.-P. Bui, and N.-S. Vo, "UAV caching in 6G networks: A survey on models, techniques, and applications," *Physical Communication*, vol. 51, pp. 1–19, Apr. 2022.
- [3] B. Chen, L. Liu, M. Sun, and H. Ma, "IoT-Cache: Toward data-driven network caching for Internet of things," *IEEE Internet of Things J.*, vol. 6, no. 6, pp. 10064–10076, Dec. 2019.
- [4] S. Liao, J. Wu, J. Li, and K. Konstantin, "Information-centric massive IoT-based ubiquitous connected VR/AR in 6G: A proposed caching consensus approach," *IEEE Internet of Things J.*, vol. 8, no. 7, pp. 5172–5184, Apr. 2021.
- [5] N.-S. Vo *et al.*, "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.
- [6] Q.-N. Tran *et al.*, "Multi-rate probabilistic caching optimized video offloading in dense D2D networks," *IEEE Commun. Lett.*, vol. 27, no. 4, pp. 1240–1244, Apr. 2023.
- [7] L. D. Nguyen, A. Kortun, and T. Q. Duong, "An introduction of real-time embedded optimisation programming for UAV systems under disaster communication," *EAI Endorsed Trans. Ind. Netw. and Intell. Syst.*, vol. 5, no. 17, pp. 1–8, Dec. 2018.
- [8] T. Do-Duy *et al.*, "Joint optimisation of real-time deployment and resource allocation for UAV-aided disaster emergency communications," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3411–3424, Nov. 2021.
- [9] Y.-J. Chen *et al.*, "Multi-agent reinforcement learning based 3D trajectory design in aerial-terrestrial wireless caching networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 8201–8215, Aug. 2021.
- [10] J. Ji, K. Zhu, D. Niyato, and R. Wang, "Joint cache placement, flight trajectory, and transmission power optimization for multi-UAV assisted wireless networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5389–5403, Aug. 2020.
- [11] B. Zhang *et al.*, "Joint 3-D position deployment and traffic offloading for caching and computing-enabled UAV under asymmetric information," *IEEE Internet of Things J.*, vol. 10, no. 7, pp. 6312–6323, Apr. 2023.
- [12] M. Liu, J. Yang, and G. Gui, "DSF-NOMA: UAV-assisted emergency communication technology in a heterogeneous Internet of things," *IEEE Internet of Things J.*, vol. 6, no. 3, pp. 5508–5519, June 2019.
- [13] H. Li *et al.*, "Energy harvesting and resource allocation for cache-enabled UAV based IoT NOMA networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9625–9630, Sep. 2021.
- [14] S. Zhang, Z. Shi, and J. Liu, "Joint trajectory design and resource allocation for secure air-ground integrated IoT networks," *IEEE Internet of Things J.*, vol. 10, no. 23, pp. 20458–20471, Dec. 2023.
- [15] Y. Yin *et al.*, "Joint user pairing and resource allocation in air-to-ground communication-caching-charging integrated network based on NOMA," *IEEE Trans. Veh. Technol.*, vol. 72, no. 12, pp. 15819–15828, Dec. 2023.
- [16] B. Ji *et al.*, "Performance analysis of UAV relay assisted IoT communication network enhanced with energy harvesting," *IEEE Access*, vol. 7, pp. 38738–38747, Mar. 2019.
- [17] A. Al-Hilo *et al.*, "UAV-assisted content delivery in intelligent transportation systems-joint trajectory planning and cache management," *IEEE Trans. Intell. Transportation Syst.*, vol. 22, no. 8, pp. 5155–5167, Aug. 2021.
- [18] K. Zhang *et al.*, "Green offloading and trajectory scheduling of rechargeable UAVs in aerial edge networks," in *Proc. IEEE Global Commun. Conf.*, Rio de Janeiro, Brazil, Dec. 2022, pp. 1–6.
- [19] Y. Zeng, X. Xu, and R. Zhang, "Trajectory design for completion time minimization in UAV-enabled multicasting," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2233–2246, Apr. 2018.
- [20] B. Wang *et al.*, "UAV-assisted emergency communications in social IoT: A dynamic hypergraph coloring approach," *IEEE Internet of Things J.*, vol. 7, no. 8, pp. 7663–7677, Aug. 2020.
- [21] S. Zhang, J. Liu, and W. Sun, "Stochastic geometric analysis of multiple unmanned aerial vehicle-assisted communications over Internet of things," *IEEE Internet of Things J.*, vol. 6, no. 3, pp. 5446–5460, June 2019.
- [22] B. Jiang *et al.*, "Multimedia data throughput maximization in Internet-of-things system based on optimization of cache-enabled UAV," *IEEE Internet of Things J.*, vol. 6, no. 2, pp. 3525–3532, Apr. 2019.
- [23] X. Yu *et al.*, "The heterogeneous demands satisfaction in IoT network: Air-ground collaborative deployment," *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 12713–12724, Dec. 2021.
- [24] K. Rezaee *et al.*, "IoMT-assisted medical vehicle routing based on UAV-borne human crowd sensing and deep learning in smart cities," *IEEE Internet of Things J.*, vol. 10, no. 21, pp. 18529–18536, Nov. 2023.
- [25] B. Li *et al.*, "3D trajectory optimization for energy-efficient UAV communication: A control design perspective," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 4579–4593, June 2022.
- [26] L. Breslau *et al.*, "Web caching and zipf-like distributions: Evidence and implications," New York, NY, Mar. 1999, pp. 126–134.
- [27] T. C. Lam *et al.*, "Genetic algorithms for storage- and energy-aware caching and trajectory optimisation problem in UAV-assisted content delivery networks," in *Proc. EAI Int. Conf. on Ind. Netw. and Intell. Syst.*, Ho Chi Minh City, Vietnam, Aug. 2023, pp. 34–50.
- [28] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements," *IEEE Commun. Lett.*, vol. 7, no. 1, p. 38–41, Feb. 2018.
- [29] W. Saad, M. Bennis, M. Mozaffari, and X. Lin, "Chapter 7 in *Wireless Communications and Networking for Unmanned Aerial Vehicles*", 1st ed. Cambridge University Press, 2020.
- [30] A. Hourani, S. Kandeepan, and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," in *Proc. IEEE Global Commun. Conf.*, Austin, TX, Dec. 2014, pp. 1–7.
- [31] ITU-R, "Rec. p.1410-2 propagation data and prediction methods for the design of terrestrial broadband millimetric radio access systems," *Series, Radiowave propagation*, 2003.
- [32] M.-H. T. Nguyen *et al.*, "Spectrum-sharing UAV-assisted mission-critical communication: Learning-aided real-time optimisation," *IEEE Access*, vol. 9, pp. 11622–11632, Jan. 2021.
- [33] R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," in *Proc. IEEE Int. Commun. Conf.*, AKuala Lumpur, Malaysia, May 2016, pp. 1–5.
- [34] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: Past, present, and future," *Multimedia Tools and Applications*, vol. 80, p. 8091–8126, 2021.
- [35] X. Qi, S. Khattak, A. Zaib, and I. Khan, "Energy efficient resource allocation for 5G heterogeneous networks using genetic algorithm," *IEEE Access*, vol. 9, pp. 160510–160520, Nov. 2021.
- [36] H. Li *et al.*, "Energy-efficient task offloading of edge-aided maritime UAV systems," *IEEE Trans. Veh. Technol.*, vol. 72, no. 1, pp. 1116–1126, Jan. 2023.
- [37] M. I. Mushtaq *et al.*, "Framework for optimized resource allocation in multi-user, multi-service, multi-device aerial networks," *IEEE Access*, vol. 12, pp. 54866–54878, Apr. 2024.
- [38] T. Fang and L. P. Chau, "GOP-based channel rate allocation using genetic algorithm for scalable video streaming over error-prone networks," *IEEE Trans. Image Processing*, vol. 15, no. 6, pp. 1323–1330, Jun. 2006.
- [39] N.-S. Vo, T. Q. Duong, H. D. Tuan, and A. Kortun, "Optimal video streaming in dense 5G networks with D2D communications," *IEEE Access*, vol. 6, pp. 209–223, Oct. 2017.
- [40] H. Zhang, X. Cao, J. K. L. Ho, and T. W. S. Chow, "Object-level video advertising: An optimization framework," *IEEE Trans. Industrial Informatics*, vol. 13, no. 2, pp. 520–531, Apr. 2017.
- [41] G. Lai, F. F. Leymarie, and W. Latham, "On mixed-initiative content creation for video games," *IEEE Trans. Games*, vol. 14, no. 4, pp. 543–557, Dec. 2022.
- [42] A. Chipperfield, P. Fleming, H. Pohlheim, and C. Fonseca, "Genetic Algorithm TOOLBOX For Using with Matlab Ver 1.2 Users Guide". University of Sheffield, 1994.



Thuong C. Lam received the B.E and M.E. degrees in Electronics and Telecommunications from Ho Chi Minh City (HCMC) University of Technology, VNU-HCM, Vietnam, in 2002 and 2005, respectively, and the B.BA degree from University of Economics HCMC, Vietnam, in 2010. He is currently pursuing the Ph.D. degree with the Institute of Fundamental and Applied Sciences, Duy Tan University, HCMC, Vietnam. He is also the lecturer at HUTECH Institute of Engineering, HUTECH University, HCMC, Vietnam.



Van-Ca Phan received the B.E and M.E. degrees in Electronics and Telecommunications from Ho Chi Minh City University of Technology, VNU-HCM, Vietnam, in 2002 and 2005, respectively, and the Ph.D. degree in Electronics and Radio Engineering from Kyung Hee University, South Korea, in 2011. He is currently an Associate Professor in the Faculty of Electrical and Electronics Engineering at Ho Chi Minh City University of Technology and Education, Vietnam. His research interests encompass wireless sensor networks, wireless and mobile communications, the Internet of Things (IoT), energy-efficient protocol design, and the integration of machine learning into wireless systems. Currently, his research focuses on cutting-edge projects in edge computing for video delivery systems in 6G networks and optimizing protocols for ultra-dense wireless environments, driving innovation in next-generation wireless communication technologies.



Nguyen-Son Vo received the B.E and M.E. degrees in Electronics and Telecommunications from Ho Chi Minh City University of Technology, VNU-HCM, Vietnam, in 2002 and 2005, respectively, and the Ph.D. degree in communication and information systems from Huazhong University of Science and Technology, China, in 2012. He is with the Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh City, Vietnam. His research interests focus on content caching and delivering in wireless networks, multimedia wireless communications, quality of experience and sustainability provision in wireless networks for smart cities, and for disaster and environment management. He received the Best Paper Award at the IEEE Global Communications Conference 2016 and the prestigious Newton Prize 2017.



Minh-Phung Bui received the Ph.D. degree in Computer Science from Duy Tan University, Vietnam, in 2022. He is currently serving as the Vice Dean of the Faculty of Information Technology at Van Lang University, Ho Chi Minh City, Vietnam. His primary research interests include optimizing video storage and distribution in wireless networks, multimedia wireless communication, and computational algorithm optimization.



Chan Dai Truyen Thai (S'10–M'14) received the B.S. degree from Posts and Telecommunications Institute of Technology (PTIT), Ho Chi Minh City, Vietnam; the M.Sc. degree from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea; and the Ph.D. degree from Aalborg University, Denmark, in 2003, 2008, and 2012, respectively. He was with IFSTTAR, LEOST, Villeneuve d'Ascq, France; Singapore University of Technology and Design (SUTD); and is now the Vice Dean of the Faculty of Engineering, Vietnamese-German University (VGU). His research interests include cooperative communications, vehicle-to-vehicle communications, communication for high-speed vehicles, security in wireless communications, computer security, security in smart grid, and distributed networks.



