Towards a UAV-centric Content Caching Architecture for Communication-challenged Environments

Amit Kumar Bhuyan, Hrishikesh Dutta, and Subir Biswas
Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA
bhuyanam@msu.edu, duttahr1@msu.edu, sbiswas@msu.edu

***Abstract*— This article presents an unmanned aerial vehicle (UAV) based caching framework for content provisioning in disaster scenarios. In a disaster scenario without the availability of static base stations and their wireline backhauls, community of stranded users can access contents from a network of static and traveling UAVs. A set of relatively static *anchor UAVs* with vertical as well as lateral links provide content access to its local users. A set of *ferrying UAVs* with only lateral links, but with wider mobility, can also provision content to users while visiting different communities of users. The algorithmic objective is to intelligently cache contents in the storage-constrained UAVs in order to maximize content availability for the users affected by such disasters. The paper develops a novel approach of content duplication within the anchor UAVs along with a mechanism to distribute non-duplicated contents across the ferrying UAVs. Through analytical modeling and simulation experiments it is demonstrated that content availability in such an arrangement can be maximized by choosing an optimal level of duplication for content with specific popularity distributions. The paper does functional verification and performance evaluation of the proposed caching framework under a wide range of network size, UAV distribution, content popularity, and ferrying UAV trajectories.**

***Keywords*— Disasters, Unmanned Aerial Vehicles, Zipf Distribution, Caching, Content Popularity, Content Availability**

I. Introduction

Major communication challenges are usually caused by natural disasters such as tornadoes, cyclones, flood, and forest fire, which can cut off and isolate communities of users by destroying cellular network infrastructure. Similar challenges can also be caused by human-triggered calamities such as wars and conflicts, which can destroy infrastructure and move refugees to areas without communication infrastructure. Being able to provide content access to users in such challenged scenarios can be crucial, often lifesaving. Content related to relief information, state and geography of disaster/conflict dynamics, news, weather, healthcare, etc. can be lifesaving for the affected communities. In the absence of fully operational fixed communication infrastructure, Unmanned Aerial Vehicles (UAVs) can be used as an alternative provisioning platform for such content. While UAVs are leveraged for their mobility, they bring specific constraints due to their limited storage, energy, and the subsequent flight time after each recharge.

This paper develops a storage-constrained UAV-based caching and trajectory design framework for content dissemination in communication-challenged systems. The framework aims towards generalized scenarios in which a disaster/war-stricken population is forced to cluster into multiple communities. Each such community is served by an anchor-UAV (i.e., A-UAV) with expensive *vertical connectivity* such as a satellite link. There are a set of ferrying-UAVs (i.e., F-UAVs) without any vertical connectivity, whose primary function is to ferry and distribute content across the A-UAVs. The objective is to provide high availability content access to all the communities, and that is without incurring the cost of excessive vertical link usage by the A-UAVs. The paper sets out to answer the question: for a given set of A-UAVs and their storage limits, content demand model, content request distribution, F-UAV fleet size and available storage, what are the optimal content download and caching policies for both A-UAVs and F-UAVs. Also, what are the impacts of different F-UAV fleet trajectories. Both these questions would be addressed with the goal of maximizing content availability across all partitioned communities of users.

Many works have proposed the use UAVs to form Ad Hoc networks that can fill the communication gaps caused due to infrastructure destruction [5-13]. While these approaches are useful for scenarios with partial destruction, the communication services provided by such Ad Hoc networks do not work well when all communication towers are destroyed, and a brand-new network needs to be formed in a remote place. Delay Tolerant Networks (DTN) of UAVs have also been proposed for content ferrying across disconnected communities situated in disaster scenarios [3, 4]. Majority of these approaches solve the problem of low delay DTN routing across communities that become partitioned due to communication infrastructure destruction. While solving situation-specific routing problems, these works do not address the caching problem and the impacts of UAV trajectory on caching.

These are the main focus in this paper with the following scope and contributions. First, a detailed UAV-enabled content dissemination architecture model is developed for scenarios without fixed communication infrastructure. Second, optimal content placement and caching strategies are developed for given UAV trajectories. Third, the impacts of UAV trajectories on caching strategies are investigated. Fourth, analytical models are developed for content availability estimations under the developed strategies. Finally, a detailed simulation experiment model using a time driven simulator is developed for functional validation and performance evaluation of the developed strategies under various network, loading, and content popularity scenarios.

II. Related Work

UAVs have been extensively studied in the literature as a vehicle for content provisioning in infrastructure-challenged environments. The work in [5] proposes flight trajectory optimization, communication scheduling, service coverage extension using optimized UAV hovering time and multi-hop relaying through multiple UAVs. The authors in [6] target similar objectives for IoT networks using multi-antenna transceiver design and multi-hop device-to-device (D2D) routing for coverage extension for energy-constrained UAVs. While addressing coverage extension solutions, these works do not deal with content placement and caching issues, which are central to our work in this paper.

