Caching in the Sky: Proactive Deployment of Cache-Enabled Unmanned Aerial Vehicles for Optimized Quality-of-Experience

Mingzhe Chen, Mohammad Mozaffari, Walid Saad, Changchuan Yin, Merouane Debbah, Choong Seon Hong

Caching in the Sky: Proactive Deployment of Cache-Enabled Unmanned Aerial Vehicles for Optimized Quality-of-Experience

Mingzhe Chen[1], Mohammad Mozaffari[2], Walid Saad[2,5], Changchuan Yin[1], Mérouane Debbah[3,4], and Choong-Seon Hong[5]

[1] Beijing Laboratory of Advanced Information Network, Beijing University of Posts and Telecommunications, Beijing, China 100876, Emails: chenmingzhe@bupt.edu.cn and ccyin@ieee.org.

[2] Wireless@VT, Electrical and Computer Engineering Department, Virginia Tech, VA, USA, Emails:{mmozaff,walids}@vt.edu.

[3] Large Networks and Systems Group (LANEAS), CentraleSupélec, Université Paris-Saclay, Gif-sur-Yvette, France.

[4] Mathematical and Algorithmic Sciences Lab, Huawei France R & D, Paris, France, Email: merouane.debbah@huawei.com.

[5] Department of Computer Science and Engineering, Kyung Hee University, Yongin, South Korea, Email: cshong@khu.ac.kr.

***Abstract*—In this paper, the problem of proactive deployment of cache-enabled unmanned aerial vehicles (UAVs) for optimizing the quality-of-experience (QoE) of wireless devices in a cloud radio access network (CRAN) is studied. In the considered model, the network can leverage human-centric information such as users' visited locations, requested contents, gender, job, and device type to predict the content request distribution and mobility pattern of each user. Then, given these behavior predictions, the proposed approach seeks to find the user-UAV associations, the optimal UAVs' locations, and the contents to cache at UAVs. This problem is formulated as an optimization problem whose goal is to maximize the users' QoE while minimizing the transmit power used by the UAVs. To solve this problem, a novel algorithm based on the machine learning framework of conceptor-based echo state networks (ESNs) is proposed. Using ESNs, the network can effectively predict each user's content request distribution and its mobility pattern when limited information on the states of users and the network is available. Based on the predictions of the users' content request distribution and their mobility patterns, we derive the optimal locations of UAVs as well as the content to cache at UAVs. Simulation results using real pedestrian mobility patterns from BUPT and actual content transmission data from Youku show that the proposed algorithm can yield 33.3% and 59.6% gains, respectively, in terms of the average transmit power and the percentage of the users with satisfied QoE compared to a benchmark algorithm without caching and a benchmark solution without UAVs.**

I. Introduction

The next-generation of cellular systems is expected to be largely user centric and, as such, it must be cognizant of human-related information such as users' behavior, mobility patterns, and quality-of-experience (QoE) expectations [1]. One promising approach to introduce such wireless network designs with human-in-the-loop is through the use of cloud radio access networks (CRANs) [2]. In CRANs, a central cloud processor can parse through the massive users' data to learn the users' information such as content request distribution and mobility patterns, then, determine how to manage resources in the network. However, an effective exploration of human-in-the-loop features in a CRAN faces many challenges that range from effective predictions to user behavior tracking, effective caching, and optimized resource management.

*This work was supported in part by the National Natural Science Foundation of China under Grants 61671086, 61629101, and 60972073, by the ERC Starting Grant 305123 MORE, and by the U.S. National Science Foundation under Grants IIS-1633363, CNS-1460316 and CNS-1513697.

A. Related Works

The existing literature has studied a number of problems related to caching in CRANs and heterogeneous networks with human-in-the-loop such as [3]–[9]. In [3], the authors exploited the instantaneous demands of wireless users to estimate the content popularity and devise an optimal random caching strategy. The authors in [4] proposed an echo state network to predict the users' content request distribution and mobility patterns in CRANs. The work in [5] proposed a caching-based milimiter wave (mmWave) framework, in which base stations pre-store video contents and service users with a high data rate. In [6], a caching framework aimed at fully exploiting the potential of such CRAN systems through cooperative hierarchical caching is proposed. The authors in [7] investigated the fully cooperative caching case during which a centralizer helps all base stations to make caching decisions. The work in [8] studied the use of cache-enabled small base stations to alleviate the load of backhaul. In [9], a content caching strategy is proposed to jointly minimize the cell average outage probability and fronthaul usage in CRANs. However, most of these existing caching works [3]–[9] were typically restricted to static networks without mobility and ultra dense users. Note that, in these contributions [3]–[9], the cached content is stored at the terrestrial static base stations. However, in an area with ultra dense users and high rise buildings (i.e. stadium or hotspots), the static ground base stations with caching may not be able to meet high capacity demands of the users. Moreover, caching at ground base stations may not be effective in serving the mobile users once they move outside the coverage range of the ground base stations. For instance, whenever a mobile user moves to a new cell, its requested content may not be available at the new base station and, consequently, the users cannot be serviced properly. In such case, to effectively serve mobile user, the requested content needs to be cached at multiple base stations which may not be efficient. Therefore, there is a need to introduce a more flexible base station that can boost the capacity and track the users' mobility patterns so as to improve the caching efficiency. To this end, unmanned aerial vehicles (UAVs) can be used as flying base stations to dynamically cache the popular contents, track the mobility pattern of the corresponding users then, effectively serve them. In this case, due to the high altitude and flexible deployment of the UAVs,

they can establish reliable communication links to the users by mitigating the blockage effect.

The use of UAVs for enhancing wireless communications in cellular and ad hoc networks was studied in [10]–[17]. However, this existing literature [10]–[17] was focused on performance analysis and did not consider prediction of user-centric behavioral patterns such as mobility nor it study the use of UAVs for caching purposes. The prediction of the users' mobility patterns can enable the UAVs to effectively move and provide service for the ground users. Moreover, when UAVs are considered within a CRAN system, the network must take into account the fact that the fronthaul links that connect the UAVs to the cloud will be capacity-limited. This is due to the fact that the bandwidth of the UAVs fronthaul links is limited. To overcome this limited-fronthaul capacity challenge, one can use content caching techniques to proactively download and cache content at the UAVs during off peak hours or when the UAVs are back at their docking stations. The use of caching enables the UAVs directly transmit the content to its requested user, thus reducing the fronthaul traffic load.

Some recent works such as in [4] and [18]–[23] have studied a number of ideas related to the predictions of human behavior in wireless networking scenarios. In [18] and [19], the authors proposed a prediction algorithm for the users' mobility patterns based on a deep learning algorithm and a semi-Markov process. In [20], a type of user-initiated network is proposed for cellular users to trade data plans by leveraging personal hotspots with users' smartphones. The work in [21] investigated the predictable degree of users' mobility patterns. Nevertheless, the mobility prediction works in [18]–[21] focused only on the prediction phase and did not study how the users' mobility patterns can be used to optimize the wireless performance using user-centric caching and resource allocation techniques. The authors in [22] developed a data extraction method using the Hadoop platform to predict content popularity. The work in [23] proposed a fast threshold spread model to predict the future access patterns of multimedia content based on social information. However, the works in [22] and [23] do not consider the complexity of using the users' traffic data while predicting the users' content popularity. In particular, in a dense network with high traffic demands, the proposed approaches in [22] and [23] are not practical as they require each base station to store all users' traffic data to predict the users' content popularity. In contrast, our proposed conceptor based ESN algorithm can record the users' historical traffic data and, consequently, use them to predict the users' behavior. In this case, the complexity of the learning algorithm will be significantly lower compared to the algorithms without ability of recording historical data. The most related work here is our work [4] in which we exploit the echo state network to predict the content request distribution. However, in [4], the ESN-based algorithm can only predict one non-linear system. Here, we proposed an *conceptor*-based ESN algorithm which can separate the users' behavior into different patterns and learn these patterns independently, thus leading to a significant improvement in the accuracy of predictions. For example, in order to predict the user's mobility patterns, the ESN algorithm in [4] will collect all users' mobility data from Monday to Sunday to train the ESN as a prediction system. However, the conceptor-based ESN algorithm can separate the data from Monday to Sunday into seven patterns and use the data of seven patterns to train the conceptor-based ESN as seven independent prediction systems. In this case, the conceptor-based ESN can use seven independent prediction system to predict the user's mobility during each day. Since in reality the user's mobility pattern in each day will be different, the conceptor-based ESN algorithm can improve the accuracy of the prediction.

B. Contributions

The main contribution of this paper is to develop a novel framework that leverages user-centric information, such as content request distribution and mobility patterns, to effectively deploy cache-enabled UAVs while maximizing the users' QoE using a minimum total transmit power of the UAVs. The adopted QoE metric captures human-in-the-loop features such as transmission delay and the users' perceptions on the rate requirement, depending on their device type. In the proposed framework, the cloud can accurately predict the content request distribution and mobility patterns of each user. These predictions of user's behavior can then be used to find the optimal locations and content caching strategies for the UAVs. Unlike previous studies such as [18]–[23] that predict the users' behavior using only one non-linear system, we propose a conceptor-based echo state network (ESN) approach to perform users' behavior prediction. Such an ESN model with conceptors enables the cloud to separate the users' behaviors into different patterns and learn these patterns independently, thus leading to a significant improvement in the accuracy of predictions. Moreover, unlike previous studies such as [10]–[16] that consider the deployment of the UAVs assuming static users, we study the deployment of cache-enabled UAVs in CRANs with mobile users. In the proposed CRANs model, we derive the user-UAV association, the optimal locations of the UAVs as well as the content to cache at the UAVs. *To our best knowledge, this work is the first to analyze the use of caching at the level of UAVs, given ESN-based predictions on the users' behavior*. To evaluate the performance of the proposed approach, we use real data from Youku for content requests as well as realistic measured mobility data from the Beijing University of Posts and Telecommunications for mobility simulations. Simulation results show that the proposed algorithm can yield 33.3% gain in terms of the average transmit power of the UAVs compared to a baseline algorithm without cache. Moreover, the proposed algorithm can also yield 59.6% gain in terms of the percentage of the users with satisfied QoE compared to a benchmark scenario without UAVs. In summary, the main contributions of the paper are as follows:

1) We propose a novel cache-enabled UAV framework in CRANs that can meet the mobile user's QoE requirement while minimizing the transmit power of the UAVs. In this framework, the capacity of the links from the BBUs to the UAVs is limited and the transmission from the UAVs to the users is over mmWave. To our best knowledge, this is the first work that considers the limited capacity links from the BBUs to the UAVs.