Haejoon Jung (Senior Member, IEEE) received the B.S. degree (Hons.) from Yonsei University, South Korea, in 2008, and the M.S. and Ph.D. degrees from the Georgia Institute of Technology (Georgia Tech), Atlanta, GA, USA, in 2010 and 2014, respectively, all in electrical engineering. From 2014 to 2016, he was a Wireless Systems Engineer at Apple, Cupertino, CA, USA. From 2016 to 2021, he was with Incheon National University, Incheon, South Korea. Since September 2021, he has been with the Department of Electronic Engineering, Kyung Hee University, as an Associate Professor. His research interests include communication theory, wireless communications, wireless power transfer, and statistical signal processing. He was a recipient of the Haedong Young Scholar Award from the Korean Institute of Communications and Information Sciences (KICS) in 2022. He has served as an Editor for IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON AEROSPACE and ELECTRONIC SYSTEMS, IEEE COMMUNICATIONS LETTERS, IEEE WIRELESS COMMUNICATIONS LETTERS, and ICT Express.

Genetic Algorithms for Swarm UAV-Assisted Edge Content Delivery Optimisation Problem

Thuong C. Lam¹, Nguyen-Son Vo^{2,*}, Minh-Phung Bui³, De-Thu Huynh⁴, Thanh-Dung Tran², and Ha Duy Hung⁵

¹HUTECH Institute of Engineering, HUTECH University, Ho Chi Minh City, Vietnam (e-mail: lc.thuong@hutech.edu.vn)

²Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh City, Vietnam

(*Corresponding author, e-mail: vonguyenson@duytan.edu.vn, tdungtran@gmail.com)

³Faculty of Information Technology, School of Technology, Van Lang University, Ho Chi Minh City, Vietnam
(e-mail: phung.bm@vlu.edu.vn)

⁴School of Computer Science & Engineering, The Saigon International University, Ho Chi Minh City, Vietnam
(e-mail: huynhdethu@siu.edu.vn)

⁵Wireless Communications Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam (e-mail: haduyhung@tdtu.edu.vn)

Abstract—Swarm unmanned aerial vehicles (UAV) have been increasingly studied for delivering the contents in 6G networks. In this paper, we propose an energy-aware capacity and trajectory (ECT) optimisation solution for swarm UAV-assisted edge content delivery networks (eCDN). The ECT optimisation solution enables a swarm UAV system to cooperatively fly 1) to the optimal set of stops and serve the user equipments in different clusters at maximum content delivery capacity and 2) in the optimal trajectory planning at minimum distance, under the constraint of energy resource. However, the ECT optimisation problem is challenging due to multiple objectives with respect to different optimisation variable types and complicated constraints. Genetic algorithms (GA) are therefore modified to address the challenge of solving the so-called multi-objective optimisation problem feasibly. Simulation results are shown to demonstrate the feasibility of GA and the benefits of ECT method for swarm UAV-assisted eCDN.

Index Terms—6G networks, edge content delivery networks, genetic algorithms, swarm UAV networks, trajectory planning.

I. INTRODUCTION

With the development of mobile device technology, advanced applications and services (AAS), Internet service providers, and content providers, content delivery networks (CDN) have commonly considered as an important pillar to improve the experience of mobile users [1], [2]. However, in the era of Internet of Things (IoT), a proliferation of wide-area, cellular, and short-range IoT devices and users, namely user equipments (UE), which requests various AASs with high transmission rate and huge volume of data, e.g., multimedia applications and services, makes such central CDNs degraded due to severe bottleneck situations at the backhaul links of 6G networks [3]–[5]. The conventional CDNs have been therefore expanded next to the UEs by integrating with edge networks, forming an emerging architecture of edge CDNs (eCDN).

Recently, we have witnessed an efficient launch of unmanned aerial vehicles (UAV) based flying platform in assisting the eCDN to serve the UEs. On the one hand, by caching, the UAVs can bring the contents closer to the UEs for reducing the transmission distance. On the other hand, the

UAVs are planned to fly in an optimal trajectory to probably achieve the line-of-sight (LoS) propagation for improving the transmission rate and to expand the functional coverage area for providing more UEs with a much higher quality of service [4]. The key solution for UAVs is the combination of caching and flight trajectory, UE association, or/and power allocation optimisation techniques studied in [6]–[11].

In particular, the work in [6] introduced a joint caching and trajectory (JCT) optimisation problem under the constraints on caching storage and total energy consumed by transmitting and flying. Applying exhausted and greedy search algorithms, the JCT optimisation problem is solved for optimally selecting which contents to cache (C2C), deciding where to stop (W2S), and planning how to fly (H2F) throughout the stops. The objective is to maximise the content delivery capacity while minimising the flight distance and conserving the storage and energy resources. In [7], the authors have formulated an intractable distributionally robust stochastic optimisation problem for the optimal results of C2C and W2S under the constraint on the storage resource. The intractable optimisation problem is reformulated as a tractable semi-definite programming problem so that the solution is probably obtained to maximise the reduced delay brought by the UAV.

To exploit the storage resource, the authors in [8] have further considered caching in mobile devices (MD) to propose a cache-enabling UAV-device-to-device (UAV-D2D) network. The ideas of C2C in UAV, caching placement in MDs, and W2S are formulated in the form of a mixed integer nonlinear programming problem for maximising the cache utility under the constraint on storage resource. To reduce the complexity, the original optimisation problem is decomposed into three sub-problems which can be feasibly solved by dynamic programming for C2C in UAV, many-to-many swap matching for caching placement in MDs, and approximate convex optimisation for W2S. Interestingly, by applying reinforcement learning method in [9], [10], UE association and power allocation are optimised together with caching placement and W2S of

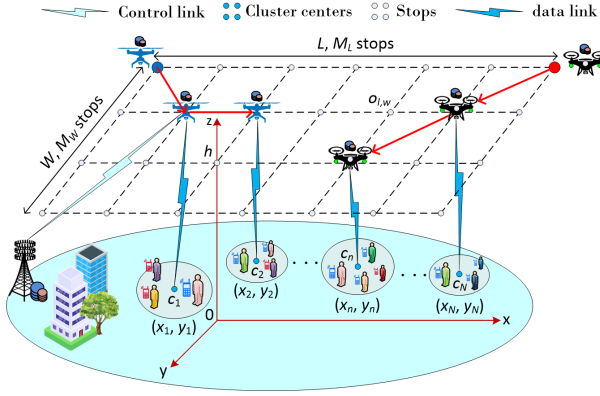


Fig. 1: Swarm UAV-assisted eCDN.

multiple UAVs for minimising the total content delay.

We can see that the deployment of the above works falls into three cases: 1) single UAV with H2F for minimum flight distance and multiple objectives of high complexity for exact results [6], 2) single UAV without H2F and single objective of lower complexity for approximate or exact results [7], [8], and 3) multiple UAVs without H2F and single objective of low complexity for approximate or exact results [9], [10]. In our previous work [11], the time complexity in [6] is reduced by applying genetic algorithms (GA) for approximate or exact results, but there still has only one UAV deployed.

In this paper, we propose an energy-aware capacity and trajectory (ECT) optimisation solution for swarm UAV-assisted eCDN. Assuming that the UAVs can cache all the contents requested by the UEs in different clusters, the ECT optimisation problem is formulated in the form of multiple objectives with respect to (w.r.t) two optimisation variable types of W2S (binary variable) and H2F (integer variable) under the complicated constraint on transmission and flight energy resource. GA is modified to feasibly solve the complicated ECT optimisation problem to maximise the content delivery capacity and minimise the flight distance handled by the available energy resource. The feasibility of GA (accuracy and time complexity) and the benefits of ECT method for swarm UAV-assisted eCDN (capacity, distance, and service time) are investigated by simulation results.

The rest of this paper is organised as follows. In Section II, we introduce the swarm UAV system for eCDN and describe how it works. The system is formulated in Section III. The formulations enable us to propose the ECT optimisation problem and solution with GA in Section IV. In Section V, the performance of GA and ECT method is evaluated with insightful analysis and discussion. Finally, Section VI is dedicated to concluding the paper.

II. SYSTEM MODEL

In this paper, we consider a system of swarm UAV-assisted eCDN in the same model with [11] but multiple UAVs cooperatively deployed to serve the UEs. We further assume that each UAV flies to serve a limited number of clusters within a coverage radius R of the macro base station (MBS) and

TABLE I: Main notations

Notations	Descriptions
R	Radius flight area of UAVs covered by MBS
M	Number of UAVs
N	Number of UE clusters
I	Number of contents
$N_T^{(m)}$	Number of stops in the optimal trajectory $\mathcal{T}_m$ where UAV m flies throughout, $m = 1, 2, \dots, M$
p_i	Popularity of content i , $p_i = i^{-\alpha} (\sum_{i=1}^I i^{-\alpha})^{-1}$ [12]
N_i	Number of UE clusters requesting content i , $N_i = \lfloor p_i N \rfloor$, the method to adjust N_i for ensuring $N = \sum_{i=1}^I N_i$ is given in [11]
$a_{i,n}$	UAV-cluster association index, $a_{i,n} = 1$ if at least one EU in cluster n request content i , $a_{i,n} = 0$ otherwise, $N_i = \sum_{n=1}^N a_{i,n}$
h	Fixed altitude of UAVs
$o_{l,w}$	Horizontal coordinates of stops, $o_{l,w} = (x_l, y_w)$, $l = 1, 2, \dots, M_L$ and $w = 1, 2, \dots, M_W$
$d(a, b)$	Distance between stop a and stop b
$F(o_{l,w}^{(m)})$	Stop index matrix of UAV m , if UAV m stops at $o_{l,w}$, $F(o_{l,w}^{(m)}) = 1$, otherwise $F(o_{l,w}^{(m)}) = 0$, $N_T^{(m)} = \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} F(o_{l,w}^{(m)})$
c_n	Center of UE cluster n located at (x_n, y_n) , $n = 1, 2, \dots, N$

thus it can cache all the requested contents. Therefore, C2C optimisation is not utilised in the system formulations. The main notations, descriptions, and expressions of the system are given in Table I. After collecting all the system parameters, the MBS solves the ECT optimisation problem for the optimal solutions of W2S and H2F. As a result, UAV m is assigned to fly to $N_T^{(m)}$ stops so that all UAVs can cooperatively fly to provide the UEs with maximum content delivery capacity, minimum flight distance, and faster service time properly handled by the available energy resource.

For convenience of content delivery and trajectory design, in each UE cluster with center c_n , a cluster head (CH) is elected to receive the contents delivered from the UAVs and then forwarding them to its' members. In addition, the UAVs are planned to fly throughout the stops located in a rectangular grid of length $L(m)$ and width $W(m)$, at the fixed altitude h . Given N UE clusters distributed in a circular area with radius R , the rectangular grid is the one with smallest area that covers all the centers of the UE clusters. There are M_L stops and M_W stops equally distributed along the length and width of the rectangular forming the $M_L \times M_W$ stop matrix for the UAVs to fly. The number of stops in the matrix depends on the density of UE clusters and the accuracy requirement of the ECT method for the swarm UAV-assisted eCDN.

III. SYSTEM FORMULATIONS

A. Average Content Delivery Capacity

To derive the average content delivery capacity, we assume that the UAVs utilise the inband frequency (licensed band) and outband frequency (unlicensed band) for communicating with the MBS and the CHs, respectively. Furthermore, CH n in UE cluster n is the one located at or nearby c_n . The capacity to

deliver content i from UAV m stopping at $o_{l,w}^{(m)}$ to CH n is given by

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

where B is the system bandwidth, and $\gamma_{l,w}^{(i,n,m)}$, which is the signal-to-noise ratio of the channel between UAV m at $o_{l,w}$ and CH n , is dominated by LoS propagation [13], [14], expressed as

$$\gamma_{l,w}^{(i,n,m)} = \frac{a_{i,n} P_T \beta_0 (d_{l,w}^{(n,m)})^{-2}}{\sigma^2}. \quad (2)$$

In (2), P_T is the transmission power of all UAVs, β_0 is the power gain of UAV-to-CH channel at the reference distance $d_0 = 1$ m, σ^2 is the additive white Gaussian noise power, and $d_{l,w}^{(n,m)}$ is the distance between UAV m at $o_{l,w}$ and CH n computed as

$$d_{l,w}^{(n,m)} = \sqrt{h^2 + \|o_{l,w} - c_n\|^2}, \quad (3)$$

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

Finally, the average content delivery capacity per cluster provided by all UAVs is given by

$$\bar{C} = \frac{1}{N} \sum_{n=1}^N \sum_{m=1}^M \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^I p_i F(o_{l,w}^{(m)}) C_{l,w}^{(i,n,m)}, \quad (4)$$

where, $F(o_{l,w}^{(m)})$, which is the stop index matrix defined in Table I, is optimally found by W2S to maximise $\bar{C}$.