The problem of content placement and caching are handled in [7-9]. The paper in [7] proposes a way of using named data networking (NDN) architecture in IoT networks, in which

978-1-6654-3540-6/22/$31.00 ©2022 IEEE

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001616

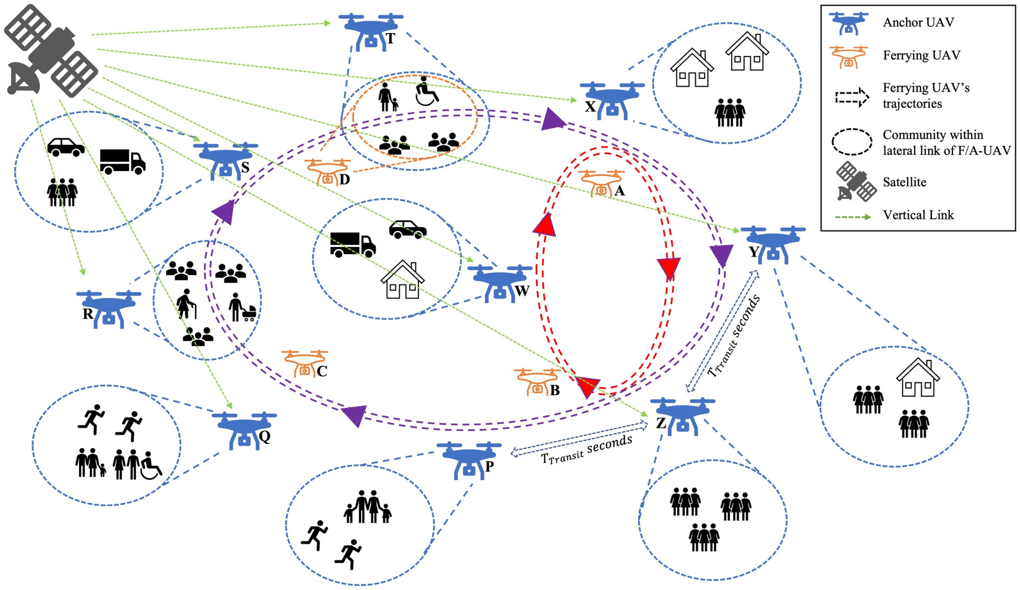


Fig. 1. Coordinated UAV system for content caching and distribution in environments without communication infrastructure

UAVs collect data from the IoT field and deliver to interested recipients to avoid retransmission. The paper in [8] explores UAV-enabled communication with caching at the ground nodes (GNs). The UAVs pro-actively transmits content to an algorithmically selected subset of GNs that cooperatively cache all the required contents. The paper in [9] proposes a probabilistic cache placement technique in order to maximize cache hit probabilities in networks in which wireless nodes are placed using a homogeneous Poisson Point Process (PPP). The work in [10] primarily focuses on the security and denial of service attacks while using UAVs for communication. These mechanisms do not consider impacts of storage space and UAV trajectory design, thus making them not suitable for the problem addressed in this paper.

Joint caching and trajectory decision techniques for UAVs are proposed in [11]. Online decisions are made by a Convolution Neural Network (CNN) based system with images as inputs, while the caching and trajectory optimization is done offline using a clustering-based two layered algorithm (CBTL) paradigm to produce labels. The paper in [12] develops a technique to minimize content delivery delay by joint optimization of UAV trajectory and radio resource allocation. A deep Q-learning based approach is used for such optimization in large networks with exploding state-action pairs. [11, 12] deal with issues that are very similar to ours. The results from these papers will be compared with results from the next step of our work in which online decision making will be incorporated.

III. System Model

A. UAV Hierarchy

The content distribution system is organized in two layers, namely, the anchor UAVs (i.e., A-UAVs) and the ferrying UAVs (i.e., F-UAVs). As shown in Fig. 1, each partitioned community of users is served by an A-UAV using a lateral wireless link such as WiFi. A-UAVs can also download content form the internet via an expensive vertical link such as satellite-based internet. One monolithic system design approach is to let the A-UAVs download all needed content, as requested by their local users, via the vertical links. In this approach, with no inter-A-UAV data transfer, the following shortcomings will be encountered. First, there will be duplications of downloads via the expensive vertical links by different A-UAVs due to overlaps in requests for popular contents. This will incur high download costs. Second, storage constraints will cap the number of contents that can be downloaded and stored in an A-UAV, thus limiting the content availability. Finally, due to limited infrastructure availability, some of the communities of users are rendered isolated from content access without a dedicated A-UAVs assigned to them.