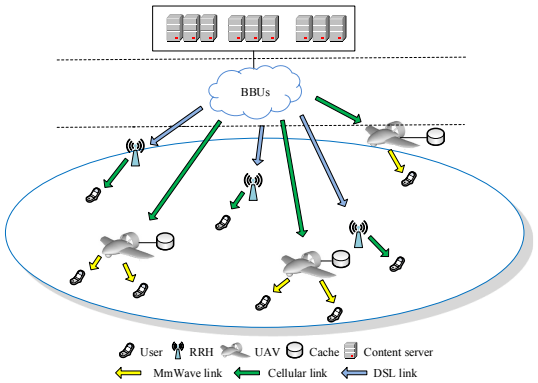


Fig. 1. A CRAN with cache-enabled UAVs.

2) We analyze the use of caching at the level of UAVs. Here, caching is used to store the most popular contents that the users may request. Caching with UAVs can reduce the transmission delay and hence, reduce the transmit power of the UAVs. This is the first work that considers caching with UAVs.
3) We develop an effective approach for leveraging the cache-enabled UAVs to service mobile wireless users. The users have their own mobility patterns which are measured from the realistic students. The cache-enabled UAVs will service the users that the RRHs cannot meet the users' rate requirement.
4) We introduce a new conceptor-based ESN learning algorithm to predict the users' content request distribution and mobility patterns with users' contexts. The context represents the users' information related to the users' behavior such as mobility pattern. Such an ESN model with conceptors enables the cloud to separate the users' behavior into different patterns and learn these patterns independently, thus leading to a significant improvement in the accuracy of predictions.
5) We perform fundamental analysis on the user-UAV association, optimal locations of the UAVs as well as the content to cache at UAVs.

The rest of this paper is organized as follows. The system model and problem formulation are presented in Section II. The conceptor ESN for content request distribution and mobility patterns predictions is proposed in Section III. The proposed approach for user-UAV association, content caching, and optimal location of each UAV is presented in Section IV. In Section V, we provide numerical and simulation results. Finally, conclusions are drawn in Section VI.

II. System Model and Problem Formulation

Consider the downlink of a CRAN system servicing a set $\mathcal{U}$ of U mobile users via a set $\mathcal{R}$ of R remote radio heads (RRHs) acting as distributed antennas. The RRHs are grouped into E clusters using *K-mean clustering* approach [24] so that zero-forcing beamforming (ZFBF) [25] can be used to service the users. In this system, a set $\mathcal{K}$ of K UAVs equipped with cache storage units can be deployed to act as flying cache-enabled RRHs to serve the ground users along with the terrestrial RRHs. For the UAV-to-users communication links, since the high altitude of the UAVs can significantly reduce the blocking effect due to obstacles, we consider air-to-ground UAV transmissions using the millimeter wave (mmWave) frequency spectrum. Meanwhile, the terrestrial RRHs transmit over the cellular band and are connected to the cloud's pool of the baseband units (BBUs) via capacity-constrained, digital subscriber line (DSL) fronthaul links. Further, the cloud connects to the content servers via fiber backhaul links. The transmissions from the cloud to the UAVs occurs over wireless fronthaul links using the licensed cellular band. Consequently, the UAVs' wireless fronthaul links may interfere with the transmission links from the RRHs to the users when the user's requested content needs to be transmitted from the content server.

In our model, the content server stores a set $\mathcal{N}$ of all N contents required by all users. The contents are of equal size L. Caching at the UAVs, referred to as "UAV cache" hereinafter, will be used to store the popular content that the users request. By caching predicted content, the transmission delay from the content server to the UAVs can be significantly reduced as each UAV can directly transmit its stored content to the users. Different from caching at the RRHs or BBUs, caching at UAVs allows servicing mobile users when their QoE requirement cannot be satisfied by the RRHs. We denote the set of C cached contents in the storage units of UAV k by $\mathcal{C}_k$, where $C \leq N$ and $k \in \mathcal{K}$. For simplicity, we assume that each user can request at most one content during each specified time slot τ. We also let Δ_τ be the duration of time slot τ that also represents the maximum transmission duration of each content. The maximum transmission duration Δ_τ is determined by the proposed algorithm in Section IV. We assume that the content stored at the UAV cache will be refreshed every period that consists of T time slots and this UAVs caching is performed at off peak hours when the UAVs return to their cloud-based docking stations for purposes such as battery charge. Table I provides a summary of the notations used in this paper.

A. Mobility Model

In our system, we assume that the users can move continuously. In this case, we consider a realistic model for periodic, daily, and *pedestrian mobility patterns* according to which each user will regularly visit a certain location of interest. For example, certain users will often go to the same office for work at the same time during weekdays. The locations of each user are collected by the BBUs once every H time slots. Here, the duration of H time slots is considered a period of one hour for each user. In addition, we assume that each user moves between two collected locations at a constant speed. The mobility pattern for each user will then be used to determine the content that must be cached as well as the optimal location of each UAV which will naturally impact the QoE of each user.

In this model, the associations of the mobile users with the UAVs or the RRHs can change depending on the QoE requirement. Since the users are moving continuously, the locations of the UAVs must change accordingly so as to serve the users effectively. However, for tractability, we assume that the UAVs will remain static during each content transmission. In essence, the UAVs will update their locations according to the mobility of the users after each content transmission is complete at a current location.

TABLE I
LIST OF NOTATIONS

Notation	Description	Notation	Description
U	Number of users	C	Number of contents stored at UAV cache
K	Number of UAVs	F	Number of intervals in each time slot
R	Number of RRHs	H	Number of time slots to collect user mobility
P_R	Transmit power of RRHs	$P_{t,ki}$	Transmitted power of UAV or RRH
N	Number of contents	τ, Δ_τ	Time slot index, Time slot duration
$l_{t,ki}$	Path loss of UAVs-users	$d_{t,ki}$	Distance between RRHs or UAVs and users
$x_{\tau,k}, y_{\tau,k}, h_{\tau,k}$	Coordinates of UAVs	$\delta_{S_i,n}$	Rate requirement of device type
L_{FS}	Free space path loss	d_0	Free-space reference distance
f_c	Carrier frequency	$l^F_{t,ki}$	Path loss of fronthaul links
μ_{LoS}, μ_{NLoS}	Path loss exponents	$\chi_{\sigma_{\text{LoS}}}, \chi_{\sigma_{\text{NLoS}}}$	Shadowing random variable
$\gamma^{\text{V}}_{t,ki}, \gamma^{\text{H}}_{t,ki}$	SINR of user i	$L^{\text{LoS}}_{t,k}$, $L^{\text{NLoS}}_{t,k}$	LoS/NLoS path loss from the BBUs to UAV k
t, Δ_t	Small interval, interval duration	$l^{\text{LoS}}_{t,k}$, $l^{\text{NLoS}}_{t,k}$	LoS/NLoS path loss from UAV k to users
c	Speed of light	$h_{t,ki}$	Channel gains between RRHs k and user i
$\bar{D}_{\tau,i,n}$	Delay	$C^F_{\tau,ki}$	Fronthaul rate of UAV or RRH k
$C^{\text{V}}_{\tau,ki}$	Rate of UAV-user link	$C^{\text{H}}_{\tau,qi}$	Rate of RRH-user link
$Q_{\tau,i,n}$	QoE of each user i	T	Number of time slots for caching update
$x_{t,i}, y_{t,i}$	Coordinates of users	P_B	Transmit power of the BBUs

B. Transmission Model

Next, we introduce the models for transmission links between BBUs and UAVs, UAVs and users, and RRHs and users. For ease of exposition, a time slot τ is discretized into F equally spaced time intervals Δ_t, i.e., $\Delta_\tau = F\Delta_t$. The time interval Δ_t is chosen to be sufficiently small so that each user's location can be considered constant during t as in [11].

1) UAVs-Users Links: The mmWave propagation channel of the UAVs-user link is modeled using the standard log-normal shadowing model of [26]. The standard log-normal shadowing model can be used to model the line-of-sight (LoS) and non-line-of-sight (NLoS) links by choosing specific channel parameters. Therefore, the LoS and NLoS path loss of UAV k located at $(x_{\tau,k}, y_{\tau,k}, h_{\tau,k})$ transmitting a content to user i at interval t of time slot τ is [27] (in dB):

$$l^{\text{LoS}}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right) = L_{FS}\left(d_0\right) + 10\mu_{\text{LoS}}\log\left(d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right)\right) + \chi_{\sigma_{\text{LoS}}}, \quad (1)$$

$$l^{\text{NLoS}}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right) = L_{FS}\left(d_0\right) + 10\mu_{\text{NLoS}}\log\left(d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right)\right) + \chi_{\sigma_{\text{NLoS}}}, \quad (2)$$

where $\boldsymbol{w}_{\tau,t,k} = [x_{\tau,k}, y_{\tau,k}, h_{\tau,k}]$ is the coordinate of UAV k during time slot τ with $h_{\tau,k}$ being the altitude of UAV k at time slot τ. Also, $\boldsymbol{w}_{\tau,t,k} = [x_{t,i}, y_{t,i}]$ is the time-varying coordinate of user i at interval t. $L_{FS}\left(d_0\right)$ is the free space path loss given by $20\log\left(d_0 f_c 4\pi/c\right)$ with d_0 being the free-space reference distance, f_c being the carrier frequency and c being the speed of light. $d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right) = \sqrt{\left(x_{t,i} - x_{\tau,k}\right)^2 + \left(y_{t,i} - y_{\tau,k}\right)^2 + h^2_{\tau,k}}$ is the distance between user i and UAV k. μ_{LoS} and μ_{NLoS} are the path loss exponents for LoS and NLoS links. $\chi_{\sigma_{\text{LoS}}}$ and $\chi_{\sigma_{\text{NLoS}}}$ are the shadowing random variables which are, respectively, represented as the Gaussian random variables in dB with zero mean and σ_{LoS}, σ_{NLoS} dB standard deviations.