B. Flight Distance

Given $N_{\mathcal{T}}^{(m)}$ stops, let $d_{n_{\mathcal{T}}}^{(m)}$ be the distance from stop $(n_{\mathcal{T}} - 1)$ to stop $n_{\mathcal{T}}$ for UAV m to fly in the optimal stop sequence $\mathcal{T}_m$ found by H2F, the total flight distance of all UAVs is expressed as

$$d = \sum_{m=1}^M d_m = \sum_{m=1}^M \sum_{n_{\mathcal{T}}=1}^{N_{\mathcal{T}}^{(m)}} d_{n_{\mathcal{T}}}^{(m)}. \quad (5)$$

In (5), $d_1^{(m)}$ is the distance from the original position of UAV m to the first stop in $\mathcal{T}_m$. With H2F, obviously minimising d is equivalent to minimising each d_m separately. Given a sub-set $\mathcal{T}_s^{(m)}$ of $N_{\mathcal{T}}^{(m)}$ stops for UAV m selected from the whole set $\mathcal{T}$ of N stops, i.e., $\mathcal{T}_s^{(m)} \in \mathcal{T}$, $\mathcal{T}_m$ for UAV m to fly that can minimise the distance d_m is certainly an arbitrary combination of the sub-set $\mathcal{T}_s^{(m)}$. For example, assume that $N = 5$ and $\mathcal{T} = \{1, 2, 3, 4, 5\}$, with $M = 2$, we may have $\mathcal{T}_s^{(1)} = \{1, 3, 4\}$, $\mathcal{T}_s^{(2)} = \{2, 5\}$. So, d_1 and d_2 are minimised respectively by finding the best combinations of $\mathcal{T}_s^{(1)}$ among $\{1, 3, 4\}$, $\{1, 4, 3\}$, $\{3, 1, 4\}$, $\{3, 4, 1\}$, $\{4, 1, 3\}$, and $\{4, 3, 1\}$ and of $\mathcal{T}_s^{(2)}$ between $\{2, 5\}$ and $\{5, 2\}$, to obtain $\mathcal{T}_1$ and $\mathcal{T}_2$.

C. Energy Consumption

There are three types of energy consumed by flight, transmission, and standstill. To obtain the energy consumption, we need to derive the durations of flight and transmission/standstill¹. It is certain that the total flight duration of all UAVs is

$$t_F = d/V, \quad (6)$$

where V is the flight speed of the UAVs.

The standstill (or transmission) duration of UAV m at $o_{l,w}^{(m)}$ to transmit content i to CH n is expressed as

$$t_{l,w}^{(i,n,m)} = \begin{cases} \frac{F(o_{l,w}^{(m)}) s_i}{C_{l,w}^{(i,n,m)}}, & \text{if } a_{i,n} = 1, \\ 0, & \text{if } a_{i,n} = 0. \end{cases} \quad (7)$$

So, the total standstill (or transmission) duration is given by

$$t_T = \sum_{n=1}^N \sum_{m=1}^M \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^I t_{l,w}^{(i,n,m)}. \quad (8)$$

By taking into account the flight power (P_F), standstill power (P_S), and transmission power (P_T), the total energy consumption is expressed as

$$E_C = P_F t_F + (P_S + P_T) t_T. \quad (9)$$

D. Average Service Time

The average service time is defined as the average flight and transmission duration of the UAVs to serve each CH, i.e., equivalent to the average waiting and receiving time of each CH to be served by the UAVs. In particular, the flight duration over hop $n_{\mathcal{T}}$, $n_{\mathcal{T}} = 1, 2, \dots, N_{\mathcal{T}}^{(m)}$, is given by

$$t_{n_{\mathcal{T}}}^{(m)} = \frac{d_{n_{\mathcal{T}}}^{(m)}}{V}. \quad (10)$$

Together with the transmission duration, the average service time per cluster is given by

$$\bar{T} = \frac{1}{N} \left[\sum_{n_{\mathcal{T}}=1}^{N_{\mathcal{T}}^{(m)}} \sum_{m=1}^M (N_{\mathcal{T}}^{(m)} - n_{\mathcal{T}} + 1) t_{n_{\mathcal{T}}}^{(m)} + \sum_{n=1}^{N-1} \sum_{m=1}^M \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} \sum_{i=1}^I (N - n) t_{l,w}^{(i,n,m)} \right]. \quad (11)$$

IV. ECT OPTIMISATION PROBLEM AND SOLUTION

A. ECT Optimisation Problem

To maximise the average content delivery capacity $\bar{C}$ and minimise the total flight distance d of all UAVs simultaneously under the constraint on the available energy resource E , mathematically, the ECT optimisation problem is formulated by merging W2S and H2F as below

$$\max_{F(o_{l,w}^{(m)}), \mathcal{T}_m} \bar{C} - \gamma d \quad (12a)$$

¹The standstill duration is also for transmission.

$$\sum_{m=1}^M N_{\mathcal{T}}^{(m)} \leq N, \quad (12b)$$

$$E_C \leq E, \quad (12c)$$

where $\gamma > 0$ is used to balance the trade-off between the two objectives of maximising $\bar{C}$ and minimising d .

B. ECT Solution with GA

1) *Penalty Function Design:* In (12), because a UAV can fly throughout $K = M_L \times M_W$ stops to transmit the requested contents as long as it has enough energy resource. The binary stop index matrix $F(o_{l,w}^{(m)})$ for UAV m therefore has K bits. For the ease of GA implementation, the matrix $F(o_{l,w}^{(m)})$ is mapped onto a binary array (string) f_m of length K bits so that we have $N_{\mathcal{T}}^{(m)} = \sum_{l=1}^{M_L} \sum_{w=1}^{M_W} F(o_{l,w}^{(m)}) = \sum_{k=1}^K f_m(k)$. However, f_m from the initial generation of random individuals and during the crossover and mutation processes may be invalid resulting in the fact that multiple UAVs fly to the same stops. To avoid this invalid situation, let $f = \sum_{m=1}^M f_m$, $f(k) > 1$ means that there are multiple UAVs fly to stop k and thus there is a punishment for this invalidity. The number of invalid positions punished in f is given by

$$f_{\text{inv}} = \sum_{k=1, f(k) > 1}^K 1. \quad (13)$$

Similarly, the stop sequence $\mathcal{T}_m$ in general has N integers from 1 to N . However, UAV m only uses the first $N_{\mathcal{T}}^{(m)}$ integers in $\mathcal{T}_m$ in each generation of GA, the rest $N - N_{\mathcal{T}}^{(m)}$ integers are ignored. Due to the initial generation of random individuals and the crossover and mutation processes, $\mathcal{T}_m$ is likely invalid resulting the fact that UAV m flies to the next stop(s) where it has visited earlier. Let $\mathcal{T}'_m = [\]$ and $d_{\text{pun}}^{(m)} = 0$, to avoid the invalidity, the punishment is performed by increasing $d_{\text{pun}}^{(m)}$ by 1 when computing $d_{n_{\mathcal{T}}}^{(m)}$ as below

$$d_{n_{\mathcal{T}}}^{(m)} = \begin{cases} d(\mathcal{T}_m(n_{\mathcal{T}} - 1), \mathcal{T}_m(n_{\mathcal{T}})), & \text{if } \mathcal{T}_m(n_{\mathcal{T}}) \notin \mathcal{T}'_m, \\ \mathcal{T}'_m = [\mathcal{T}'_m \ \mathcal{T}_m(n_{\mathcal{T}})], & \\ 0, & \text{if } \mathcal{T}_m(n_{\mathcal{T}}) \in \mathcal{T}'_m, d_{\text{pun}}^{(m)} = d_{\text{pun}}^{(m)} + 1. \end{cases} \quad (14)$$

The total punishment for all UAVs flying to invalid stops is given by

$$d_{\text{pun}} = \sum_{m=1}^M d_{\text{pun}}^{(m)}. \quad (15)$$

Furthermore, GA can easily deal with a simple constraint in the form of lower bounds and upper bounds of optimisation variables. However, the ECT optimisation problem consists of two complicated constraints (12b) and (12c). For complicated constraints, GA further applies penalty function which requires to rewrite (12b) and (12c) as below

$$\begin{cases} \Delta N = N - \sum_{m=1}^M N_{\mathcal{T}}^{(m)} \geq 0, \\ \Delta E = E - E_C \geq 0. \end{cases} \quad (16)$$

So, the whole penalty function, which is used to punish the individuals if they violate the constraints, is given by

$$F = \lambda_1 (f_{\text{inv}})^2 + \lambda_2 (d_{\text{pun}})^2 + \lambda_3 (\min\{0, \Delta N\})^2 + \lambda_4 (\min\{0, \Delta E\})^2, \quad (17)$$

Algorithm 1 GA Implementation

Input: System and GA parameters in Table II

TC : Termination conditions

$Gen = 1$: Generation count

Output: $\bar{C}_F^*$, f_m^* , and $\mathcal{T}_m^*$

- 1: Randomly generating N_P binary strings $\{X_B^{(z)}\}$, $z = 1, 2, \dots, N_P$, each has MK bits to represent individual z
- 2: Randomly generating N_P integer strings $\{X_I^{(z)}\}$, each has MN integers to represent individual z
- 3: Computing N_P fitness values $\bar{C}_F(X_B^{(z)}, X_I^{(z)})$, i.e. $\bar{C}_F^{(z)}$
- 4: **while** TC does not hold **do**
- 5: Putting $\bar{C}_F^{(z)}$ w.r.t $\{X_B^{(z)}\}$ and $\{X_I^{(z)}\}$ in the mating pool for ranking
- 6: Selecting $N_G = N_P P_G$ better individuals with higher $\bar{C}_F^{(z)}$ for breeding by using stochastic universal sampling operator [15], so as to get $\{X_B^{(t)}\}$ and $\{X_I^{(t)}\}$, $t = 1, 2, \dots, N_G$
- 7: Selecting a pair of parents to make a pair of offsprings by using 1) multiple point crossover with probability P_{CB} for $\{X_B^{(t)}\}$ and 2) single point crossover with probability P_{CI} for $\{X_I^{(t)}\}$ [15]
- 8: Mutating the offsprings $\{X_B^{(t)}\}$ with probability P_{MB} and the offsprings $\{X_I^{(t)}\}$ with probability P_{MI} , to recover the good genetic materials likely lost in the previous steps
- 9: Computing N_G fitness values $\bar{C}_F^{(t)}$ w.r.t $\{X_B^{(t)}\}$ and $\{X_I^{(t)}\}$ and reinserting them into the present generation
- 10: $Gen = Gen + 1$
- 11: **end while**
- 12: Finding the best fitness value $\bar{C}_F^*$ associated with f_m^* and $\mathcal{T}_m^*$ extracted from $\{X_B^{(z)}\}$ and $\{X_I^{(z)}\}$ in the last generation

where λ_1 , λ_1 , λ_3 , and λ_4 are the constraint violation degrees.