To address these problems, a set of ferrying UAVs (i.e., F-UAVs) are introduced. Unlike A-UAVs, the F-UAVs do not possess vertical links, but they do have lateral links such as WiFi, using which they can communicate with the A-UAVs. The role of these UAVs is to cache and transfer content around the A-UAVs such that the users in a community are able to access content that was downloaded by A-UAVs serving other communities via F-UAVs.

B. Content Request and Provisioning Model

Content requests are generated by the community users and sent to the local A-UAV or a visiting F-UAV, in that order.

Content Popularity and Requests: Studies have shown that content request pattern often follows a Zipf distribution in which a requested content's popularity is a geometric multiple of the next popular content in a larger pool [1]. Popularity of content $'i'$ is given as $p_\alpha(i) = (1/i)^\alpha / \sum_{c \in C} (1/c)^\alpha$. The parameter C represents the total number of contents in the pool, and the Zipf parameter α determines the skewness of the distribution. Poisson request generation is the most prevalent way to capture real-time user requests.

Tolerable Access Delay and Content Provisioning: For each generated request, a Tolerable Access Delay (TAD) is specified. TAD is a quality-of-service parameter that indicates the duration a requesting user waits before the content is provisioned via download. After receiving a request from one of its community users, the relevant A-UAV first searches its local storage for the content. If not found, it waits for a potential future delivery of the content by one of the traveling F-UAVs.

If no F-UAV with that content arrives within the specified TAD, the A-UAV downloads it via the vertical link.

IV. CACHING POLICIES

The caching related design questions to be addressed are: a) which content to be downloaded in the A-UAVs via the vertical links so that they can serve their own community directly, and the remote communities via the traveling F-UAVs; b) which content to be transferred from the A-UAVs to the F-UAVs via the lateral links, and cached within the F-UAVs subsequently; and finally, c) what inter-community trajectories should be followed by the F-AUVs.

This paper addresses these questions in that it assumes pre-assigned globally known content popularities and static content pre-placements before user request are generated. In terms of F-UAV trajectories, different pre-programmed trajectories are characterized along with different static content placement strategies. After understating and characterizing such static policies, the goal will be to develop runtime and dynamic mechanisms for all these design components and report it in a future publication.

A. Caching at Anchor UAVs (A-UAVs)

A naïve strategy for the A-UAVs would be to cache the most popular contents (i.e., following the globally known Zipf distribution) to fill out their individual storage space of C_A contents. This naïve fully duplicated (FD) [2, 14] mechanism has the shortcoming in that it limits the number of accessible contents for all user communities to C_A, the A-UAV cache size. This limitation can be addressed by storing a certain number of unique (exclusive) contents in all the A-UAVs and share those contents across the communities via the traveling F-AUVs. This *Smart Cache Duplication* (SCD) mechanism can effectively increase the access to the number of contents for all users across the entire system, thus improving the overall availability within a given TAD.

Let the size of the duplicate segment of A-UAV cache be $\lambda.C_A$ and that of the unique segment be $(1-\lambda).C_A$where λ is a *duplication factor* that decides the level of content duplication in A-UAVs. This results into $N_A.(1-\lambda).C_A$ unique contents stored across all N_A number of A-UAVs in the system, and these can be shared across all user communities via the mobile F-UAVs. These unique contents have popularities after the top $\lambda.C_A$ popular duplicated contents in all the A-UAVs. For symmetry, all $N_A.(1-\lambda).C_A$ unique contents are uniformly randomly distributed across N_A number of A-UAVs. The total number of contents in system: $C_{sys}=\lambda.C_A+N_A.(1-\lambda).C_A$.

It should be noted that with λ set to one, the SCD system reduces to the fully duplicated (FD) strategy. With higher λ values, the users have better access to more number of highly popular contents, but to fewer of them with low popularity, that are stored across the system-wide A-UAVs and can be accessed via the mobile F-UAVs. A lower λ creates an opposite effect. The goal is to be able to choose a λ, that strikes the right balance between those effects and maximizes the overall availability.

B. Caching at Ferrying UAVs (F-UAVs)

The purpose of the F-UAVs is to ferry around $N_A.(1-\lambda).C_A$ unique contents stored in all N_A A-UAVs. In the presence of limited per-F-UAV caching space, C_F, its caching policy can be determined based on its trajectories, the value of λ, and the Zipf parameter defining the content popularity.