In our model, the probability of LoS connection depends on the environment, density and height of buildings, the locations of the user and the UAV, and the elevation angle between the user and UAV. The LoS probability is given by [10] and [13]:

$$\Pr\left(l^{\text{LoS}}_{t,ki}\right) = \left(1 + X\exp\left(-Y\left[\phi_t - X\right]\right)\right)^{-1}, \quad (3)$$

where X and Y are constants which depend on the environment (rural, urban, dense urban, or others) and $\phi_t = \sin^{-1}\left(h_{\tau,k}/d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right)\right)$ is the elevation angle. Clearly, the average path loss from the UAV k to user i at interval t is [13]:

$$\bar{l}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right) = \Pr\left(l^{\text{LoS}}_{t,ki}\right) \times l^{\text{LoS}}_{t,ki} + \Pr\left(l^{\text{NLoS}}_{t,ki}\right) \times l^{\text{NLoS}}_{t,ki}, \quad (4)$$

where $\Pr\left(l^{\text{NLoS}}_{t,ki}\right) = 1 - \Pr\left(l^{\text{LoS}}_{t,ki}\right)$. Based on the path loss, the average signal-to-noise ratio (SNR) of user i located at $\boldsymbol{w}_{\tau,t,i}$ from the associated UAV k at interval t is given by:

$$\gamma^{\text{V}}_{t,ki} = \frac{P_{t,ki}}{10^{\bar{l}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right)/10}\sigma^2}, \quad (5)$$

where $P_{t,ki}$ is the transmit power of UAV k to user i at time t, and σ^2 is the variance of the Gaussian noise. We assume that the total bandwidth available for each UAV is B_V which is equally divided among the associated users. The channel capacity between UAV k and user i for each content transmission will be $C^{\text{V}}_{\tau,ki} = \frac{1}{F_{\tau,i}}\sum_{t=1}^{F_{\tau,i}}\frac{B_V}{U_k}\log_2\left(1 + \gamma^{\text{V}}_{t,ki}\right)$, where U_k is the number of the users associated with UAV k and $F_{\tau,i}$ is the number of the intervals that user i uses to receive a content during time slot τ.

2) BBUs-UAVs Ground-to-Air Links: For the BBUs-UAVs (ground-to-air) link, we consider probabilistic LoS and NLoS links over the licensed band. Since the distance of the UAVs fronthaul link may be larger compared to the distance of the UAV-user link, the cellular band can provide a more reliable transmission and a smaller path loss compared to the mmWave channel. In such a model, NLoS links experience higher attenuations than LoS links due to the shadowing and diffraction loss. The LoS and NLoS path loss from the BBUs to UAV k at time t of time slot τ can be given by [10]:

$$L^{\text{LoS}}_{t,k} = d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,B}\right)^{-\beta}, \quad (6)$$

$$L^{\text{NLoS}}_{t,k} = \eta d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,B}\right)^{-\beta}, \quad (7)$$

where $\boldsymbol{w}_{\tau,t,B} = [x_B, y_B]$ is the location of the BBUs, and β is the path loss exponent. The LoS connection probability and the average SNR of the link between the BBUs and UAV k can be calculated using (3)-(5).

3) RRHs-Users Links: In our model, RRHs are grouped into E clusters. Then, the RRHs in each cluster use ZFBF to improve the users' rates. The received signals of the users

associated with RRHs cluster q at interval t is:

$$\boldsymbol{b}_{t,q} = \sqrt{P_R}\boldsymbol{H}_{t,q}\boldsymbol{F}_{t,q}\boldsymbol{a}_{t,q} + \boldsymbol{n}, \tag{8}$$

where $\boldsymbol{H}_{t,q} \in \mathbb{R}^{U_q \times R_q}$ is the path loss matrix with U_q being the number of users associated with RRH cluster q, and R_q is the number of RRHs' antennas. P_R is the transmit power of each RRH which is assumed to be equal for all RRHs. $\boldsymbol{a}_{t,q} \in \mathbb{R}^{U_q \times 1}$ is the transmitted content at interval t and $\boldsymbol{n}_{t,q} \in \mathbb{R}^{U_q \times 1}$ is the noise power. Also, $\boldsymbol{F}_{t,q} = \boldsymbol{H}_{t,q}^{\mathrm{H}}\left(\boldsymbol{H}_{t,q}\boldsymbol{H}_{t,q}^{\mathrm{H}}\right)^{-1} \in \mathbb{R}^{R_q \times U_q}$ is the beamforming matrix [28]. We also assume that the bandwidth of each user associated wth the RRHs is B. Then, the received signal-to-interference-plus-noise-ratio (SINR) of user i in cluster $\mathcal{M}_q$ at interval t will be:

$$\gamma_{t,qi}^{\mathrm{H}} = \frac{P_R\left\|\boldsymbol{h}_{t,qi}\boldsymbol{f}_{t,qi}\right\|^2}{\underbrace{\sum_{j=1,j\neq q}^{E}\sum_{u\in\mathcal{U}_j} P_R\left\|\boldsymbol{h}_{t,ji}\boldsymbol{f}_{t,ju}\right\|^2}_{\text{other cluster RRHs interference}} + \underbrace{P_B g_{t,Bi} d_{t,Bi}^{-\beta}}_{\text{wireless fronthaul interference}} + \sigma^2},$$

where $\mathcal{M}_j$ is the set of the RRHs in group j, $\mathcal{U}_j$ is the set of the users associated with the RRHs in group j, $\boldsymbol{h}_{t,qi} \in \mathbb{R}^{1\times R_q}$ is the channel gain between the RRHs in cluster $\mathcal{M}_q$ and user i with $h_{t,ki} = g_{t,ki}d_{t,ki}\left(x_i, y_i\right)^{-\beta}$, $g_{t,ki}$ is the Rayleigh fading parameter at interval t, and $d_{t,ki}\left(x_i, y_i\right) = \sqrt{\left(x_{t,k} - x_{t,i}\right)^2 + \left(y_{t,k} - y_{t,i}\right)^2}$ is the distance between RRH k and user i at interval t. $\boldsymbol{f}_{t,qi} \in \mathbb{R}^{R_q \times 1}$ is the beamforming vector. Given (9), the channel capacity between RRH cluster $\mathcal{M}_q$ and user i for each content transmission is:

$$C_{\tau,qi}^{\mathrm{H}} = \frac{1}{F_{\tau,i}}\sum_{t=1}^{F_{\tau,i}} B\log_2\left(1 + \gamma_{t,qi}^{\mathrm{H}}\right). \tag{9}$$

C. Quality-of-Experience Model

Given the proposed models in the previous subsections, here, we present the QoE model for each user. The *quality-of-experience* of each user is formally defined as a concrete human-in-the-loop metric that captures each user's data rate, delay, and device type.

1) Delay: In the considered CRAN system, contents can be transmitted to the users via three types of links: (a) content server-BBUs-RRHs-user, (b) content server-BBUs-UAV-user, and (c) UAV cache-user. The backhaul link connecting the cloud to the core network is assumed to be fiber and, therefore, its delay is neglected. We assume that the capacity of the wired fronthaul links between the BBUs and the RRHs is limited to a maximum rate of v_F for all users. Consequently, the fronthaul rate for each user receiving a content from the RRHs will be $v_{FU} = v_F/N_{FR}$ with N_{FR} being the number of the users receive contents from the RRHs. Thus, the delay of a user i receiving content n over the three types of links at each time slot τ can be written as:

$$D_{\tau,i,n} = \begin{cases} \frac{L}{v_{FU}} + \frac{L}{C_{\tau,qi}^{\mathrm{H}}}, & \text{link}\,(a)\,, \\ \frac{L}{C_{\tau,k}^{F}} + \frac{L}{C_{\tau,ki}^{\mathrm{V}}}, & \text{link}\,(b)\,, \\ \frac{L}{C_{\tau,ki}^{\mathrm{V}}}, & \text{link}\,(c)\,, \end{cases} \tag{10}$$

where $C_{\tau,k}^{F}$ is the rate of content transmission from the BBUs to UAV k which is calculated analogously to (II-B3) and (9). Next, we derive the lower bound on the delay that each user can tolerate for each content transmission.

Proposition 1. The lower bound of the delay for each user i receiving content n are given by:

$$\min\left\{\frac{L}{v_F}, \frac{L}{C_K^{\max}}\right\} \leq D_{\tau,i,n}, \tag{11}$$

where $C_K^{\max} = B_V\log_2\left(1 + \frac{P_{\max}}{10^{\left(L_{FS}(d_0)+10\mu_{\mathrm{LoS}}\log(h_{\min})-4\sigma_{\mathrm{LoS}}\right)/10}\sigma^2}\right)$ with $P_{\max}$ being the maximum transmit power of each UAV, and $h_{\min}$ being the minimum altitude of the UAV.

Proof. See Appendix A. □

From Proposition 1, we can see that the minimum delay of each user depends on the rate of the fronthaul links and the maximum transmit power of the UAVs. Therefore, we can improve the QoE of each user by adjusting the UAV's transmit power. In particular, as the number of users increases and the rate of fronthaul links decreases, the QoE requirement of users can be satisfied by adjusting the UAVs' transmit power. Note that, the upper bound of the delay Δ_τ is set by the system requirement. Using the results of Proposition 1, we can categorize the sensitivity to the delay into five groups using the popular mean opinion score (MOS) model [29] which is often used to measure the QoE of a wireless user. The mapping between delay and MOS model [29] is given by:

$$\bar{D}_{\tau,i,n} = \frac{\Delta_\tau - D_{\tau,i,n}}{\Delta_\tau - \min\left\{\frac{L}{v_F}, \frac{L}{C_K^{\max}}\right\}}, \tag{12}$$

which is shown in Table II.