2) *GA Implementation:* By combining (12) with the penalty function, the GA can be used to solve the unconstrained ECT optimisation problem given by

$$\max_{f_m, \mathcal{T}_m} \bar{C}_F = \bar{C} - (\gamma d + F). \quad (18)$$

The detailed GA implementation to solve (18) is shown in Algorithm 1. Termination condition (TC) of GA holds whether $\bar{C}_F$ does not change while the average penalty value per individual $\bar{F} \leq 10^{-3}$ in 3 successive generations or $gen = N_G$.

V. PERFORMANCE EVALUATION

Table II provides the system and GA parameters. GA is executed by a desktop computer with detailed information listed in Table III. The UE clusters are randomly distributed within a circle with $R = 100\text{m}$. Three UAVs are originally located at $(x_U^{(1)}, y_U^{(1)}) = (-100\text{m}, 100\text{m})$, $(x_U^{(2)}, y_U^{(2)}) = (100\text{m}, -100\text{m})$, and $(x_U^{(3)}, y_U^{(3)}) = (-100\text{m}, -100\text{m})$ with fixed altitude $h = 50\text{m}$.

TABLE II: Parameters setting

Symbols	Specifications
$\{R, N, I, M\}$	{100 meters, 10 clusters, 5 videos, 3 UAVs}
$\{s_i\}$	{300, 100, 200, 400, 900}Mbits
α	1
V	15m/s
$\{P_T, P_F, P_S\}$	{0.5, 50, $0.25 \times P_F$ }W
$\{E, B\}$	{5000Joules, 10MHz}
$\{\beta_0, \sigma^2\}$	{ 10^{-5} , 10^{-13} }W
N_P	10,000 individuals
K	$M_L \times M_W = 4 \times 3 = 12$ stops
$\{N_G, P_G\}$	{100, 0.8}
$\{P_{CB}, P_{CI}, P_{MB}, P_{MI}\}$	{0.6, 0.9, 10^{-6} , 10^{-10} }
$\{\gamma, \lambda_1, \lambda_2, \lambda_3, \lambda_4\}$	{0.05, 10^2 , 10, 10^2 , 10^2 }

TABLE III: Computer information

Processor	12th Gen Intel(R) Core(TM) i7-1255U 1.70GHz
Processor type	x64-based processor
Processor cores/Threads	10/12
Total PHY memory	16GB
Operating System	Windows 11

TABLE IV: ES and GA comparison ($M = 2$)

	$\bar{C}$ (Mbps)	d (m)	E_C (Joules)	Time (s)	Algorithm
6 stops	197.44	471.62	3,565.26	33.08	ES
6 stops	197.44	471.62	3,565.26	300.08	GA
8 stops	260.34	483.99	4,295.38	2,640.34	ES
8 stops	260.34	499.40	4,346.74	343.23	GA

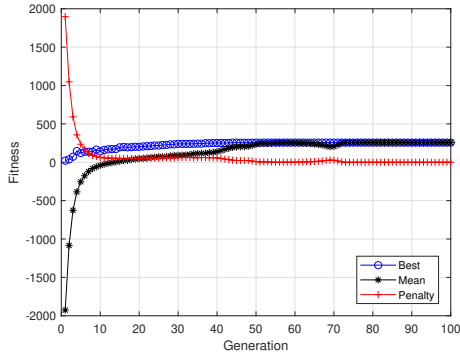
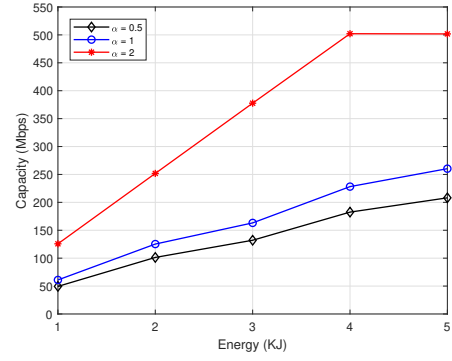
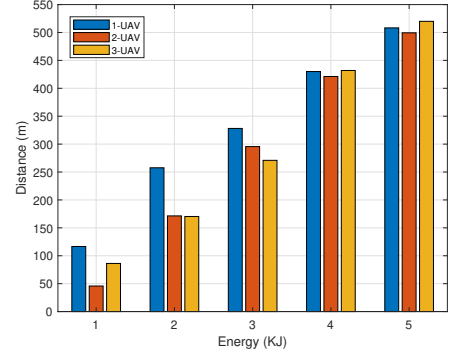
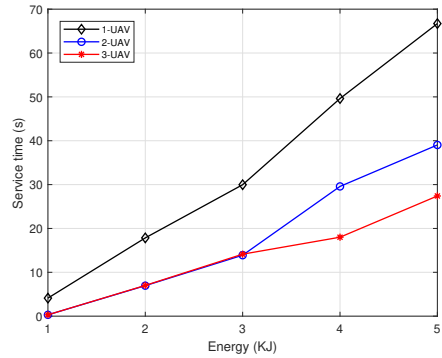


Fig. 2: Convergence rate of GAs.

A. GA Performance

The convergence rate of GA is evaluated by considering 100 generations as shown in Fig. 2. The plots include the best value ($\max\{\bar{C}_z - F_z\}$) and the mean values of $\{\bar{C}_z - F_z\}$ and penalty $\bar{F}$ per individual in each generation. The results show that the convergence situation of GAs starts from the 50th generation when the mean value is closer to the best value and $\bar{F} = 0$. It indicates that all the individuals become better after each generation while ensuring the given constraints for optimal solutions. We further evaluate the GA performance by comparing to exhausted search (ES) algorithm which is implemented similarly to the work in [6], i.e., W2S to maximise $\bar{C}$ and H2F for each UAV to minimise the flight distance d . The results in Table IV show that if the number of stops is small (6 stops), GA has the same optimal solutions with ES but obviously ES outperforms GA in terms

Fig. 3: Capacity versus E with different values of α .Fig. 4: Flight distance versus E with different values of M .Fig. 5: Service time versus E with different values of M .

of time complexity (33.08s vs. 300.08s). When increasing to 8 stops, GA provides exact or approximate optimal solutions but with extremely lower time complexity compared to ES (343.23s vs. 2,640.34s). The convergence rate, accuracy, and time complexity of GA demonstrate that it is feasible to use GA for solving the ECT optimisation problem.

B. Performance of ECT for swarm UAV-assisted eCDN

We evaluate the average content delivery capacity $\bar{C}$ versus different energy constraints E and skewed popularity value α as shown in Fig. 3. It is certain that the increase of E enhances $\bar{C}$ but getting a saturated result due to the limit of N and the fixed transmission power of UAVs. In addition, increasing α enables the ECT method to focus on the contents with higher popularity values and thus achieving higher $\bar{C}$.

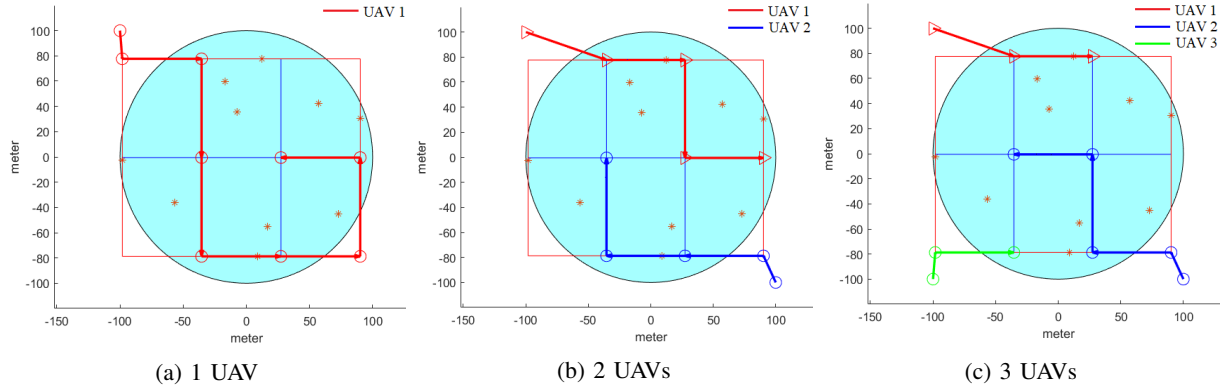
Fig. 6: Trajectories versus different values of M .

Fig. 4 plots the distance versus the energy constraint E with 1 UAV, 2 UAVs, and 3 UAVs. The results show that the higher energy resource the UAVs have, the more stops the UAVs fly to serve more UE clusters, and thus with longer flight distances. The total flight distances associated with 1 UAV, 2 UAVs, and 3 UAVs are different depending on the sets of stops and the original positions of the UAVs. The most benefit of multi-UAV deployment is the significant reduction in service time ($\bar{T}$) as shown in Fig. 5. We can see that the higher number of UAVs (2 UAVs and 3 UAVs) significantly reduces $\bar{T}$ when E is high enough ($E = 3$ KJ).

To fulfill the evaluation of ECT for swarm UAV-assisted eCDN, we investigate the trajectories for 1 UAV, 2 UAVs, and 3 UAVs as shown in Fig. 6. The trade-off of multi-UAV deployment is the distance from the original positions to the first stops added. However, the service time is significantly improved if more UAVs are cooperatively deployed. Assigning an optimal number of UAVs deployed is crucial to meet the service time required by different content applications and services, especially by real-time multimedia communications.

VI. CONCLUSION

We have proposed the multi-objective optimisation problem of ECT for swarm UAV-assisted eCDNs. The ECT optimisation problem consists of two objectives of W2S and H2F w.r.t complicated optimisation variables and constraints. We apply GA to solve the ECT optimisation problem feasibly, i.e., high accuracy and low time complexity, for maximum average content delivery capacity and minimum flight distance flexibly handled by the energy resource of UAVs. The feasibility of GA and benefits of ECT method for swarm UAV-assisted eCDNs are analysed and discussed to provide valuable insights to the design of swarm UAV-assisted eCDNs. In future works, we will explore more powerful computing systems and advanced optimisation algorithms for cooperative multi-UAV deployment over larger areas to provide the UEs with real-time multimedia applications and services.

ACKNOWLEDGEMENT

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.04-2023.25.