Consider a situation in which an F-UAV k is approaching towards the A-UAV i. Let U_i be the set of all unique contents in the entire system except the ones stored in A-UAV i. To maximize content availability for the users in A-UAV *i's* community, the F-UAV should carry as many low popularity contents from set U_i as its cache space permits. To enable such access, F-UAV k should carry C_F top popular contents from the set U_i while approaching A-UAV i. The size of the set U_i can be expressed as $|U_i|=(N_A-1).(1-\lambda).C_A$. In scenarios when $C_F\le|U_i|$, the F-UAV should carry the C_F top popular contents as outlined above. Otherwise, the F-UAV will carry all $|U_i|$ unique contents, leaving part of the F-UAV cache (i.e., $C_F-|U_i|$) empty. This causes underutilization of F-UAV cache space due to large λ values, leading to heavy in-A-UAV duplications, thus storing few unique contents.

C. Trajectory of Ferring UAVs

An F-UAV's trajectory is represented by the *sequence* of visited A-UAVs, and the *hovering duration* at each A-UAV. Trajectory sequence can be categorized as *partitioned* or *global* cycles. With a partitioned trajectory cycle, an F-UAV go around a specific part of the system containing a fixed subset of all the A-UAVs like F-AUVs *A* and *B* follow a partitioned cycle of A-UAVs *X, Y, Z* and *W* in Fig. 1. With a global cycle, an F-UAV moves around all the A-UAVs in the system like F-UAVs *C* and *D* in Fig. 1. Intuitively, if the contents cached in the unique segments of A-UAVs have very low popularity then the global sequence cycle would be beneficial. Conversely, when some of the A-UAVs maintain unique contents with comparatively very high popularity, then using partitioned cycle may be rewarding. These will be evaluated in the experiment in Section VI.

The cycle time of an F-UAV trajectory is $T_{cycle}=N_A^C\times(T_{Hover}+T_{Transit})$, where N_A^C is the number of A-UAVs in the cycle (partitioned or global), T_{Hover} is the hover duration at each A-UAV, and $T_{Transit}$ is the transit time between two consecutive A-UAVs in a sequence. $T_{Transit}$ depends on the F-UAV flying speed, inter-A-UAV distance, wind speed/directions, and other environmental factors. T_{Hover} should be set to a value which is determined by the data transfer rate and the amount of data needs to be exchanged between F-UAV to/from A-UAV. It should be noted that A-UAVs don't follow a trajectory since they are stationed at their respective communities for uninterrupted content dissemination.

V. CONTENT DISSEMINATON PERFORMANCE

A. Content Availability

Availability is defined as the probability of finding a requested content within the local A-UAV or a future visiting F-UAV within a TAD. Consider a situation in which a single F-UAV cycles in a round-robin manner through all the A-UAVs with hovering and transit respectively. For a content requested from a community, the F-UAV may or may not be accessible within the specified TAD. This probability is as follows:

$$P_{FA}=\begin{cases}\frac{N_F\times(T_{Hover}+TAD)}{N_A\times(T_{Hover}+T_{Transit})} & for\ TAD<((\frac{N_A}{N_F}-1)T_{Hover}+\frac{N_A}{N_F}T_{Transit})\\ 1 & for\ TAD\ge((\frac{N_A}{N_F}-1)T_{Hover}+\frac{N_A}{N_F}T_{Transit})\end{cases}\quad(1)$$

If the TAD is larger than a specific duration, then the F-UAV's accessibility to the requesting community is guaranteed.

Otherwise, it follows the first expression in Eqn. 1. Note that the physical accessibility to the F-UAV does not guarantee the access to the requested content since the F-UAV can store only a limited number (i.e., C_F) of unique contents. Let P_F be the probability that the requested content can be found within the F-UAV following a caching strategy as stated in Section IV. It can be expressed as:

$$P_F = \sum_{i=C_A+1}^{C_A+1+C_{EFF}} p_\alpha(i) \tag{2}$$

where, $p_\alpha(i)$ is the Zipf distributed popularity as defined in Section III. The effective cache size of the F-UAV is given as: $C_{EFF} = min\{C_F, (N_A - 1) \times (1 - \lambda) \times C_A\}$. Effective cache size is less than C_F when F-UAVs cache is partly empty i.e., underutilized (see Section IV). Now, the probability that requested content can be found within a A-UAV that is local to the request generating community can be expressed as:

$$P_A = \sum_{i=1}^{\lambda \times C_A + (1-\lambda) \times C_A} p_\alpha(i) \tag{3}$$

Combining those three probabilities above, the overall availability can be stated as:

$$P_{Avail} = P_A + P_{FA} \times P_F \tag{4}$$

To summarize, local contents from A-UAVs (i.e., both duplicate and unique) and unique contents from future visiting F-UAVs contribute towards the overall availability P_{Avail} within a specified TAD. Note that all unavailable contents within the specified TAD will have to be downloaded by the A-UAVs using their expensive vertical links such as the satellite Internet. Therefore, availability indirectly indicates the content download cost in the system.