2) Device Type: The screen size of each device type of the user will also affect the QoE perception of the user, especially for video-oriented applications. Indeed, users who own devices that have larger screens (such as tablets) will be more sensitive to QoE compared to those who own smaller devices (such as small smartphones). We capture the impact of the screen size of each user i using a parameter S_i that reflects the diameter length of the user's device. Typically, devices with a larger screen size, can display content at a higher resolution thus requiring a higher data rate. We assume that the rate requirement of user i with device S_i receiving a content n at interval t is $\delta_{S_i,n} = S_i\hat{C}_n$, where $\hat{C}_n$ is the rate requirement of each user receiving content n. The mapping from the rate requirement of user device to the MOS model is:

$$V_{t,i} = \begin{cases} 1, & j \geq \delta_{S_i,n}, \\ 0, & j < \delta_{S_i,n}, \end{cases} \tag{13}$$

where $j \in \left\{C_{t,ki}^{\mathrm{V}}, C_{t,qi}^{\mathrm{H}}\right\}$. Here, $V_{t,i} = 1$ indicates that the user's data rate satisfies the requirement of its device type and 0 represents the user's data rate cannot satisfy the requirement. In this case, the rate requirement of of a user's device is mapped to the MOS. The QoE of each user i receiving content n at time slot τ can be given by [29]:

$$Q_{\tau,i,n} = \zeta_1\bar{D}_{\tau,i,n} + \frac{\zeta_2}{F_{\tau,i}}\sum_{t=1}^{F_{\tau,i}} V_{t,i}, \tag{14}$$

where q_1 and q_2 are weighting parameters with $\zeta_1 + \zeta_2 = 1$.

TABLE II
MEAN OPINION SCORE MODEL [29]

QoE	Poor	Fair	Good	Very Good	Excellent
Interval scale	0-0.2	0.2-0.4	0.4-0.6	0.6-0.8	0.8-1

D. Problem Formulation

Here, we first find the minimum rate required to meet the QoE requirement of each user associated with the UAVs. Next, we determine the minimum transmit power of each UAV required to meet the QoE threshold of the associated users. Finally, we formulate the minimization problem. From Table II, we can see that, for $0.8 \leq \bar{D}_{\tau,i,n} \leq 1$, the MOS of delay will be "Excellent", which means that the delay is minimized. In this case, $\bar{D}_{\min} = 0.8$ is the minimum value that maximizes the delay component of user i's QoE, during the transmission of a given content n. We define the rate that achieves the optimal delay as the delay requirement and also, define the rate that meets the rate requirement of device as the device rate requirement. Consider the transmission between a UAV k located at $\boldsymbol{w}_{\tau,t,k}$ and a user i located at coordinates $\boldsymbol{w}_{\tau,t,i}$. From (10), the delay requirement for UAV k transmitting content n to user i at time slot τ is:

$$C^R_{\tau,ki,n} = \begin{cases} \frac{L}{\left(\Delta_\tau - \bar{D}_{\min}\left(\Delta_\tau - \min\left\{\frac{L}{v_F}, \frac{L}{C^{\max}_k}\right\}\right) - \frac{L}{C^F_{\tau,k}}\right)}, n \notin \mathcal{C}_k, \\ \frac{L}{\left(\Delta_\tau - \bar{D}_{\min}\left(\Delta_\tau - \min\left\{\frac{L}{v_F}, \frac{L}{C^{\max}_k}\right\}\right)\right)}, \quad n \in \mathcal{C}_k. \end{cases} \tag{15}$$

From (15), we can see that, by storing content n at cache of UAV k, the delay requirement for minimizing delay decreases.

Let $\delta_{S_i,n}$ be the device rate requirement of user i associated with a UAV. Clearly, the QoE is maximized when $C^{\mathrm{V}}_{t,ki} \geq \max\left\{C^R_{\tau,ki,n}, \delta_{S_i,n}\right\}$. Hence, the minimum rate required to maximize the user's QoE is $\delta^R_{i,n} = \max\left\{C^R_{\tau,ki,n}, \delta_{S_i,n}\right\}$. Based on (5), the minimum transmit power needed to guarantee the QoE requirement of user i receiving content n at interval t is:

$$P^{\min}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \delta^R_{i,n}, n\right) = \left(2^{\delta^R_{i,n} U_k / B_V} - 1\right)\sigma^2 10^{\bar{l}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right)/10}. \tag{16}$$

From (16), we can see that the minimum transmit power of UAV k transmitting content n to user i depends on the UAV's location, the rate needed to satisfy the QoE requirement of user i, and the transmitted content n.

Given this system model, our goal is to find an effective deployment of cache-enabled UAVs to enhance the QoE of each user while minimizing the transmit power of the UAVs. This problem involves predicting the content request distribution and periodic locations for each user, finding the optimal contents to cache at the UAVs, determining the users' associations and adjusting the locations*, and transmit power of the UAVs. This problem can be formulated as follows:

$$\min_{\mathcal{C}_k, \mathcal{U}_{\tau,k}, \boldsymbol{w}_{\tau,t,k}} \sum_{\tau=1}^{T} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} P^{\min}_{\tau,t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \delta^R_{i,n}, n_{\tau,i}\right), \tag{17}$$

$$\text{s. t. } h_{\min} \leq h_{\tau,k}, k \in \mathcal{K}, \tag{17a}$$

$$m \neq j, m, j \in \mathcal{C}_k, \mathcal{C}_k \subseteq \mathcal{N}, k \in \mathcal{K}, \tag{17b}$$

$$0 < P^{\min}_{\tau,t,ki} \leq P_{\max}, i \in \mathcal{U}, k \in \mathcal{K}, \tag{17c}$$

where $P^{\min}_{\tau,t,ki}$ is the minimum transmit power of UAV k to user i at interval t during time slot τ. $n_{\tau,i}$ is the content that user i requests at time slot τ, $\mathcal{U}_{\tau,k}$ is the set of the users that are associated with UAV k at time slot τ. $h_{\min}$ is the minimum altitude that each UAV can reach at time slot τ. Here, (17b) captures the fact that each cache storage unit at the UAV stores a single, unique content, and (17c) indicates that the transmit power of the UAVs should be minimized. Since the problem as per (17) is to satisfy the rate needed for meeting each user's QoE requirement during the next time period T, the predictions of users behavior will directly impact the solution. From (17), we can see that the prediction of the users' mobility patterns enable the BBUs to find the optimal locations of the UAVs. Moreover, by predicting the users' content request distribution the BBUs can determine the most popular content to cache at the UAVs. In addition, since the content transmission link affects the transmit power that the UAV must use to satisfy the user's QoE requirement, the problem of minimizing the transmit powers of the UAVs in (17) inherently incorporates the caching constraints.

III. CONCEPTOR ECHO STATE NETWORKS FOR CONTENT AND MOBILITY PREDICTIONS

In this section, we propose a prediction algorithm using the framework of ESN with conceptors, to find the users' content request distributions and their mobility patterns. The predictions of the users' content request distribution and their mobility patterns will then be used in Section IV to find the user-UAV association, optimal locations of the UAVs and content caching at the UAVs. Echo state networks are a special type of recurrent neural networks designed for performing non-linear systems forecasting [30]. The ESN architecture is based on a randomly connected recurrent neural network, called reservoir, which is driven by a temporal input. The state of the reservoir is a rich representation of the history of the inputs so that a simple linear combination of the reservoir units is a good predictor of the future inputs. In our model, the reservoir will be combined with the input to store the users' context information and will also be combined with the trained output matrix to output the predictions of the users' content request distribution and mobility patterns. Here, a user's *context* is defined as the current state and attribute of a user including time, week, gender, occupation, age, and device type (e.g., tablet or smartphone). Therefore, an ESN-based approach can use the users' context to predict the corresponding behavior such as content request and mobility.

Compared to traditional neural network and deep learning approaches such as in [18], an ESN-based approach can quickly learn the mobility pattern and content request distribution without requiring significant training data due to the use of the echo state property. However, traditional ESN-based prediction algorithms such as in [4] can be trained to predict only one mobility pattern for each user. In particular, to predict the weekly mobility pattern of each user using the traditional ESN approach, the users' context information for an entire

*Typically, the speed of a UAV can reach up to 30 m/s while the average speed of each pedestrian ground user is less than 2 m/s. Therefore, in our model, we ignore the time duration that each UAV uses to change its location.

week need to be used as input of the ESNs that act as one non-linear system. In this conventional ESN approach, it is not possible to separate the users' contexts in a week into several days and train the ESNs to predict the user's mobility in each day with one specific non-linear system. To enable the ESN algorithm to predict the user's mobility pattern and content request distribution with various non-linear systems, the notion of a *conceptor* as defined in [31], is an effective solution that allows characterizing the ESN's reservoir. Conceptors enable an ESN to perform a large number of mobility and content request patterns predictions. Moreover, new patterns can be added to the reservoir of the ESN without interfering with previously acquired ones. For each ESN algorithm, an ESN can record a limited number of history input data due to the echo state property of each ESN. Consequently, the learned pattern will be removed as the recorded input data is updated. Here, we call the ability of recording a limited number of history input data as the *memory* of the ESN's reservoir. The idea of a conceptor can be used to allocate any free memory of an ESN's reservoir to the new learned patterns of the mobility and content request distribution.

Next, we first introduce the components of a conceptor ESN-based prediction algorithm. Then, we formulate the conceptor ESN algorithm to predict the content request distribution and mobility patterns of the users.

A. Conceptor ESN Components

The conceptor ESN-based prediction approach consists of five components: a) agents, b) input, c) output, d) ESN model, and e) conceptor. Since the content request and mobility pattern are user-specific, we design the specific components for the algorithms of the content request distribution and mobility pattern predictions, separately.

1) Content request distribution prediction: The content request distribution prediction algorithm has the following components:

• *Agent*: The agent in our ESNs is the cloud. Since each ESN scheme typically performs a content request distribution prediction for just one user, the cloud's BBUs must implement U conceptor ESN algorithms.

• *Input:* The conceptor ESN takes input as a vector $\boldsymbol{x}_{t,j} = [x_{tj1}, \cdots, x_{tjN_x}]^{\mathrm{T}}$ that represents the context of user j at time t which includes gender, occupation, age, and device type (e.g., tablet or smartphone). Here, N_x is the number of properties that constitute the context information of user j. The vector $\boldsymbol{x}_{t,j}$ is then used to determine the content request distribution $\boldsymbol{y}_{t,j}$ for user j. Note that, the input of the ESNs is the information related to the users' content requests. Our goal is to predict the content request distribution using the context of each user.

• *Output:* The output of the content request distribution prediction ESN at time t is a vector of probabilities $\boldsymbol{y}_{t,j} = [p_{tj1}, p_{tj2}, \ldots, p_{tjN}]$ that represents the probability distribution of content request of user j with p_{tjn} being the probability that user j requests content n at time t.