REFERENCES

- [1] Y. Gao *et al.*, "Video content placement at the network edge: Centralized and distributed algorithms," *IEEE Trans. Mobile Comput.*, vol. 22, no. 11, pp. 6843–6859, Nov. 2023.
- [2] Y. Wang *et al.*, "Cost-driven data caching in edge-based content delivery networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 3, pp. 1384–1400, Mar. 2023.
- [3] B. Chen, L. Liu, M. Sun, and H. Ma, "IoTCache: Toward data-driven network caching for Internet of things," *IEEE Internet of Things J.*, vol. 6, no. 6, pp. 10064–10076, Dec. 2019.
- [4] T. Q. Duong, K. J. Kim, Z. Kaleem, M.-P. Bui, and N.-S. Vo, "UAV caching in 6G networks: A survey on models, techniques, and applications," *Physical Communication*, vol. 51, pp. 1–19, Apr. 2022.
- [5] S. Liao, J. Wu, J. Li, and K. Konstantin, "Information-centric massive IoT-based ubiquitous connected VR/AR in 6G: A proposed caching consensus approach," *IEEE Internet of Things J.*, vol. 8, no. 7, pp. 5172–5184, Apr. 2021.
- [6] N.-S. Vo, T. C. Lam, M.-P. Bui, T.-M. Phan, and Q.-N. Tran, "UAV assisted video multicasting in 6G networks: A joint caching and trajectory optimisation," *Journal of Aviation Science and Technology*, vol. 1, no. 1, pp. 37–42, Mar. 2022.
- [7] X. Li, J. Liu, N. Zhao, and X. Wang, "UAV-assisted edge caching under uncertain demand: A data-driven distributionally robust joint strategy," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3499–3511, May 2022.
- [8] T. Zhang, Y. Wang, W. Yi, Y. Liu, and A. Nallanathan, "Joint optimization of caching placement and trajectory for UAV-D2D networks," *IEEE Trans. Commun.*, vol. 7, no. 8, pp. 5514–5527, Aug. 2022.
- [9] H. Xu, J. Ji, K. Zhu, and R. Wang, "Reinforcement learning for trajectory design in cache-enabled UAV-assisted cellular networks," in *Proc. IEEE Wireless Commun. and Networking Conf.*, Austin, TX, Apr. 2022, pp. 1–6.
- [10] J. Ji, K. Zhu, and L. Cai, "Trajectory and communication design for cache-enabled UAVs in cellular networks: A deep reinforcement learning approach," *IEEE Trans. Mobile Comput.*, vol. 22, no. 10, pp. 6190–6204, Oct. 2023.
- [11] T. C. Lam, N.-S. Vo, T.-H. Nguyen, T.-M. Phan, and D.-T. Huynh, "Genetic algorithms for storage- and energy-aware caching and trajectory optimisation problem in UAV-assisted content delivery networks," in *Proc. EAI Int. Conf. on Industrial Networks and Intelligent Systems*, Ho Chi Minh City, Vietnam, Aug. 2023, pp. 34–50.
- [12] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Proc. of IEEE Inter. Conf. on Computer Commun. (INFOCOM)*, New York, NY, Mar. 1999, pp. 126–134.
- [13] X. Xu, Y. Zeng, Y. L. Guan, and R. Zhang, "Overcoming endurance issue: UAV-enabled communications with proactive caching," *IEEE J. Sel. Areas Commun.*, pp. 1231–1244, June 2018.
- [14] H. Wu, F. Lyu, C. Zhou, J. Chen, L. Wang, and X. Shen, "Optimal UAV caching and trajectory in aerial-assisted vehicular networks: A learning-based approach," *IEEE J. Sel. Areas Commun.*, pp. 1–14, June 2020.
- [15] A. Chipperfield, P. Fleming, H. Pohlheim, and C. Fonseca, "Genetic Algorithm TOOLBOX For Using with Matlab Ver 1.2 Users Guide". University of Sheffield, 1994.

Transmission Mode Selection-based Content Caching for Real-time Swarm UAV-assisted 6G IoT Networks

Thuong C. Lam¹, Nguyen-Son Vo^{2,*}, Tan Do-Duy³, Long D. Nguyen⁴,
Thanh-Minh Phan⁵, Hung T. Nguyen⁶, and De-Thu Huynh⁷

¹ HUTECH University, Ho Chi Minh City, Vietnam (lc.thuong@hutech.edu.vn)

² Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh City, Vietnam (vonguyenson@duytan.edu.vn)

³ Ho Chi Minh City University of Technology and Education, Ho Chi Minh City, Vietnam (tandd@hcmute.edu.vn)

⁴ Dong Nai University, Dong Nai, Vietnam (dinhlonghcmute@gmail.com)

⁵ Vietnam Aviation Academy, Ho Chi Minh City, Vietnam (minhpt@vaa.edu.vn)

⁶ Ho Chi Minh City University of Transport, Vietnam (thaihung.nguyen@ut.edu.vn)

⁷ The Saigon International University, Ho Chi Minh City, Vietnam
(huynhdethu@siu.edu.vn)

Abstract. As 6G Internet of Things (IoT) networks evolve toward high data rate communications and low latency applications and services, this paper proposes a transmission mode selection-based content caching (MSC) integrated with real-time swarm UAV trajectory (RTT) optimization to enhance both system sum rate and service time. In MSC, IoT devices and users (IDUs) in different clusters are served cooperatively by unmanned aerial vehicles (UAVs) and macro base station (MBS) to maximize the overall sum rate, while the RTT design further ensures fast service time provided by the UAVs. Both MSC and RTT are jointly formulated as a unified MSC-RTT optimization problem, which determines (i) the optimal transmission mode selection, i.e., whether a UAV with proper cached contents or the MBS should serve the IDUs in a given cluster, and (ii) the optimal set of UAVs and their trajectories for efficient content delivery across clusters. Simulation results demonstrate that the proposed MSC-RTT framework significantly improves both sum rate and service time, outperforming the existing benchmarks in real-time content delivery applications and services over 6G IoT networks.

Keywords: 6G Internet of Things, content delivery applications and services, transmission mode selection, real-time swarm UAVs, UAV caching

1 Introduction

Real-time Internet of Things (IoT) applications and services in 6G networks demand extremely high data rate, low latency, ultra reliability, and massive connectivity to support seamless content delivery across heterogeneous IoT devices

and users (IDUs) [1–4]. Emerging use cases including autonomous transportation, industrial automation, remote health care, and extended reality, further require instantaneous data processing and continuous content updates, under strict resource consumption constraints. With the integration of edge techniques, THz bands, large-scale unmanned aerial vehicles (UAV)-assisted infrastructures, intelligent reflecting surfaces, sensing and communications, artificial intelligence, and digital twins, 6G IoT networks are envisioned to enable highly adaptive and context-aware content delivery strategies that are able to address the diverse real-time demands of applications and services [5].

Regarding real-time system designs for UAV-assisted IoT networks, existing studies have primarily focused on low latency and accuracy-guaranteed scheme [6], in-flight decision making [7, 8], service caching-based data processing and analysis [9], path planning and transmission scheduling [10], trajectory planning for data updating and collection [1, 11, 12]. Particularly, the authors in [6] proposed a joint task assignment, model selection, and resource allocation framework for UAV-assisted IoT networks that achieves low task latency while strictly satisfying accuracy constraints with real-time deep neural networks. The work in [7] designed a UAV path-planning algorithm to optimize energy consumption and risk in IoT networks. A tiny machine-learning model is applied to help the UAV make real-time decisions for adjusting the locations of virtual nodes. In [8], together with task-offloading decision making in real time, a joint optimization of number and 3D placement of UAV cloudlets was developed to achieve near-optimal performance with low complexity for large-scale IoT networks supporting latency-sensitive services.

More practically, a UAV-assisted service provisioning framework with caching was studied for different real-time levels of services in IoT networks [9]. To achieve high real-time caching performance, a deep Q-network is employed to update the positions of UAV and provide the necessary prerequisite for service caching. Interestingly, the work in [10] proposes a joint multi-UAV path planning and transmission scheduling algorithm for mobile-edge computing in IoT networks with reconfigurable intelligent surfaces. The algorithm is computationally efficient and easily implementable for real-time operation. Aiming to support real-time data updating, collection, and monitoring applications, the authors in [1, 11, 12] studied multi-objective optimization problems for UAV-assisted IoT networks. These studies jointly optimize trajectory planning, data rate, age of information, or/and energy consumption using deep reinforcement learning and real-time optimization algorithms.

It can be observed that the aforementioned studies mainly focus on real-time optimization algorithms rather than on the coordinated deployment of swarm UAV trajectories for delivering the requested contents, i.e., how many UAVs are selected and the trajectory of each UAV. Furthermore, cooperative transmission between the UAVs and the macro base station (MBS) has not been considered to provide the IDUs with high data rates. In our previous work [13], we proposed an energy-aware capacity and trajectory optimization solution for swarm UAV-assisted edge content delivery networks, which maximizes content deliv-

ery capacity and minimizes flight distance to reduce service time¹. However, the role of the MBS was neglected, and thus limiting delivery capacity and failing to reduce the number of UAVs deployed in practical real-time design.

In this paper, we propose a transmission mode selection-based content caching (MSC) integrated with real-time swarm UAV trajectory (RTT) for 6G IoT networks. The MSC-RTT scheme take into account the role of MBS in cooperating with UAVs to transmit the requested contents to the IDUs across different clusters. After the clusters are assigned, each UAV caches the relevant contents and flies to serve its designated clusters in real time. To to so, we formulate an MSC-RTT optimization problem and solve it for (i) the optimal transmission mode selection-based caching that determines whether the MBS or a UAV should serve the IDUs in a cluster and (ii) the optimal number of UAVs and the trajectory of each UAV under their limited caching storage. As a result, the MSC-RTT scheme can provide the IDUs with maximum sum rate while ensuring fast service time for real-time applications and services and conserving the caching storage.

The rest of this paper is organized as follows. In Section 2, we introduce the system model and describe how it works. The system is analyzed to derive all the essential expressions in Section 2. Based on these expressions, Section 4 presents the system design with MSC-RTT optimization problem and its solution with genetic algorithms (GA). Section 5 is dedicated to evaluating the proposed MSC-RTT scheme. Finally, we conclude the paper in Section 6.

2 System Model

We deploy a system of real-time swarm UAV trajectory (RTT) for 6G IoT networks as shown in Fig. 1. The model includes one MBS cooperates with N UAVs to deliver I contents to the IDUs in N clusters. The MBS is located at $(0, 0, h_b)$ and UAV u is originally located at (x_u, y_u, h_u) , $u = 1, 2, \dots, N$. Each cluster n has M_n IDUs of density ρ_c distributed within a circular area of radius R_n , $n = 1, 2, \dots, N$. The center of cluster n is $C_n = (x_n, y_n)$. If UAV u is assigned to serve the IDUs in cluster n , it flies to stop at (C_n, h_n) to transmit the requested content, and then to the next clusters if needed.

In this system, let $s_n \in \{0, 1\}$ be the transmission mode selection index, we aim to find which transmission mode is selected to serve the IDUs in cluster n , i.e., by UAV ($s_n = 1$) or MBS ($s_n = 0$), to maximize the system sum rate. By this way, we obtain a set $\mathcal{L}$ of $L = \sum_{n=1}^N s_n$ clusters served by the UAVs. Simultaneously, we need to know: how to select K out N UAVs ($0 \leq K \leq L$) and how to find the trajectory $\mathcal{T}_k$ of each selected UAV k , $k = 1, 2, \dots, K$, to serve the IDUs with minimum service time for real-time content delivery. By following trajectory $\mathcal{T}_k$, UAV k flies through a set $\mathcal{N}_k$ clusters such that $\mathcal{N}_1 \cup \mathcal{N}_2 \dots \cup \mathcal{N}_K = \mathcal{L}$. Here, we use an operator $\pi(k)$ to map k in K selected UAVs to u in N initial ones. Furthermore, the service time, which is

¹ The service time is defined as the average waiting time of the IDUs in a cluster to be served by a UAV and the MBS.

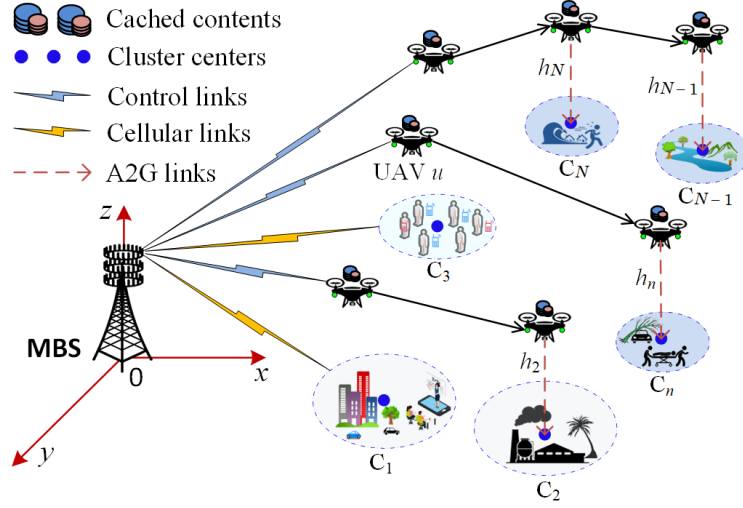


Fig. 1: Swarm UAV-assisted 6G IoT networks.

used for the selected UAVs to serve the IDUs in L clusters, is defined as the longest waiting time of the IDUs in a cluster to receive the first bits transmitted by the UAVs. And thus, it is obvious that we ignore the service time provided by the MBS because the first bits are assumed to be transmitted by the MBS immediately.