B. Low Avilability Period

Consider the scenario in Fig. 2 with two A-UAVs and one F-UAV. The users in a community have access to the content in the F-UAV for a duration of $TAD + T_{Hover}$. Time taken for the F-UAV to come back to the same community before the users in the community will have access to its content again is: $2.T_{Transit} + T_{Hover} - TAD$. This is the period during which the content availability for the users will only be from the local A-UAV, and that is without access to the F-UAV.

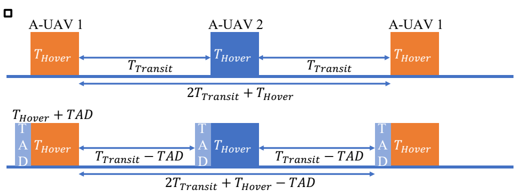


Fig. 2. (Top) Scenario with $TAD = 0$; (Bottom) With non-zero TAD

This duration is referred to as the *low availability period,* which can be generally expressed as:

$$LAP = \frac{N_A T_{Transit} + (N_A - 1) T_{Hover} - TAD}{N_F} \tag{5}$$

where N_A and N_F are the number of A-UAVs and F-UAVs in the system. With higher transit and hovering times and N_A, while the *low availability period* goes up, the overall availability, as derived in Eqns. 1 through 4, goes down.

C. Content Access Delay

Any request that is served by a local A-UAV experience zero access delay. There is no access delay if the request for content from F-UAV is generated when the F-UAV is hovering in the community. Therefore, the only scenario with a non-zero access delay would be the one in which the requested content is available at an F-UAV, and it is currently not visiting the requesting community. The probability of that scenario P_{CAD} can be expressed as:

$$P_{CAD} = \begin{cases} \frac{N_F \times TAD}{N_A \times (T_{Hover} + T_{Transit})} & for\ TAD < \frac{T_{cycle}}{N_F} - T_{Hover} \\ \frac{N_F \times \left(\frac{T_{cycle}}{N_F} - T_{Hover}\right)}{N_A \times (T_{Hover} + T_{Transit})} & for\ TAD \geq \frac{T_{cycle}}{N_F} - T_{Hover} \end{cases} \tag{6}$$

Note that the access delay is upper bounded by the specified TAD. As per the second expression in Eqn. 6, if the TAD is larger than the time it takes for the F-UAV to reach the request generating community, then content is delayed by the time taken by the F-UAV to reach. Conversely, for lower $TADs$, the content is delayed just by the TAD duration.

The average delay incurred in those two cases are:

$$Delay_{av} = \begin{cases} \frac{TAD}{2} & for\ TAD < \frac{T_{cycle}}{N_F} - T_{Hover} \\ \frac{\frac{T_{cycle}}{N_F} - X}{2} & for\ TAD \geq \frac{T_{cycle}}{N_F} - T_{Hover} \end{cases} \tag{7}$$

These averages are based on the maximum and the minimum possible delays. Combining P_{CAD} and $Delay_{av}$, the access delay (AD) can be expressed as:

$$AD = P_{CAD} \times \sum_{i=\forall C_F} p_\alpha(i) \times Delay_{av} \tag{8}$$

VI. EXPERIMENTAL RESULTS AND ANALYSIS

Experiments were carried out using time driven simulation kernel, added for implementing the request generation, UAV caching, and F-UAV movement strategies presented in Sections IV. Default experimental parameters are $N_C = 1000, N_A = 20, N_F = 10, C_A = C_F = 50, \mu = 1, T_{Hover} = 20\ secs, T_{Transit} = 10\ secs, TAD = 20\ secs\ and\ \alpha = 1.001$.

A. Impacts of F-UAVs on Content Availability

Fig. 3 depicts the benefits of the ferrying UAVs in terms of improving content availability as defined in Section IV. The figures show availability computed analytically and from simulation experiments (i.e., average computed from the success of 10^5 requests for each availability point), both of which are validated through their excellent agreements.