• *ESN Model:* An ESN model for each user j can find the relationship between the input $\boldsymbol{x}_{t,j}$ and output $\boldsymbol{y}_{t,j}$, thus building the function between the user's context and the content request distribution. Mathematically, the ESN model consists of the output weight matrix $\boldsymbol{W}_j^{\alpha,\text{out}} \in \mathbb{R}^{N \times N_w}$ and the dynamic reservoir containing the input weight matrix $\boldsymbol{W}_j^{\alpha,\text{in}} \in \mathbb{R}^{N_w \times N_x}$, and the recurrent matrix $\boldsymbol{W}_j^{\alpha} \in \mathbb{R}^{N_w \times N_w}$ with N_w being the number of the dynamic reservoir units. For each user j, the dynamic reservoir will be combined with the input $\boldsymbol{x}_{t,j}$ to store the history context of user j. The output weight matrix $\boldsymbol{W}_j^{\alpha,\text{out}}$ with the reservior is trained to approximate the prediction function. The ESN model of user j is initially randomly generated following a uniform distribution. To ensure that the reservoir has the echo state property, $\boldsymbol{W}_j^{\alpha}$ is defined as a sparse matrix with a spectral radius less than one [32].

• *Conceptors:* For content request distribution prediction, we collect the users' context information and the corresponding content requests during the same time slots for different weeks to train one content request distribution. We refer to each content request distribution as one prediction pattern. Given a sequence of the reservoir states $\boldsymbol{v}_j^i = \left[\boldsymbol{v}_{1,j}^i, \ldots, \boldsymbol{v}_{t,j}^i\right]$ with $\boldsymbol{v}_{t,j}^i = \left[v_{t,j1}^i, \ldots, v_{t,jN_w}^i\right]^{\mathrm{T}}$ being the reservoir state of prediction pattern i at time t and the state correlation matrix $\boldsymbol{R}_j^i = \mathbb{E}\left[\boldsymbol{v}_{t,j}^i \left(\boldsymbol{v}_{t,j}^i\right)^{\mathrm{T}}\right]$, the conceptor of prediction pattern i will be [31]:

$$\boldsymbol{M}_j^i = \boldsymbol{R}_j^i \left(\boldsymbol{R}_j^i + \chi^{-2}\boldsymbol{I}\right)^{-1}, \tag{18}$$

where χ is *aperture* defined in [31]. The aperture χ needs to be appropriately set for accurately learning several mobility patterns. When the aperture is small, the reservoir of the ESN slightly changes for learning each new pattern. However, for a large aperture, the reservoir of the ESN changes significantly.

2) Mobility pattern prediction: The components of mobility pattern prediction algorithm are:

• *Agents*: The agents in our conceptor ESNs are the BBUs. Since each ESN scheme typically performs mobility prediction for only one user, the BBUs must also implement U conceptor ESN algorithms.

• *Input:* $\boldsymbol{m}_{t,j} = [m_{tj1}, \cdots, m_{tjN_x+1}]^{\mathrm{T}}$ represents the current location of user j and the context of this user at time t. Using input $\boldsymbol{m}_{t,j}$, the future locations of user j can be predicted.

• *Output:* $\boldsymbol{s}_{t,j} = [s_{tj1}, \cdots, s_{tjN_s}]^{\mathrm{T}}$ represents the predicted locations of user j in the next time slots, where N_s is the number of locations in the next N_s time duration H.

• *ESN Model:* The ESN model of mobility prediction consists of the output weight matrix $\boldsymbol{W}_j^{\text{out}} \in \mathbb{R}^{N_s \times N_w}$, the dynamic reservoir containing the input weight matrix $\boldsymbol{W}_j^{\text{in}} \in \mathbb{R}^{N_w \times N_x+1}$, and the recurrent matrix $\boldsymbol{W}_j \in \mathbb{R}^{N_w \times N_w}$. The generation of the mobility prediction ESN model is similar to the one in the content request distribution prediction case.

• *Conceptors:* For mobility pattern prediction, we consider each user's mobility in each day during one week as one prediction pattern. The expression of the conceptors is the same as the one for the content request distribution given in (18).

B. Conceptor ESN Algorithm for Content and Mobility Predictions

Here, we present the proposed conceptor ESN algorithm to predict the content request distribution and mobility. The pro-

posed algorithm consists of two stages: training and prediction stages.

1) Training Stage: The dynamic reservoir state $\boldsymbol{v}_{t,j}^{i}$ of prediction pattern i for user j at time t which is used to store the states of user j is given by [32]:

$$\boldsymbol{v}_{t,j}^{i} = f\left(\boldsymbol{W}_{j}^{\alpha}\boldsymbol{v}_{t-1,j}^{i} + \boldsymbol{W}_{j}^{\alpha,\text{in}}\boldsymbol{x}_{t,j}\right), \tag{19}$$

where $f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$. Note that, we consider the input and corresponding prediction output as a training data. In this case, we use N_{tr} training data that consists of N_{tr} users' contexts and the corresponding content request to calculate the conceptors and train the output weight matrix $\boldsymbol{W}_{j}^{\alpha,\text{out}}$. Based on N_{tr} training data and (19), the reservoir states before update for each prediction pattern j is $\boldsymbol{v}_{\text{old},j}^{i} = \left[0, \boldsymbol{v}_{1,j}^{i}, \ldots, \boldsymbol{v}_{N_{tr}-1,j}^{i}\right]$ and the updated reservoir states are $\boldsymbol{v}_{j}^{i} = \left[\boldsymbol{v}_{1,j}^{i}, \ldots, \boldsymbol{v}_{N_{tr},j}^{i}\right]$. The matrix $\boldsymbol{v}_{\text{old},j}^{i}$ will be used to train an *input simulation matrix* $\boldsymbol{D}_{j} \in \mathbb{R}^{N_{w}\times N_{w}}$ and $\boldsymbol{v}_{j}^{i}$ will be combined with the updated reservoir states of other prediction patterns to train the output weight matrix.

Then, $\boldsymbol{D}_{j}$ will be combined with output weight matrix $\boldsymbol{W}_{j}^{\alpha,\text{out}}$ to predict the content request distribution pattern for each user. For each added learning pattern i of each user j, the update of $\boldsymbol{D}_{j}$ will be [31]:

$$\boldsymbol{D}_{j} = \boldsymbol{D}_{\text{old},j} + \boldsymbol{D}_{\text{inc},j}^{i}, \tag{20}$$

where $\boldsymbol{D}_{\text{inc},j}^{i} = \left((\boldsymbol{S}\boldsymbol{S}^{\text{T}}/N_{tr} + \chi^{-2}\boldsymbol{I})^{\dagger}\boldsymbol{S}\boldsymbol{T}^{\text{T}}/N_{tr}\right)^{\text{T}}$ with $\boldsymbol{S} = \boldsymbol{F}_{j}^{i-1}\boldsymbol{v}_{\text{old},j}^{i}$ and $\boldsymbol{T} = \boldsymbol{W}_{j}^{\alpha,\text{in}}\boldsymbol{x}_{j}^{i} - \boldsymbol{D}_{\text{old},j}\boldsymbol{v}_{\text{old},j}^{i}$. Here, $\boldsymbol{F}_{j}^{i-1} = \neg \vee \left\{\boldsymbol{M}_{j}^{1}, \ldots, \boldsymbol{M}_{j}^{i-1}\right\}$ is the free memory of the reservoir with $\neg$ and $\vee$ being the boolean operators [31], and $\boldsymbol{x}_{j}^{i} = \left[\boldsymbol{x}_{1,j}^{i}, \ldots, \boldsymbol{x}_{N_{tr},j}^{i}\right]$ is the input sequences of prediction pattern i. During the learning of each pattern i of user j, the conceptor $\boldsymbol{M}_{j}^{i}$ can be computed using (18).

In our proposed ESN algorithm, the output weight matrix $\boldsymbol{W}_{j}^{\alpha,\text{out}}$ is trained in an offline manner using ridge regression [32] to approximate the prediction function which is given by:

$$\boldsymbol{W}_{j}^{\alpha,\text{out}} = \boldsymbol{y}_{j}\boldsymbol{v}_{j}^{\text{T}}\left(\boldsymbol{v}_{j}^{\text{T}}\boldsymbol{v}_{j} + \lambda^{2}\mathbf{I}\right)^{-1}, \tag{21}$$

where $\boldsymbol{v}_{j} = \left[\boldsymbol{v}_{j}^{1}, \boldsymbol{v}_{j}^{2}, \ldots, \boldsymbol{v}_{j}^{N_{M}}\right]^{\text{T}}$ with $\boldsymbol{v}_{j}^{i} = \left[\boldsymbol{v}_{1,j}^{i}, \ldots, \boldsymbol{v}_{N_{tr},j}^{i}\right]$ being the reservoir state sequence of prediction pattern i for user j, λ is the learning rate, and N_{M} being the number of the prediction patterns of each user's content request distribution. In (21), $\boldsymbol{v}_{j}^{i}$ can also be used to calculate the conceptor $\boldsymbol{M}_{j}^{i}$ for prediction pattern i of user j.