Given the aforementioned scenario, the system works as follows. Assume that the MBS acknowledges the content request pattern of the IDUs, it first categorizes them into N clusters and initially selects N UAVs that are ready for deployment. Then, the MBS collects the system information such as channel characteristics, coordinates, contents (size and popularity), and UAV configurations (storage, power, and speed). Next, the MBS formulates the MSC-RTT optimization problem and solves it for the optimal s_n , set of K UAVs, and $\mathcal{T}_k$. Finally, the system is deployed to serve the IDUs in all clusters with maximum sum rate and minimum service time.

3 System Analysis

3.1 Content Requesting and Caching Patterns

The requesting pattern of IDUs over a set of I contents follows a Zipf-like popularity distribution [14], expressed as the probability that content i is requested, given by

$$p_i = \frac{i^{-\alpha}}{\sum_{l=1}^I l^{-\alpha}}, \quad (1)$$

where $\alpha \geq 0$ is the skewed coefficient representing a particular popularity distribution of I contents.

Consider N clusters, each cluster n has M_n IDUs having the same content interest, the number of clusters, which requests content i , is directly proportional to p_i , computed as

$$N_i = \lfloor p_i N \rfloor, \quad (2)$$

where the operator $\lfloor \cdot \rfloor$ is used to round a real value to the nearest integer. This may result in $\sum_{i=1}^I N_i \neq N$ caused by the round operator. An adjustment process to make $\sum_{i=1}^I N_i = N$ is presented in [3].

Let $q_{n,i} \in \{0, 1\}$ be the requesting index to indicate that cluster n requests content i ($q_{n,i} = 1$) or does not ($q_{n,i} = 0$), we have

$$N = \sum_{i=1}^I N_i = \sum_{i=1}^I \sum_{n=1}^N q_{n,i}. \quad (3)$$

So, let $w_{n,i} \in \{0, 1\}$ be the caching index such that the UAVs are assigned to cache content i if $w_{n,i} = 1$, otherwise $w_{n,i} = 0$. The caching index depends on both s_n and $q_{n,i}$, expressed as

$$w_{n,i} = q_{n,i} s_n. \quad (4)$$

3.2 UAV Coverage

In cluster n , the elevation angle θ_n of a UAV is carefully selected depending on the propagation environments, e.g., suburban, urban, dense urban, and high-rise urban, as presented in [15]. Therefore, given an altitude, the proper number of IDUs in cluster n covered by the UAV is

$$M_n = \lfloor \rho_n \pi R_n^2 \rfloor, \quad (5)$$

where $R_n = \frac{h_n}{\tan(\theta_n)}$.

3.3 Wireless Channels

UAV-to-IDU Channel The UAV-to-IDU (U2I) channel is considered as an air-to-ground channel which is defined by probabilistic path loss model [16–19]. In this model, the path loss of line-of-sight (LoS) and none-LoS (NLoS) links between the UAV and IDU m , $m = 1, 2, \dots, M_n$ in cluster n is given by

$$\psi_{n,m}^{\mathcal{A}} = \left(\frac{4\pi f_c d_{n,m}}{c} \right)^2 \eta_n^{\mathcal{A}}, \mathcal{A} \in \{\text{LoS}, \text{NLoS}\}, \quad (6)$$

where f_c is the carrier frequency, c is the speed of light, $\eta_n^{\mathcal{A}}$ is the attenuation factor of LoS link or NLoS link, and $d_{n,m}$ is the distance from the UAV to IDU m in cluster n .

Taking into account the occurrence probabilities of LoS ($P_{n,m}^{\text{LoS}}$) and NLoS ($P_{n,m}^{\text{NLoS}} = 1 - P_{n,m}^{\text{LoS}}$), the average path loss can be derived as

$$\begin{aligned}\bar{\psi}_{n,m} &= \psi_{n,m}^{\text{LoS}} P_{n,m}^{\text{LoS}} + \psi_{n,m}^{\text{NLoS}} P_{n,m}^{\text{NLoS}} \\ &= \left(\frac{4\pi f_c d_{n,m}}{c} \right)^2 [\eta_n^{\text{LoS}} P_{n,m}^{\text{LoS}} + \eta_n^{\text{NLoS}} P_{n,m}^{\text{NLoS}}],\end{aligned}\quad (7)$$

where the occurrence probability of LoS link is expressed as

$$P_{n,m}^{\text{LoS}} = \frac{1}{1 + \kappa_n \exp[-\nu_n(\xi_{n,m} - \kappa_n)]}, \quad (8)$$

where κ_n and ν_n are the constants related to the environment of cluster n [20], $r_{n,m}$ is the distance from center C_n to IDU m in cluster n , and $\xi_{n,m} = \frac{180}{\pi} \arctan(\frac{h_n}{r_{n,m}})$.

MBS-to-IDU Channel When the MBS transmits contents to IDUs in cluster n , the MBS-to-IDU (M2I) channel for IDU m [21] can be expressed as

$$\begin{aligned}H_{n,m}^{\text{M2I}} &= \sqrt{\beta_0 (d_{n,m}^{\text{M2I}})^{-\gamma_0}} \left(\sqrt{\frac{\beta_1}{1 + \beta_1}} h_{n,m}^{\text{M2I,LoS}} + \right. \\ &\quad \left. \sqrt{\frac{1}{1 + \beta_1}} h_{n,m}^{\text{M2I,NLoS}} \right),\end{aligned}\quad (9)$$

β_0 is the M2I channel gain at the reference distance of 1 m, γ_0 and β_1 are the path-loss exponent and Rician factor, $d_{n,m}^{\text{M2I}}$ denotes the distance between the MBS and IDU m in cluster n , $h_{n,m}^{\text{M2I,NLoS}} \sim \mathcal{CN}(0, 1)$, and

$$h_{n,m}^{\text{M2I,LoS}} = e^{-j \frac{2\pi d_{n,m}^{\text{M2I}}}{\lambda}}. \quad (10)$$

3.4 Sum Rate and Service Time

Sum Rate: For the U2I channels, each UAV is assumed to act as a regular cellular user and is allocated a dedicated downlink spectrum resource for transmission. The signal-to-noise ratio (SNR) of the channel used to deliver content i from a UAV to IDU m in cluster n is given by [16]

$$\gamma_{n,m,i}^{\text{U2I}} = \frac{s_n q_{n,i} (\bar{\psi}_{n,m})^{-1} P^{\text{U2I}}}{\sigma^2}, \quad (11)$$

where P^{U2I} denotes the transmission power of the UAV used for delivering content i to IDUs in cluster n . From (11), the transmission rate to deliver content i is given by

$$C_{n,m,i}^{\text{U2I}} = B \log_2(1 + \gamma_{n,m,i}^{\text{U2I}}), \quad (12)$$

where B is the system bandwidth.

So, the average delivery sum rate per content multicasted from the UAVs to the IDUs in all corresponding clusters is given by

$$\bar{C}_{\text{U2I}} = \sum_{n=1}^N \sum_{m=1}^{M_n} \sum_{i=1}^I p_i C_{n,m,i}^{\text{U2I}}. \quad (13)$$

For the M2I channels, the SNR of the channel for delivering content i from the MBS to IDU m in cluster n is given by

$$\gamma_{n,m,i}^{\text{M2I}} = \frac{(1 - s_n) q_{n,i} P^{\text{M2I}} |H_{n,m}^{\text{M2I}}|^2}{\sigma^2}, \quad (14)$$

where P^{M2I} is the power of MBS for transmitting content i to the IDUs in cluster n . From (14), the transmission rate to deliver content i from the MBS is expressed as

$$C_{n,m,i}^{\text{M2I}} = B \log_2(1 + \gamma_{n,m,i}^{\text{M2I}}). \quad (15)$$

Accordingly, the average delivery sum rate per content multicasted from the MBS to the IDUs in all associated clusters is given by

$$\bar{C}_{\text{M2I}} = \sum_{n=1}^N \sum_{m=1}^{M_n} \sum_{i=1}^I p_i C_{n,m,i}^{\text{M2I}}. \quad (16)$$

Finally, the overall average sum rate per cluster achieved by the cooperation between the MBS and UAVs is

$$\bar{C} = \frac{\bar{C}_{\text{U2I}} + \bar{C}_{\text{M2I}}}{N}. \quad (17)$$

Service Time: The service time is defined as the waiting time of the IDUs in a cluster to receive the first bits from the UAVs and MBS. Therefore, the waiting time for IDUs to receive the first bits from the MBS can be neglected, as the first bits are delivered immediately upon request. Let $N_k = |\mathcal{N}_k|$ denote the number of clusters served by UAV k , $k = 1, 2, \dots, K$, and $d(n_k - 1, n_k)$ denote the distance from the stop at cluster $n_k - 1$ to the stop at cluster n_k in trajectory $\mathcal{T}_k$ of UAV k , $n_k = 1, 2, \dots, N_k$, the flight time of UAV k over the distance $d(n_k - 1, n_k)$ is given by

$$T_{n_k}^{\text{FLY}} = \frac{d(n_k - 1, n_k)}{v_k}, \quad (18)$$

where v_k is the speed of UAV k and stop "0" ($n_k = 1$) is the original position of UAV k . It is noted that an operator $\pi(n_k)$ is also applied to map n_k to n such that we obtain $(C_{n_k}, h_{n_k}) = (C_n, h_n)$.

The transmission time of UAV k when it hovers over cluster $n_n \in \mathcal{N}_k$ to transmit content i is given by

$$T_{n_k,i}^{\text{U2I}} = \frac{s_i}{C_{n_k,i}^{\text{U2I}}}, \quad (19)$$

where $C_{n_k,i}^{\text{U2I}} = \min\{C_{n_k,m,i}^{\text{U2I}}, s_{n_k} = 1, \forall m\}$. It means that we choose the worst channel capacity between UAV k and IDUs in cluster n_k to transmit content i . Based on (19), the average service time per cluster is derived as

$$\bar{S}_T = \frac{1}{N} \sum_{k=1}^K \sum_{n_k=1}^{N_k} \left((N_k - n_k + 1) T_{n_k}^{\text{FLY}} + (N_k - n_k) \sum_{i=1}^I T_{n_k,i}^{\text{U2I}} \right). \quad (20)$$

Similarly, we compute the transmission time of MBS to transmit content i to the IDUs in cluster n and its average, for further purpose of performance evaluation, respectively presented as below

$$T_{n,i}^{\text{M2I}} = \frac{s_i}{C_{n,i}^{\text{M2I}}}, \quad (21)$$

and

$$\bar{T}_T = \frac{1}{N} \sum_{n=1}^N \sum_{i=1}^I T_{n,i}^{\text{M2I}}, \quad (22)$$

where $C_{n,i}^{\text{M2I}} = \min\{C_{n,m,i}^{\text{M2I}}, s_n = 0, \forall m\}$.

3.5 Storage Consumption

To ensure the caching efficiency, the maximum caching storage consumption, which depends on not only the content size s_i but also the caching index $w_{n,i}$, is computed as

$$S_C = \sum_{n=1}^N \sum_{i=1}^I w_{n,i} s_i. \quad (23)$$

4 System Optimization Design

The objective of the MSC-RTT optimization problem for swarm UAV-assisted 6G IoT networks is how to maximize the overall average sum rate $\bar{C}$ (17) in a real-time manner, i.e., with minimum service time $\bar{S}_T$, while utilizing the caching storage resource. Solving the MSC-RTT optimization problem, we achieve the optimal solutions of 1) mode selection-based caching index s_n and 2) number of UAVs K and their trajectories $\mathcal{T}_k$. Mathematically, the MSC-RTT optimization problem is formulated as follows

$$\max_{s_n, K, \mathcal{T}_k} \bar{C} - \bar{S}_T, \quad (24a)$$

$$\text{s.t. } S_C \leq S_C^*, \quad (24b)$$

$$0 \leq K \leq L, \quad (24c)$$

where S_C^* is used to limit the caching storage consumption.