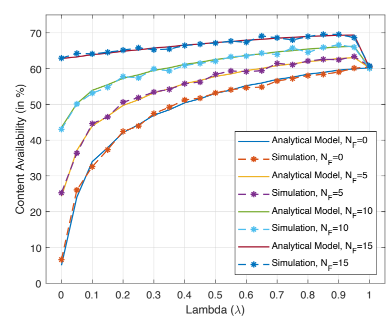


Fig. 3 Content Availability with changing λ for different N_F

Content availability is evaluated for varying λ values, representing the split between cached duplicated and unique objects within the A-UAVs, as described in Section IV. The following observations can be made from Fig. 3. First, increasing F-UAVs can improve availability by ferrying contents that are not otherwise available to a community in its local A-UAV's cache. Second, the percentage increase in availability is more drastic for lower values of λ for which more unique contents are cached in the A-UAVs. Since the F-UAVs ferry around those unique contents across different communities, the dependance of availability on cached contents

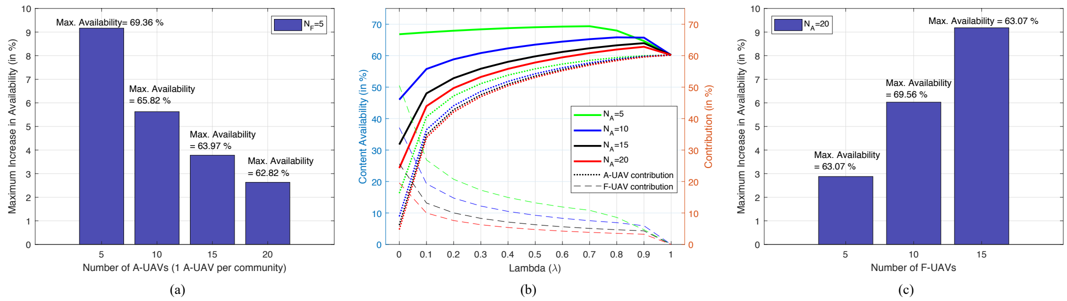


Fig. 4 (a) Maximum increase in Availability with A-UAVs; (b) Contribution % from A- and F-UAVs; (c) Maximum increase in Availability with F-UAVs

in the F-UAVs is more pronounced for smaller λ. Third, there is an optimum duplication factor λ, for which the content availability is the maximum for a given number of A-UAVs, F-UAVs, and default system parameters. Beyond the optimal operating point, availability reduces due to cache underutilization in F-UAVs, as shown in Section IV.

B. Impacts of the Number of User Communities

Fig. 4(a) shows the impacts of the number of deployed A-UAVs (i.e., number of communities) on availability, while keeping the number of F-UAVs constant. These results are computed analytically from the equations provided in Section V. The numbers show percentage increase in availability compared to the no-F-UAV case. The figure shows that the benefits of data ferrying consistently go down with increasing number of A-UAVs. The main reason for this is in the reduction in probability P_{FA} (i.e., in Eqn. 1) of physical access to the F-UAVs due to the increase in their overall cycle times. This can be mitigated using more F-UAVs and is shown later.

A content can be provisioned to a user either by its local A-UAV or by a visiting F-AUV. Hence, availability has an A-UAV component and an F-UAV component. These two are shown separately in Fig 4(b). As expected, as the amount of duplicated cached contents in the A-UAVs go up (i.e., with larger λ), the contribution from the A-UAVs go up accordingly. A-UAVs' contribution, however, is lesser for larger number of communities since the unique contents are uniformly randomly distributed across more A-UAVs as explained in Section IV. The contributions of the F-UAVs reduce because of the fall in P_{FA}, as stated for Fig. 4(a).

C. Impacts of Deploying Multiple F-UAVs

Deploying more F-UAVs increase the probability of physical access to the F-UAVs (i.e., P_{FA}), thus improving availability over the corresponding no-F-UAV scenarios which is shown in Fig. 4(c). The results are with 20 A-UAVs, computed analytically and from simulation. The improvement in P_{FA} with increasing number of F-UAVs can be derived from Eqn.1 as $\Delta P_{FA} = \frac{(N_F^{new} - N_F^{old}) \times (TAD + T_{Hover})}{N_A \times (T_{Hover} + T_{Transit})}$. Here N_F^{new} and N_F^{old} are the number of deployed F-UAVs after and before additional deployments. ΔP_{FA} shows the rise in accessibility of F-UAVs to communities, which in turn, improves overall availability as given in Eqns. 1-4.