2) Prediction Stage: Based on the learning stage, we can use the input simulation matrix $\boldsymbol{D}_{j}$, conceptors $\boldsymbol{M}_{j} = \left[\boldsymbol{M}_{j}^{1}, \ldots, \boldsymbol{M}_{j}^{N_{M}}\right]$, and output weight matrix $\boldsymbol{W}_{j}^{\alpha,\text{out}}$ to obtain the corresponding predictions. In the prediction stage, the reservoir state of pattern i of user j is [31]:

$$\boldsymbol{v}_{t,j}^{i} = \boldsymbol{C}_{j}^{i} f\left(\boldsymbol{W}_{j}^{\alpha}\boldsymbol{v}_{t-1,j}^{i} + \boldsymbol{D}_{j}\boldsymbol{v}_{t-1,j}^{i}\right). \tag{22}$$

From (22), we can see that the conceptor of pattern j, $\boldsymbol{C}_{j}$, controls the update of the reservoir states. By changing the conceptor $\boldsymbol{C}_{j}$, the ESN can predict different patterns in one ESN architecture. The prediction of content request distribution i for user j can be given by:

$$\boldsymbol{y}_{t,j} = \boldsymbol{W}_{j}^{\alpha,\text{out}}\boldsymbol{v}_{t,j}^{i}. \tag{23}$$

TABLE III
PROPOSED CONCEPTOR ESN PREDICTION ALGORITHM

Inputs: N_{tr} training data,
Initialize: $\boldsymbol{W}_{j}^{\alpha,\text{in}}$, $\boldsymbol{W}_{j}^{\alpha}$, $\boldsymbol{W}_{j}^{\alpha,\text{out}}$, $\boldsymbol{y}_{j} = 0$, $\boldsymbol{D}_{j} = 0$.
Training Stage:
for each prediction pattern i **do**.
if reservoir memory space $\boldsymbol{F}_{j}^{i-1} > 0$ **do**.
(a) BBUs collect the states $\boldsymbol{v}_{\text{old},j}^{i}$ and $\boldsymbol{v}_{j}^{i}$ to update $\boldsymbol{D}_{j}$, using (20).
(b) BBUs use the states $\boldsymbol{v}_{j}^{i}$ to calculate the conceptor $\boldsymbol{C}_{j}^{i}$ using (18).
else
(c) increase reservoir matrix $\boldsymbol{W}_{j}^{\alpha}$, re-train all prediction patterns.
end if
end for
(d) BBUs collect states for all patterns $\boldsymbol{v}_{j}$ to train $\boldsymbol{W}_{j}^{\alpha,\text{out}}$, by (21).
Prediction Stage:
(a) BBUs chooses the conceptor to obtain the corresponding reservoir state, using (22).
(b) Get the prediction of content request distribution based on (23) .
Output: Prediction $\boldsymbol{y}_{t,j}$

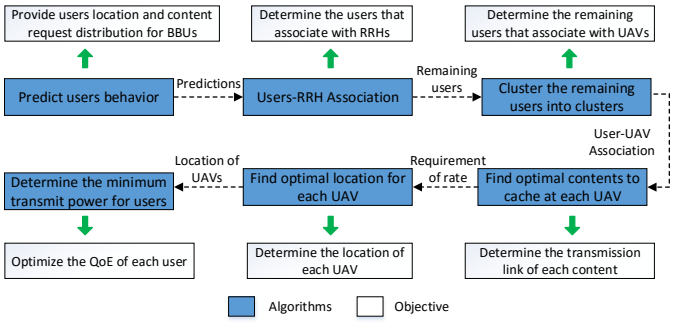


Fig. 2. The procedure used for solving the optimization problem in (17).

From (22) and (23), we can see that the conceptor ESN algorithm exploits an input simulation matrix $\boldsymbol{D}_{j}$ to control the memory of ESN reservoir. The conceptor ESN algorithm for predicting the content request distribution of each user j is shown in Table III.

As shown in Table III, the proposed conceptor ESN algorithm can learn each prediction pattern by a unique non-linear system. This property of the proposed algorithm enables the ESNs to perform the users' behavior predictions using different non-linear systems during different time periods. Furthermore, using the proposed algorithm, one can have the information of the reservoir memory and extract a specific prediction pattern from the learned patterns.

IV. Optimal Location and Content Caching for UAVs

In this section, we use the content request distribution and mobility patterns predictions resulting from the proposed conceptor ESN algorithm in Section III to solve the problem in (17). In our model, a subset of the users selected by the BBUs are connected to the RRHs. The remaining users are clustered into K clusters and each UAV provides service for one cluster. Based on the associations and predictions, we determine which contents to cache at each UAV and find the optimal location of each UAV. Finally, we analyze the implementation and complexity of the proposed algorithm. Fig. 2 summarizes the proposed framework that is used to solve the problem in (17).

A. Users-RRH Association

We find the user-RRH association based on the predicted users' locations at the next time interval. Clearly, the prediction

accuracy of the users' locations will directly affects the users association. A user is associated with RRHs if the following condition is satisfied:

Theorem 1. Given minimum $\bar{D}_{\min}$ and device screen size S_i of each user i, user i will be associated with a cluster k of RRHs if the following rate requirement is satisfied:

$$C^H_{t,qi} \geq \max\left\{\frac{L}{\left(\Delta_\tau - \bar{D}_{\min}\left(\Delta_\tau - \min\left\{\frac{L}{v_F}, \frac{L}{C^{\max}_k}\right\}\right) - \frac{L}{v_{FU}}\right)}, \delta_{S_i,n}\right\}. \tag{24}$$

Proof. See Appendix B. □

From Theorem 1, we can see that the user-RRH association depends on the fronthaul rate of each user, the delay requirement, and the device rate requirement. From (24), we can see that the fronthaul rate of each user decreases as the number of the users associated with the RRHs increases. Clearly, the decrease of the fronthaul rate for each user will improve the delay requirement. Note that, the energy consumption of the RRHs is not considered in our optimization problem as the RRHs can have continuous power supply on the ground while the UAVs are powered by on-board batteries with limited energy. Therefore, it is natural to allow the users to first associate with the RRHs when the RRHs can satisfy the users' QoE requirements.

B. Optimal Content Caching for UAVs

In our model, the remaining users who are not associated with RRHs, will be served by the UAVs. In this case, the users-UAVs associations need to be determined. To this end, we use K-mean clustering approach [24] in which the users are clustered into K groups. By implementing the K-mean clustering approach, the users that are close to each other will be grouped into one cluster. Thereby, each UAV services one cluster and the user-UAV association will be determined. Then, based on the UAV association, we find the optimal contents to cache at each UAV. The content caching will reduce the transmission delay and, hence, decrease the delay requirement. From (15), we can see that, optimal contents to store at the UAV cache lead to maximum reduction of the UAV's transmit power. The reduction of UAV transmit power is caused by the decrease of the delay requirement. Let $\boldsymbol{p}_{j,i} = [p_{j,i1}, p_{j,i2}, \ldots, p_{j,iN}]$ be the content request distribution of user i during period j that consists of H time slots. The optimal contents that will be stored at each UAV cache can be determined based on the following theorem.

Theorem 2. The optimal set of contents $\mathcal{C}_k$ to cache at each UAV k during period T is:

$$\mathcal{C}_k = \arg\max_{\mathcal{C}_k} \sum_{j=1}^{T/H} \sum_{\tau=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} \sum_{n\in\mathcal{C}_k} (p_{j,in}\Delta P_{j,\tau,ki,n}), \tag{25}$$

where $\Delta P_{j,\tau,ki,n} =$

$$\begin{cases} P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)_{n\notin\mathcal{C}_k} - P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)_{n\in\mathcal{C}_k}, C^R_{\tau,ki,n\notin\mathcal{C}_k} \geq \delta_{S_i,n}, \\ P^{\min}_{\tau,ki}(\delta_{S_i,n})_{n\notin\mathcal{C}_k} - P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)_{n\in\mathcal{C}_k}, \delta_{S_i,n} > C^R_{\tau,ki,n\notin\mathcal{C}_k}, \end{cases}$$

with $P^{\min}_{\tau,ki}\left(\boldsymbol{w}_{\tau,t,k}, C^R_{\tau,ki}, n\right)$ being simplified to $P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)$.

Proof. See Appendix C. □

From Theorem 2, we can see that when the fronthaul rates of all users are the same, the transmit power reduction $\Delta P_{j,\tau,ki,n}$ will be a constant. Subsequently, the optimal content caching becomes $\mathcal{C}_k = \arg\max_{\mathcal{C}_k} \sum_{j=1}^{T/H} \sum_{\tau=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} \sum_{n\in\mathcal{C}_k} p_{j,in}$ which corresponds to the result given in [4]. From Theorem 2, we can see that the content caching depends on the pre-knowledge of users association as well as the content request distribution of each user. Therefore, by predicting the mobility pattern and content request distribution for each user, we can determine the optimal content to cache.

C. Optimal Locations of UAVs

Here, we determine the optimal UAVs' locations where the UAVs can serve their associated users using minimum transmit power. Once each UAV selects the suitable contents to cache, the transmission link (BBUs-UAV-user or UAV-user) for each content and the delay requirement $C^R_{\tau,ki,n}$ in (15) are determined. In this case, the rate $\delta^R_{i,n}$ which is used to meet the QoE requirment of each user is also determined. Next, we derive a closed-form expression for the optimal location of UAV k during time slot τ in two special cases.

Theorem 3. To minimize the transmit power of UAV k, the optimal locations of UAV k during time slot τ for cases: a) UAV k positioned at low altitudes compared to the size of its corresponding coverage, $h^2_{\tau,k} \ll (x_{t,i} - x_{\tau,k})^2 + (y_{t,i} - y_{\tau,k})^2$ and $\mu_{\text{NLoS}} = 2$, b) UAV k is placed at high altitudes compared to the size of its corresponding coverage, $h^2_{\tau,k} \gg (x_{t,i} - x_{\tau,k})^2 + (y_{t,i} - y_{\tau,k})^2$, are given by:

$$x_{\tau,k} = \frac{\sum_{i\in\mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} x_{t,i}\psi_{t,ki}}{\sum_{i\in\mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} \psi_{t,ki}}, y_{\tau,k} = \frac{\sum_{i\in\mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} y_{t,i}\psi_{t,ki}}{\sum_{i\in\mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} \psi_{t,ki}}, \tag{26}$$

where $\psi_{t,ki} = \left(2^{\delta^R_{i,n}/B} - 1\right)\sigma^2 10^{(L_{FS}(d_0)+\chi_\sigma)/10}$ with $\sigma = \begin{cases} \sigma_{\text{NLoS}}, & \text{for case } a), \\ \sigma_{\text{LoS}}, & \text{for case } b). \end{cases}$

Proof. See Appendix D. □

Using Theorem 3, we can find the optimal locations of the UAVs given the users association and altitude $h_{\tau,k}$ for the two special cases. For more generic cases, it is highly challenging to find the optimal UAVs' locations using derivation, since the UAV's altitude depends on the x and y coordinates of the UAV. Therefore, we use a learning algorithm given in [33] and [34] to find a sub-optimal solution. The learning algorithm can learn the network state and exploit different actions to adapt the UAV's location according to the network. After the learning step, each UAV will find a sub-optimal location to service the users in a power efficient way.

D. Implementation and Complexity

The complexity of the proposed algorithm pertains to two components: the conceptor ESN algorithm and the optimization algorithm. In the conceptor ESN algorithm, the ESN

needs to be trained and hence, the complexity of the training depends on the users' data. The complexity of implementing the conceptor-based ESN algorithm depends on the number of users and the number of needed predictions. Since predictions occur once every H, during a period T, the ESN algorithm needs to be executed $\frac{T}{H}$ times. Since the ESN needs to predict the users' content request distribution and mobility patterns, the total complexity of the ESN algorithm is $O(U \times 2T/H)$.