By using genetic algorithms (GA) to solve (24), we propose the following scheme. First, we define a binary mask matrix of size $N \times N$, namely $\mathbf{B}_{N \times N}$. This matrix is used to select K UAVs out of N UAVs to fly and which clusters to stop in $\mathcal{T}_k$. By this way, if UAV k is selected to serve the set of clusters $\mathcal{N}_k$, we have $N_k = \sum_{n_k=1}^{N_k} B_{\pi(k), \pi(n_k)} \geq 1$, and by counting the number of UAVs selected, we obtain K . In addition, if UAV k stops to serve cluster $\pi(n_k)$, we have $B_{\pi(k), \pi(n_k)} = 1$, otherwise, $B_{\pi(k), \pi(n_k)} = 0$. Second, we also define an integer matrix of size $N \times N$, namely $\mathbf{I}_{N \times N}$ for obtaining a trajectory matrix given by

$$\mathbf{T}_{N \times N} = \mathbf{B}_{N \times N} \circ \mathbf{I}_{N \times N}, \quad (25)$$

where the operator $\circ$ is use to have $\mathbf{T}_{\pi(k), \pi(n_k)} = \mathbf{B}_{\pi(k), \pi(n_k)} \mathbf{I}_{\pi(k), \pi(n_k)}$, $\forall \pi(k) = 1, 2, \dots, N, \pi(n_k) = 1, 2, \dots, N$.

However, the problems are that 1) there are more than one UAVs flying to the same stop (inter violation, i.e., a column of $\mathbf{B}_{N \times N}$ contains more than one 1-bit), 2) no UAVs stop to serve the IDUs in cluster $\pi(n_k)$ (i.e., a column of $\mathbf{B}_{N \times N}$ contains no 1-bits while $s_n = 1$), 3) a selected UAV does not know to which stop to fly first (intra violation, i.e., a row of $\mathbf{I}_{N \times N}$ has the same integers), and 4) the optimal number of UAVs deployed $1 \leq K \leq L$. To address the first two problems, a punishment is applied if $s_{\pi(n_k)} \sum_{k=1}^K \mathbf{B}_{\pi(k), \pi(n_k)} \neq 1, \forall n_k = 1, 2, \dots, N_k$, leading to a punishment amount of $\sum_{n_k=1}^{N_k} [s_{\pi(n_k)} \sum_{k=1}^K \mathbf{B}_{\pi(k), \pi(n_k)} - 1]^2$. To address the third problem, a much higher punishment amount of distance $d_{\max}$ is directly added to the flight time (18) such that

$$T_{n_k}^{\text{FLY}} = \begin{cases} \frac{d(n_k-1, n_k)}{v_k}, & \text{if } \mathbf{I}_{\pi(k), \pi(n_k-1)} \neq \mathbf{I}_{\pi(k), \pi(n_k)}, \forall k = 1, 2, \dots, K \\ \frac{d_{\max}}{v_k}, & \text{otherwise.} \end{cases} \quad (26)$$

and to address the last problem, we define $b_u = \min\{1, \sum_{n_k=1}^{N_k} \mathbf{B}_{u, \pi(n_k)} s_{\pi(n_k)}\}$ as active UAV k , i.e., UAV k is selected to fly if $b_u = 1$, otherwise, $b_u = 0$, here it is noted that $u = \pi(k)$ as we defined earlier. So, a punishment amount applied to this case is $(\min\{0, \sum_{u=1}^N b_u - 1\})^2 + (\min\{0, L - \sum_{u=1}^N b_u\})^2$ to ensure $1 \leq K = \sum_{u=1}^N b_u \leq L$.

Furthermore, there is one more punishment if (24b) is not satisfied. To address this, we rewrite (24b) in the penalty form as below

$$\Delta S_C = S_C^* - S_C \geq 0, \quad (27)$$

and a punishment amount of this case is $(\min\{0, \Delta S_C\})^2$.

So far, by considering all punishment amounts, the whole punishment amount is given by

$$F = \lambda_1 \sum_{n_k}^{N_k} [s_{\pi(n_k)} \sum_{k=1}^K \mathbf{B}_{\pi(k), \pi(n_k)} - 1]^2 + \lambda_2 [(\min\{0, \sum_{u=1}^N b_u - 1\})^2 + (\min\{0, L - \sum_{u=1}^N b_u\})^2] + \lambda_3 (\min\{0, \Delta S_C\})^2. \quad (28)$$

Finally, instead of solving (24), GA is applied to solve the following unconstrained optimization problem

$$\min_{s_n, \mathbf{B}_{N \times N}, \mathbf{I}_{N \times N}} \bar{C}_F = \bar{C} - \bar{S}_T - F. \quad (29)$$

In conventional GA implementations, an individual is encoded as a single chromosome containing all optimization variables. However, when the variables include different types, a significantly larger population size is required to maintain accuracy and stability. To overcome this issue, the GA with a truncated-string (GTS) scheme is adopted, in which the long chromosome is partitioned into two sub-strings: (i) the first encodes all binary variables for s_n and $\mathbf{B}_{N \times N}$, and (ii) the second encodes the integer variable $\mathbf{I}_{N \times N}$. Due to space limitations, the GTS implementation is omitted, the complete procedure and its advantages can be found in [3].

5 Performance Evaluation

5.1 Parameters Setting

In this paper, the main parameters of system and GA are shown in Table 1. The propagation features κ_n , ν_n , η_n^{LoS} , η_n^{NLoS} , and θ_n are randomly selected in accordance with different environments as detailed in [15, 20]. The requesting index $q_{n,i}$, which is generated with respect to (w.r.t) α , is presented in [3]. In addition, the flight power of all UAVs, which follows the voltage dropping across and the current flowing through the UAV's electric motors [22], is set to 1000 W (i.e., equivalent to a flight speed v_n of 23.4 m/s, $\forall n = 1, 2, \dots, N$). GA terminates when either $N_G = 100$ generations are reached, or when both the change in the fitness function $\bar{C}_F$ between two generations and the penalty function F remain below 10^{-3} for 10 successive generations.

5.2 GA Evaluation

The convergence behavior of the GA is demonstrated in Fig. 2. Over 100 generations, the algorithm converges at approximately the 65th generation. At convergence, the mean fitness value of all individuals in the population approaches the best one. In addition, the penalty value becomes zero, indicating that all individuals satisfy the constraints without any punishments. This result confirms that the GA achieves reliable and feasible convergence for solving the MSC-RTT optimization problem.

Table 1: Parameters Setting.

Symbols	Values
N	10 clusters/ UAVs
I	10 videos
$(0, 0, h_b)$	(0, 0, 25)m
(x_u, y_u)	Random uniform distribution in the range of [0, 400]m
h_u	Random uniform distribution in the range of [5, 50]m
(x_n, y_n)	Random uniform distribution in the range of [-200, 200]m
h_n	Fixed to 50 m for all clusters
R_n	Random uniform distribution in the range of [10, 20]m
ρ_n	Random uniform distribution in the range of [0.01, 0.1] IDUs/m ²
s_i	Random uniform distribution in the range of [100, 500]Mbit
S_C^*	$0.5 \times \sum_{i=1}^I s_i$
α	1
$\{c, f_c, B\}$	$\{3 \times 10^8 \text{ m/s}, 2 \text{ Ghz}, 10 \text{ Mhz}\}$
$\{\beta_0, \beta_1, \gamma_0\}$	$\{10^{-3}, 6, 4\}$
$\{\sigma^2, P^{U2I}, P^{M2I}\}$	$\{10^{-13} \text{ W}, 2.5 \text{ W}, 5 \text{ W}\}$
$d_{\max}$	1000 m
N_P	5000 individuals (population size)
N_G	100 generations
P_G	0.9 (generation gap)
P_{cb}	0.9 (crossover probability for s_n and $\mathbf{B}_{N \times N}$ without mutation)
P_{ci}	0.9 (crossover probability for $\mathbf{I}_{N \times N}$ without mutation)
$\{\lambda_1, \lambda_2, \lambda_3\}$	$\{10^{-1}, 10^4, 10^1\}$

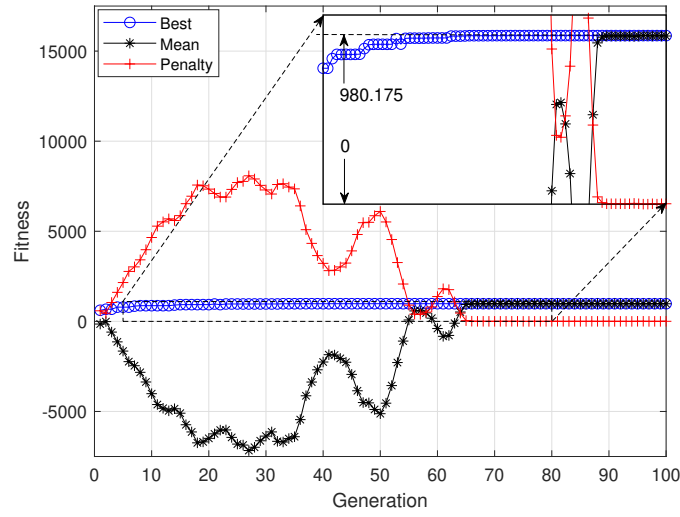
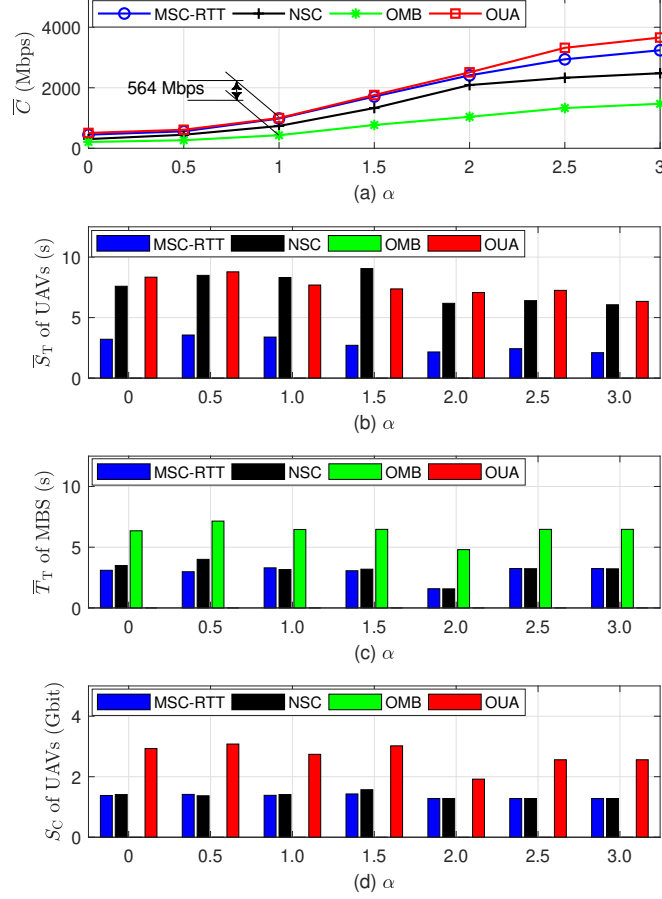
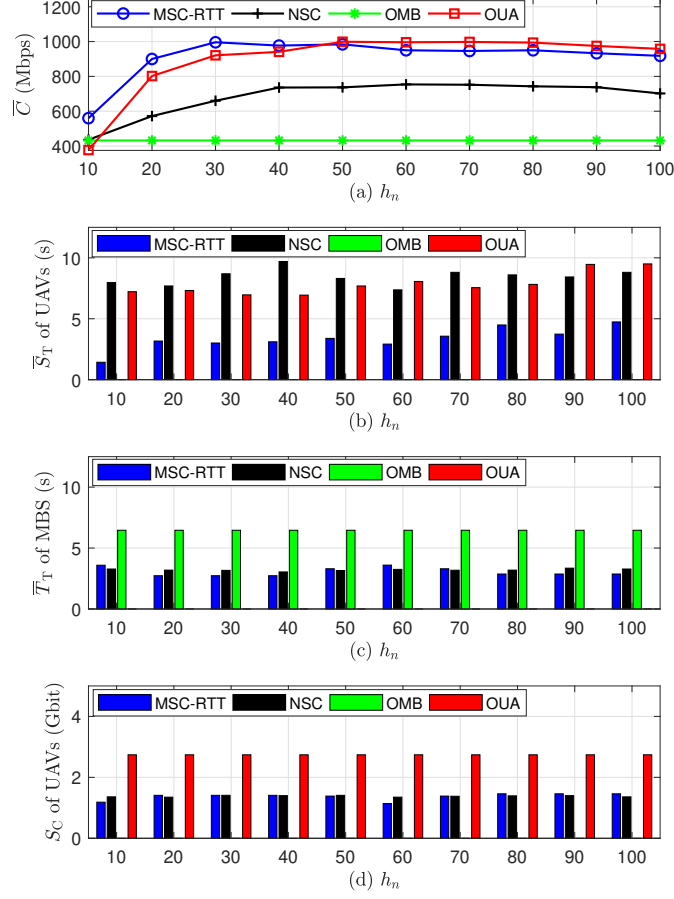


Fig. 2: GA convergence rate.