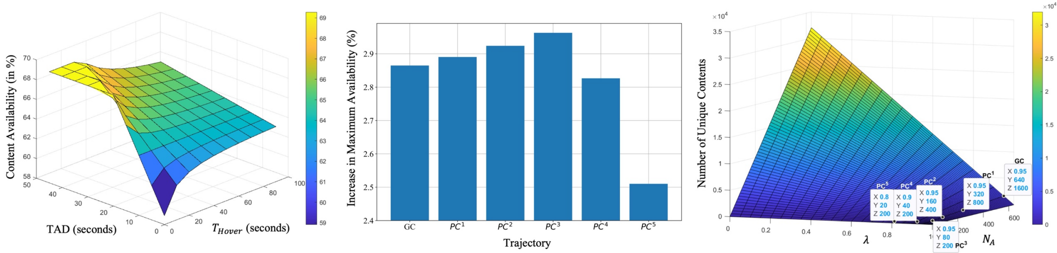
D. Effects of Hover Time and Tolerable Access Delay

Content availability is impacted by both the F-UAV hover time and the user-specified TAD in an interdependent manner. Those dependencies are shown in Fig. 5(a) with $N_A = 10, N_F = 5$, and $\lambda = 0.8$. The figure shows non-monotonic behavior of availability with varying hovering time and TAD. One notable observation is that for low $TADs$, availability increases with increase in hover time T_{Hover} and otherwise for high $TADs$. This can be explained as follows. First, for $TAD < T_{Transit}$, when an F-UAV travels from community *i* to next community *j*, the F-UAV does not contribute to availability at community *i* or *j* for $T_{Transit} - TAD$ duration (see Fig. 2). In this case, it is advantageous for the F-UAV to hover over a community. Second, for $TAD > T_{Transit}$, increase in hovering time reduces the possibility of the condition $(TAD - T_{Hover}) > T_{Transit}$ to be true. In other words, the possibility of exhausting the given TAD before reaching next community increases. So, it is beneficial to hover less, which increases the accessibility of F-UAVs at future communities in the cycle before TAD expires. Finally, for $TAD = T_{Transit}$, an F-UAV adds to availability within TAD irrespective of its hovering decision.

E. Effect of F-UAV Trajectory on Content Availability

Trajectory of an F-UAV has an impact on what content it carries and its contribution to the overall content availability based on A-UAVs in its trajectory cycle. Fig. 5(b) depicts those impacts for a system with 640 A-UAVs, 128 F-UAVs, $C_F = 200$, $\alpha = 0.8$ and all other default parameters calculated analytically. Increase in availability is reported as a percentage difference between the baseline no-F-UAV case and maximum content availability (i.e., at the optimal duplication factor λ) for specific F-UAV trajectories.

The global cycle (GC) trajectory refers to when an F-UAV visits all A-UAVs in a cycle. The content volume (i.e., $C_A.(1-\lambda).N_A$) to fill the F-UAVs in this trajectory scenario is quite high due to the large number of A-UAVs in the cycle, which is shown in Fig. 5(c). The next trajectory that was experimented with is partitioned cycle-1 (PC^1). In this, the A- and F-UAVs are divided into two sets, which are 2 sets of 320 A-UAVs and 2 sets of 64 F-UAVs. F-UAVs from the first set cycle around the first set of the A-UAVs, and the same applies to the second sets of A-UAVs and F-UAVs. Functionally, this scenario is equivalent to scaled down GC system with half as many A-UAVs and F-UAVs used for the GC results. In this scenario, the content availability is slightly larger than the GC case as can be seen in Fig. 5(b). The reasons are as follows. First, due to less cycle duration probability P_{FA} increases (i.e., in Eqn. 1). Second, is the sufficiency of unique contents in the system due to adequate count of A-UAVs in the cycle (Fig. 5(c)). Third, the optimal duplication factor λ is same for both i.e., 0.95. Thus, any increase in P_{FA} will increase content



(a) (b) (c)

Fig. 5 (a) Availability for variable T_{hover} and TAD; (b) Increase in availability for different trajectories; (c) Unique contents for varying N_A and λ

availability at optimal λ. The second and third partitioned cycles (PC^2 and PC^3) are functionally identical to PC^1 except that in these cases, both F-UAVs and A-UAVs are divided into 4 and 8 equal sets, respectively. Due to enough A-UAVs in the cycles to fill respective F-UAVs, content availability increases. Dividing the A-UAVs and F-UAVs further into 16 and 32 equal sets (i.e., in PC^4, PC^5) leads to reduction in availability due to the fewer A-UAVs in each F-UAV cycle (Fig. 5(b)). In such cases, the cache space in the F-UAVs go underutilized at the optimal λ value. To ensure adequate filling up of F-UAV, a sub-optimal λ is chosen which reduces duplication. This can be seen in Fig. 5(c) where λ, reduces from 0.95 to 0.90 for PC^4 and 0.80 for PC^5. This indicate that for a given number of A- and F-UAVs, there exists an optimal partitioning at which the overall content availability can be maximized.

F. Effects of Content Duplication on Access Delay

Fig. 6 shows consistent reduction in average access delay with increasing A-UAV duplication factor λ which is computed from the analytical equations given in Section V.