The optimization algorithm can be divided into three algorithms: 1) user-UAV association, 2) caching optimization, and 3) optimal location of the UAVs. The complexity of the user-UAV association is $O\left(U^{2K+1} \log U\right)$ [35]. The complexity will be significantly decreased since the number of users that associated with UAVs decreases as the users will first associate with RRHs. For each UAV k, the complexity of the caching optimizatation algorithm depends on the number of the associated users U_k, the number of the contents N and the number of predictions during a period T, $\frac{T}{H}$. Therefore, the complexity of the caching algorithm can be given as $O\left(U_k \times N \times \frac{T}{H}\right)$. However, in practical scenarios, only a few contents have high request probabilities. In this practical case, the complexity of the caching algorithm can be significantly reduced. Finally, for each UAV k, the complexity of the optimal UAV algorithm depends on the number of the associated users U_k and the locations of each associated user during each time slot. Therefore, the complexity of the optimal UAV location algorithm is $O\left(U_k \times F\right)$.

V. Simulation Results

For our simulations, the content request data that the ESN uses to train and predict content request distribution is obtained from *Youku* of *China network video index*[†]. Here, one circular CRAN area with a radius $r = 500$ m is considered with $U = 70$ uniformly distributed users and $R = 20$ uniformly distributed RRHs. The detailed parameters are listed in Table V. Actual pedestrian mobility data is measured from 100 students at the *Beijing University of Posts and Telecommunications*. We recorded the daily mobility pattern of each student and collected their locations every hour during 9:00 am - 12:00 pm in a period of over two months. For comparison purposes, we investigate: a) optimal algorithm that has a prior knowledge of the accurate user's mobility patterns and content request distribution, b) ESN algorithm in [4] to predict the content request distribution and mobility pattern, and c) random caching with ESN algorithm in [4] to predict content request distribution. All statistical results are averaged over 5000 independent runs. The accuracy of ESN prediction is measured by normalized root mean square error [31].

Fig. 3 shows how the memory of the conceptor ESN reservoir changes as the number of the mobility patterns that were learned by the conceptor ESN varies. Here, one mobility pattern represents the users' trajectory in one day and the colored region represents the memory used by the conceptor ESN. In Fig. 3, we can see that the memory usage increases as the number of the learned mobility patterns increases. This is due to the fact the conceptor ESN uses a limited memory to learn mobility patterns. From Fig. 3, we can also see that the conceptor ESN uses less memory for learning mobility pattern 2 compared to pattern 6. In fact, mobility pattern 2 is similar to mobility pattern 1, and, hence, the conceptor ESN requires only a small amount of memory to learn mobility pattern 2. However, the conceptor ESN needs to use more memory to learn mobility pattern 6. Clearly, when a new mobility pattern needs to be learned, the proposed approach needs to learn the difference between the learned mobility patterns and the new one.

[†]The data is available at http://index.youku.com/.

TABLE IV
SYSTEM PARAMETERS

Parameter	Value	Parameter	Value	Parameter	Value
F	1000	Y	0.13	P_B	30 dBm
X	11.9	N	25	P_R	20 dBm
$\chi_{\sigma_{\text{LoS}}}$	5.3	H	10	$P_{\max}$	20 W
N_{tr}	1000	d_0	5 m	σ^2	-95 dBm
N_s	12	λ	0.01	$h_{\min}$	100 m
N_x	4	β	2	B	1 MHz
μ_{LoS}	2	μ_{NLoS}	2.4	$\delta_{S_i,n}$	5 Mbit/s
χ	15	ζ_1	0.5	f_c	38 GHz
$\chi_{\sigma_{\text{NLoS}}}$	5.27	η	100	B_v	1 GHz
K	5	C	1	L	1 Mbit
T	120	ζ_2	0.5	N_w	1000

In Fig. 4, we show the variations of two content request probabilities of a selected user during one day. The user is randomly chosen from the set of users in the network. From Fig. 4, we can see that, the probability with which this user requests content 1 decreases during working hours (9:00-11:00 and 14:00-18:00) and increases at all other times. Similarly, the request probability of content 2 increases during working hours and decreases during the rest of the time. This is due to the fact that content 1 is an entertainment content while content 2 is a work-related content. Fig. 4 also shows that the sum of the probability with which this user requests content 1 and content 2 exceeds 0.5 during each hour. This is because the user always requests a small amount of contents during one day.

Fig. 5 shows how the total transmit power of the UAVs in a time period changes as the number of the users varies. In Fig. 5, we can see that the total UAV transmit power of all algorithms increases as the number of the users increases. This is due to the fact that the number of the users associated with the RRHs and the capacity of the wireless fronthaul link of UAVs are limited. Therefore, the UAVs need to increase their transmit power to satisfy the QoE requirement of each user. From Fig. 5(a), we can also see that the proposed approach can reduce the total transmit power of the UAVs of about 16.7% compared to the ESN algorithm used to predict the content request and mobility for a network with 70 users. This is because the conceptor ESN that separates the users' behavior into multiple patterns and uses the conceptor to learn these patterns, can predict the users' behavior more accurately compared to the ESN algorithm. Fig. 5(b) shows that the proposed algorithm can yield, respectively, 33.3% and 20% gains with respect to reducing the total transmit power compared to the proposed algorithm without cache and the proposed algorithm without optimizing the UAVs' locations for a network with 80 users.

Fig. 6 shows the rate needed for satisfying the QoE requirement of each user versus the wireless fronthaul rate of each user. In this figure, the black and blue lines represent,

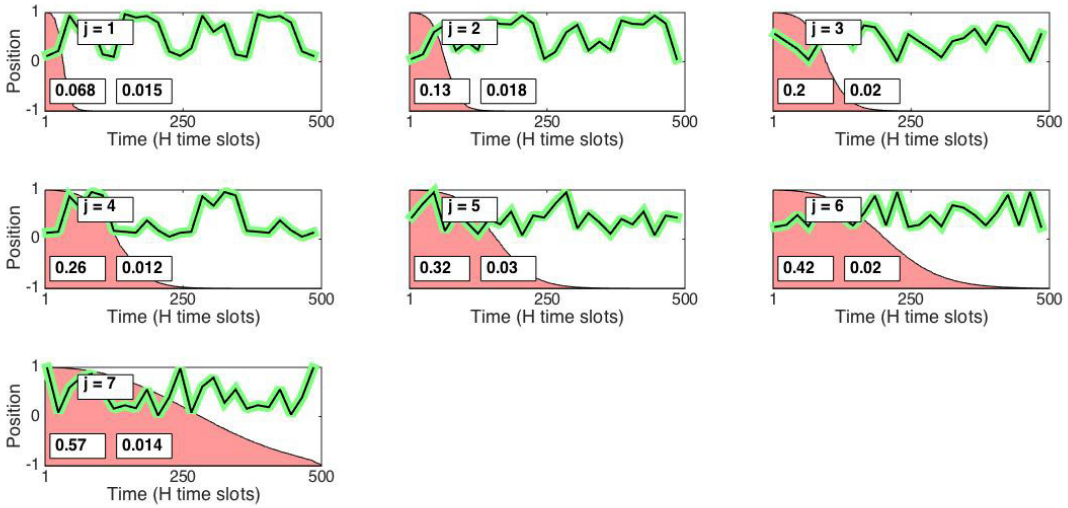


Fig. 3. Mobility patterns predictions of Conceptor ESN algorithm. In this figure, the green curve represents the conceptor ESN prediction, the black curve is the real positions, top rectangle j is the index of the mobility pattern learned by ESN, the legend on the bottom left shows the total reservoir memory used by ESN and the legend on the bottom right shows the normalized root mean square error of each mobility pattern prediction.

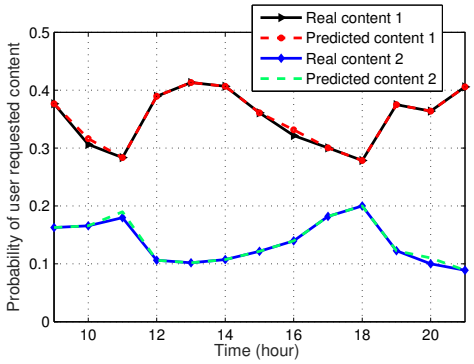
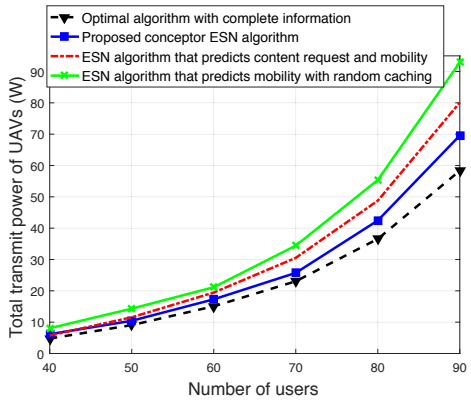
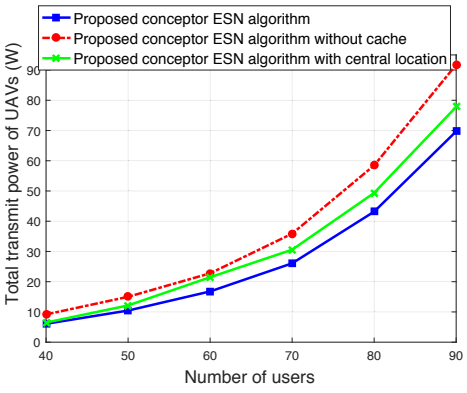


Fig. 4. Content request probability predictions.



(a)



(b)

Fig. 5. Total transmit power as the number of users varies. ($K = 5$ and $C = 1$.)

respectively, the rate requirement of a user that receives a content from the UAV cache and the BBUs. In Fig. 6, we can see that the rate required to maximize the users QoE of BBUs-UAV-user link decreases as the wireless fronthaul rate increases. However, the rate needed to maximize the user's QoE of UAV-user link not change when the fronthaul rate varies. Clearly, the use of caching at the UAVs can significantly reduce the rate required to reach the QoE threshold of each user when the wireless fronthaul rate for each user is low.