Fig. 3: Performance versus α .

5.3 MSC-RTT Evaluation

To evaluate the performance of the proposed MSC-RTT scheme, we compare it with three baseline methods: (i) no mode-selection caching (NSC), (ii) only UAVs transmission (OUA), and (iii) only MBS transmission (OMB). In the NSC scheme, the sub-string encoding s_n is initialized randomly and remains unchanged during the crossover and mutation operators, whereas the sub-strings encoding $\mathbf{B}_{N \times N}$ and $\mathbf{I}_{N \times N}$ are still processed by these genetic operators. After completing N_G generations, the final results are obtained by averaging $\bar{C}$, $\bar{S}_T$, $\bar{T}_T$, and S_C over all individuals that incur no punishments. In the OUA scheme, it follows the same implementation as the proposed MSC-RTT scheme, except

Fig. 4: Performance versus h_n .

that s_n is forced to 1-bits for all individuals. And in the OMB scheme, the GA does not need to be executed because no optimization variables are involved when $s_n = 0$.

The performance of all schemes are evaluated w.r.t α as shown in Fig. 3. In particular, MSC-RTT clearly outperforms the other three schemes because it can offer the best trade-off among high sum rate ($\bar{C}$), low service time ($\bar{S}_T$), and low caching storage (S_C) for high-data rate and real-time applications and services. OUA provides a slightly higher sum rate than MSC-RTT but suffers from much longer service time and higher caching storage consumption. OMB achieves the best performance in terms of service time and caching storage, i.e., obviously without waiting time and no caching storage consumed by UAVs, but the sum

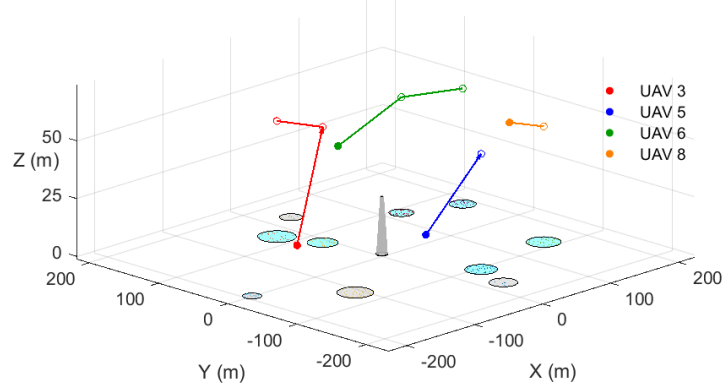


Fig. 5: Optimal trajectories with 4 UAVs flying to serve the IDUs in 6 clusters.

Table 2: Performance versus number of UAVs.

No. of UAVs	1	2	3	4	5	6	7	8	9	10
$\bar{C}$ (Mbps)	874	970	970	984	970	984	1002	1004	1018	999
$\bar{S}_T$ (s)	3.427	6.412	3.498	3.387	3.530	4.313	8.089	6.507	11.602	12.050
$\bar{T}_T$ (s)	4.248	3.596	3.596	3.305	3.596	3.305	2.120	1.020	0.729	0.000
$\bar{S}_C$ (Gbit)	0.840	1.140	1.140	1.385	1.140	1.385	1.675	2.175	2.420	2.740

rate performance is extremely poor. NSC is the worst scheme with moderate sum rate, transmission time, and caching storage, but too high service time. In general, the sum rate of all schemes increases with α , as a higher α prioritizes serving the IDUs requesting more popular contents.

The performance trends of all schemes with respect to h_n in Fig. 4 are similar to those with respect to α , except that an optimal h_n can be identified, as shown in Fig. 4(a). Specifically, the proposed MSC-RTT achieves its maximum sum rate at $h_n = 30$ m, whereas OUA and NSC reach their maxima at $h_n = 50$ m and $h_n = 60$ m, respectively. Notably, no optimal h_n is observed for the OMB scheme. These observations facilitate the design of more effective UAV trajectories to provide IDUs with higher sum rates.

Importantly, the results in Table 2 illustrate the performance of MSC-RTT for varying numbers of UAVs. It can be observed that deploying 4 UAVs achieves optimal performance in terms of providing the IDUs with high data rates, supporting real-time applications and services, and consuming low cost of storage resource. Deploying either more or fewer UAVs does not yield further improvement. The detailed trajectories of 4 UAVs are plotted in Fig. 5. It is noted that, according to the model of UAV's electric motors in [22], the flight power

of a UAV can increase to nearly 2000 W when accelerating to a speed of 30 m/s. Therefore, the flight power can be flexibly adjusted to meet the real-time requirements of different applications and services.

6 Conclusion

In this paper, we have developed a transmission mode selection-based content caching (MSC) scheme integrated with real-time UAV trajectory (RTT) optimization to address the challenges of high data rate and low latency delivery in 6G IoT networks. By jointly optimizing the transmission mode selection-based caching and UAV trajectories, the MSC-RTT framework efficiently provides the IoT devices and users with high system sum rate while maintaining low service time and low caching storage consumption. Simulation results confirm that the proposed approach consistently outperforms conventional schemes, highlighting its effectiveness for high data rate and real-time content delivery and demonstrating the potential of cooperative UAV-MBS operations. The findings also provide valuable insights in designing real-time swarm UAV-assisted 6G IoT networks for advanced applications and services.

Acknowledgement

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.04-2023.25.

References

1. D. V. Huynh, T. Do-Duy, L. D. Nguyen, M.-T. Le, N.-S. Vo, and T. Q. Duong, "Real-time optimized path planning and energy consumption for data collection in unmanned ariel vehicles-aided intelligent wireless sensing," *IEEE Trans. Industrial Informatics*, vol. 18, pp. 2753–2761, Apr. 2022.
2. L. Agrawal and A. Mondal, "Energy-efficient bandwidth orchestration for industrial IoT in softwarized 6G networks," *IEEE Internet of Things J.*, vol. 12, pp. 40440–40448, Oct. 2025.
3. T. C. Lam, N.-S. Vo, M.-P. Bui, C. D. T. Thai, H. Jung, and V.-C. Phan, "Service time-aware caching, power allocation, and 3D trajectory optimised multimedia content delivery in UAV-assisted IoT networks," *IEEE Trans. Veh. Technol.*, vol. 74, pp. 6419–6432, Apr. 2025.
4. L. D. Nguyen, A. Kortun, and T. Q. Duong, "An introduction of real-time embedded optimisation programming for UAV systems under disaster communication," *EAI Endorsed Trans. Ind. Netw. and Intell. Syst.*, vol. 5, pp. 1–8, Dec. 2018.
5. D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and D. Niyato, "6G Internet of things: A comprehensive survey," *IEEE Internet of Things J.*, vol. 9, pp. 359–383, Jan. 2022.
6. J. Xu, H. Yao, R. Zhang, T. Mai, and M. Guizani, "Low latency and accuracy-guaranteed DNN inference for UAV-assisted IoT networks," *IEEE Trans. Cognitive Commun. and Networking*, pp. 1–18, Early Access, Feb. 2025.

7. R. Liu, M. Xie, A. Liu, and H. Song, "Joint optimization risk factor and energy consumption in IoT networks with tinyML-enabled Internet of UAVs," *IEEE Internet of Things J.*, vol. 11, pp. 20983–20994, June 2024.
8. R. Islambouli and S. Sharafeddine, "Optimized 3D deployment of uav-mounted cloudlets to support latency-sensitive services in IoT networks," *IEEE Access*, vol. 7, pp. 172860–172870, Nov. 2019.
9. H. Yan, H. Li, X. Xu, and M. Bilal, "UAV-enhanced service caching for IoT systems in extreme environments," *IEEE Internet of Things J.*, vol. 11, pp. 26741–26750, Aug. 2024.
10. A. V. Savkin, C. Huang, and W. Ni, "Joint multi-UAV path planning and LoS communication for mobile-edge computing in IoT networks with RISs," *IEEE Internet of Things J.*, vol. 10, pp. 2720–2727, Feb. 2023.
11. J. Pan, Y. Li, R. Chai, S. Xia, and L. Zuo, "Multiobjective trajectory planning for UAV-assisted IoT networks based on DRL approach," *IEEE Internet of Things J.*, vol. 12, pp. 15840–15852, June 2025.
12. Y. Hu, Y. Liu, A. Kaushik, C. Masouros, and J. S. Thompson, "Timely data collection for UAV-based IoT networks: A deep reinforcement learning approach," *IEEE Sensors Journal*, vol. 2, pp. 12295–12308, June 2023.
13. T. C. Lam, N.-S. Vo, M.-P. Bui, D.-T. Huynh, T.-D. Tran, and H. D. Hung, "Genetic algorithms for swarm UAV-assisted edge content delivery optimisation problem," in *IEEE Inter. Conf. Research, Innovation and Vision for the Future in Computing and Commun. Technol.*, (Danang City, Vietnam), pp. 188–193, Dec. 2024.
14. L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Proc. of IEEE Inter. Conf. on Computer Commun. (INFOCOM)*, (New York, NY), pp. 126–134, Mar. 1999.
15. M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements," *IEEE Commun. Lett.*, vol. 7, p. 38–41, Feb. 2018.
16. W. Saad, M. Bennis, M. Mozaffari, and X. Lin, "Chapter 7 in *Wireless Communications and Networking for Unmanned Aerial Vehicles*". Cambridge University Press, 1st ed., 2020.
17. A. Hourani, S. Kandeepan, and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," in *Proc. IEEE Global Commun. Conf.*, (Austin, TX), pp. 1–7, Dec. 2014.
18. ITU-R, "Rec. p.1410-2 propagation data and prediction methods for the design of terrestrial broadband millimetric radio access systems," *Series, Radiowave propagation*, 2003.
19. M.-H. T. Nguyen *et al.*, "Spectrum-sharing UAV-assisted mission-critical communication: Learning-aided real-time optimisation," *IEEE Access*, vol. 9, pp. 11622–11632, Jan. 2021.
20. R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," in *Proc. IEEE Inter. Commun. Conf.*, (Kuala Lumpur, Malaysia), pp. 1–5, May 2016.
21. S. Zargari, A. Hakimi, C. Tellambura, and S. Herath, "User scheduling and trajectory optimization for energy-efficient IRS-UAV networks with SWIPT," *IEEE Trans. Veh. Technol.*, vol. 72, pp. 1815–1830, Feb. 2023.
22. B. Li *et al.*, "3D trajectory optimization for energy-efficient UAV communication: A control design perspective," *IEEE Trans. Wireless Commun.*, vol. 21, pp. 4579–4593, June 2022.