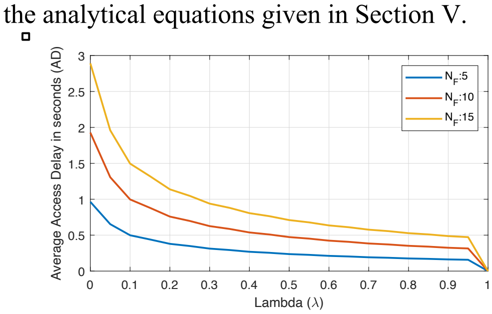


Fig. 6. Increase in delay with increasing F-UAVs for varying λ

This reduction is explained as follows. With higher λ, less popular contents are cached in F-UAVs. As contents with low popularity are less likely to be requested according to Zipf distribution (see Section III), the average access delay also goes down accordingly. Substantial reduction in access delay due to underutilization of F-UAV's cache, explained in Section IV, can be seen in Fig. 6 for values above $\lambda = 0.95$. It can also be seen that with increase in number of F-UAVs, the average content access delay increases. As, delay is only due to contents that are cached in F-UAVs, more F-UAVs increase the quantity P_{CAD} which adds to access delay.

An F-UAV's hover time impacts its overall cycle duration, that affects the duration for which the content availability from that F-AUV to the users remains low. During such Low Availability Periods (LAP), as explained in Eqn. 5 in Section V, only the locally cached contents from A-AUV's remain available. LAP reduces when more F-UAVs are deployed. This underlying effect is visible in Fig. 3 where adding F-UAVs reduces LAP and boosts availability.

VII. Conclusion and Future Work

This article investigates caching policies in UAV networks for content dissemination in communication challenged systems. Cache-enabled UAVs serve communities of users in a disaster/war-stricken area by caching popular contents in order to reduce downloading needs using satellites and other expensive vertical links. A framework is adopted in which two types of UAVs, namely anchor UAVs and ferrying UAVs, are deployed. Through analytical modeling and simulation experiments, the paper establishes an optimal content duplication strategy in which certain number of popular objects are duplicated in all anchor UAVs and certain number of non-duplicated/unique contents are carried in both types of UAVs. It was shown that content availability in such a system can be maximized by appropriately dimensioning the content duplication factor. The system was functionally validated, and performance evaluated for a different scenario including various ferrying UAV trajectories. Future work on this problem will be to develop runtime and dynamic mechanisms for all those design components so that content popularities, optimal caching, and the best UAV trajectories can be learnt online.

VIII. References

[1] Wang, Rui, et al. "Collaborative caching for dynamic map dissemination in vehicular networks." *ComSoc International Communications Quality and Reliability Workshop*. IEEE, 2019.

[2] Blasco, Pol, and Deniz Gündüz. "Learning-based optimization of cache content in a small cell base station." *ICC*. IEEE, 2014.

[3] Keränen, Ari, Jörg Ott, and Teemu Kärkkäinen. "The ONE simulator for DTN protocol evaluation." *Proceedings of the 2nd international conference on simulation tools and techniques*. 2009.

[4] Le, Michael, Joon-Sang Park, and Mario Gerla. "UAV assisted disruption tolerant routing." *Military Communications Conference*. IEEE, 2006.

[5] Zhao, Nan, et al. "UAV-assisted emergency networks in disasters." IEEE Wireless Communications 26.1 (2019): 45-51.

[6] Liu, Xiaonan, et al."Transceiver design and multihop D2D for UAV IoT coverage in disasters."IEEE Internet of Things Journal6.2(2018): 1803.

[7] Ejaz, Waleed, et al. "Unmanned aerial vehicles enabled IoT platform for disaster management." Energies 12.14 (2019): 2706.

[8] Xu, Xiaoli, et al. "Overcoming endurance issue: UAV-enabled communications with proactive caching." IEEE Journal on Selected Areas in Communications 36.6 (2018): 1231-1244.

[9] Lin, Xiaosheng, et al."Probabilistic caching placement in UAV-assisted heterogeneous wireless networks." Physical Communication 33 (2019): 54-61.

[10] Wang, Yuntao, et al. "Disaster Relief Wireless Networks: Challenges and Solutions." IEEE Wireless Communications 28.5 (2021): 148-155.

[11] Wu, Huaqing, et al. "Optimal UAV caching and trajectory in aerial-assisted vehicular networks: A learning-based approach." IEEE JSAC (2020): 2783.

[12] Zhang, Tiankui, et al. "Caching placement and resource allocation for cache-enabling UAV NOMA networks." IEEE Transactions on Vehicular Technology 69.11 (2020): 12897-12911.

[13] Wang, Rui, et al. "Connectionless Edge-Cache Servers for Reducing Cellular Bandwidth Usage in Vehicular Networks." COMSNETS. IEEE, 2021.