In Fig. 7, we show how the percentage of users with satisfied QoE requirement changes as the number of the users varies. From Fig. 7, we can see that the percentage of the satisfied users decreases as the number of the users increases. However, using the proposed approach, the QoE remains maximum for all number of users when the number of the users increases from 30 to 70. In particular, the proposed algorithm can yield a gain of 59.6% gain in terms of the percentage of the users with satisfied QoE compared to the proposed algorithm without UAVs for the network with 120 users. This is due to the fact that the UAVs can maximize the users' QoE when the RRHs are not able to satisfy the QoE requirements.

In Fig. 8, we show how the average minimum transmit power of UAVs changes as the number of the UAVs varies. From Fig. 8, we can see that the average minimum transmit power of each UAV decreases as the number of the UAVs increases. In particular, using the proposed algorithm, the average transmit power of the UAVs decreases by 86% when the number of UAVs increases from 3 to 7. This is due to the fact that for a higher number of UAVs the number of users associated with each UAV decreases, and, hence, the average transmit power per UAV also decreases. As shown in Fig. 8, the proposed approach becomes closer to the optimal one as the number of UAVs increases. The reason is that the location prediction error is higher for a lower number of UAVs (or

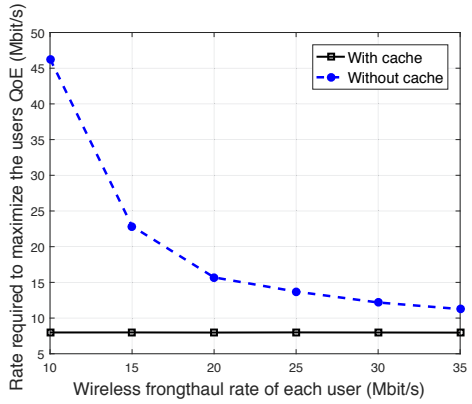


Fig. 6. Rate required to maximize the users QoE as the fronnthaul rate of each user changes.

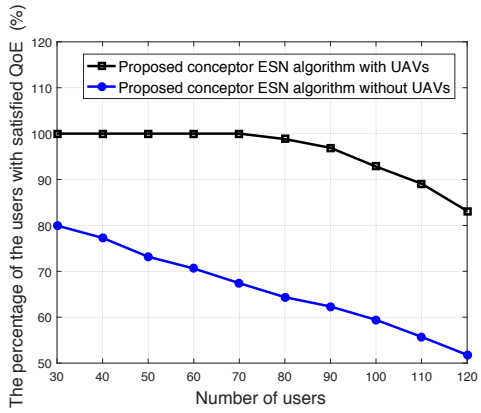


Fig. 7. The percentage of the users QoE that is maximized as the number of users varies

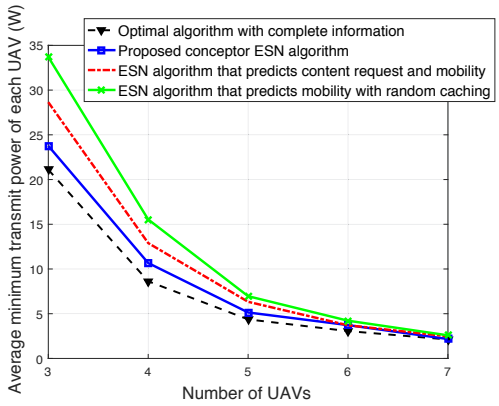


Fig. 8. Average minimum transmit power as the number of UAVs changes ($U = 70$ and $C = 1$).

equivalently the clusters).

Fig. 9 shows the total transmit power of the UAVs as a function of the number of the contents stored at the UAV cache. As shown in Fig. 9, the total transmit powers of all considered algorithms increase as the number of storage units increases. The reason is that the probability that the requested contents of the users are stored at the UAV cache increases, and, consequently, the UAV will directly transmit the requested contents to the users. Fig. 9 also shows that the ESN approach that predicts the content request and mobility can yield up to 49% power reduction compared to the ESN approach that

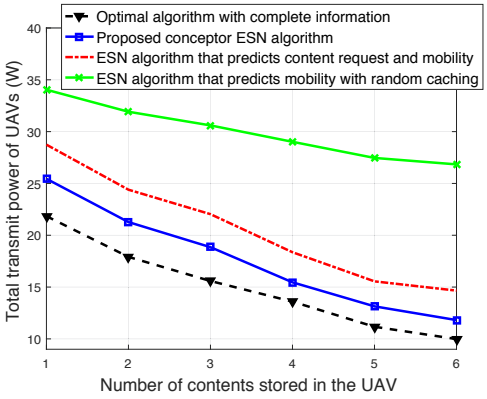
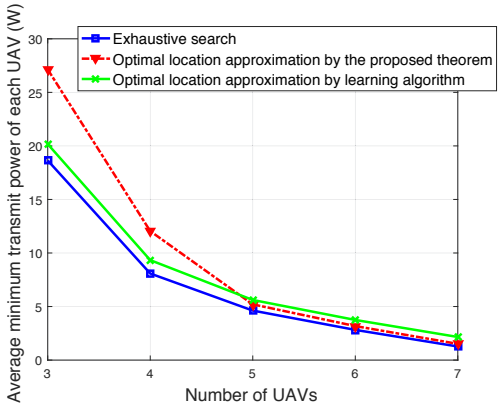
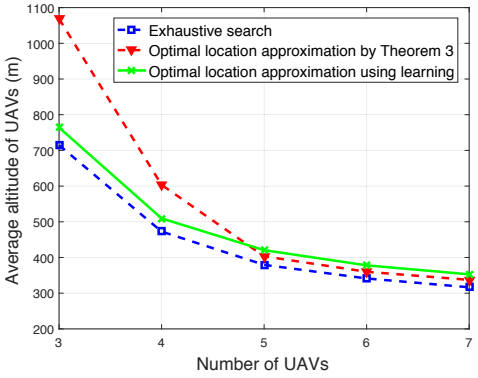


Fig. 9. Total transmit power as the number of the contents stored in a UAV cache varies ($U = 70$ and $K = 5$).



(a)



(b)

Fig. 10. Average minimum transmit power and average altitude vs. the number of UAVs.

predicts the mobility with the random caching scheme.

In Fig. 10, we show how the average transmit power and average altitude of the UAVs change as the number of UAVs varies. In this case, we compare the result of our proposed approach with the optimal result obtained by an exhaustive search method. For the learning algorithm, the interval of the neighboring action of each coordinate is 3 m. Figs. 10(a) shows that the optimal location of the UAV approached by Theorem 3 has only 6.4% deviation compared to the exhaustive search. Furthermore, as shown in Fig. 10(b), by increasing the number of UAVs from 3 to 7, the average altitude of the UAVs decreases from 1080 m to 332 m in

the proposed algorithm case. This is due to the fact that for a higher number of the UAVs, each UAV needs to provide coverage for a smaller area and, hence, it can be deployed at a lower altitude. From Figs. 10(a) and 10(b), we can also see that, as the number of the UAVs increases, the result of Theorem 3 approaches the optimal solution that is obtained by the exhaustive search. This is due to the fact that for a higher number of UAVs, the coverage area of each UAV decreases, and, hence, the approximation condition in Theorem 3 will hold with a tighter bound.

VI. Conclusions

In this paper, we have proposed a novel framework that uses flying UAVs to provide service for the mobile users in a CRAN system. First, we have presented an optimization problem that seeks to guarantee the QoE rquirement of each user using the minimmum transmit power of the UAVs. Next, to solve this problem, we have developed a novel algorithm based on the echo state networks and concepters. The proposed algorithm allows predicting the content request distribution of each user with limited information on the network state and user context. The proposed algorithm also enables the ESNs separate the users behavior into several patterns and learn these patterns with various non-linear systems. Simulation results have shown that the proposed approach yields significant performance gains in terms of minimum transmit power compared to conventional ESN approaches.

Appendix

A. Proof of Proposition 1

From (10), we can see that the delay of link (b) is larger than that of link (c) and the minimum delay of the link (a) is L/v_F. Hence, we only need to consider the delay values between L/v_{FU} and $L/C^{\mathrm{V}}_{\tau,ki}$, $k \in \mathcal{K}$. To maximize $C^{\mathrm{V}}_{\tau,ki}$, we consider $d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right) = h$, and $P_{t,ki} = P_{\max}$. Then, the rate of the UAV-user link $C^{\mathrm{V}}_{\tau,ki}$ is given by:

$$\begin{aligned} C^{\mathrm{V}}_{\tau,ki} &= B_V \log_2\left(1+\frac{P_{t,ki}}{10^{\bar{l}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k},\boldsymbol{w}_{\tau,t,i}\right)/10}\sigma^2}\right) \\ &\leq B_V \log_2\left(1+\frac{P_{\max}}{10^{l^{\mathrm{LoS}}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k},\boldsymbol{w}_{\tau,t,i}\right)/10}\sigma^2}\right) \\ &\overset{(a)}{\leq} B_V \log_2\left(1+\frac{P_{\max}}{10^{\left(L_{FS}(d_0)+10\mu_{\mathrm{LoS}}\log(h)-4\sigma_{\mathrm{LoS}}\right)/10}\sigma^2}\right), \quad (27) \end{aligned}$$

where (a) follows from the fact that with a probability close to one (greater than 99.99%), the Gaussian random variable $\chi_{\sigma_{\mathrm{LoS}}}$ will have a value larger than $-4\sigma_{\mathrm{LoS}}$. From (27), we can see that, as h increases, the capacity $C^{\mathrm{V}}_{\tau,ki}$ decreases. Therefore, we set $h = h_{\min}$. This completes the proof.

B. Proof of Theorem 1

Based on (12) and $\bar{D}_{\min}$, the delay is $D_{\tau,i,n} = \Delta_\tau - \bar{D}_{\min}\left(\Delta_\tau - \min\left\{\frac{L}{v_F}, \frac{L}{C^{\max}_k}\right\}\right)$, and, hence, the delay requirement for RRH cluster q transmitting content n to user i during time slot τ will be:

$$C^R_{\tau,qi} = \frac{L}{\Delta_\tau - \bar{D}_{\min}\left(\Delta_\tau - \min\left\{\frac{L}{v_F}, \frac{L}{C^{\max}_k}\right\}\right) - \frac{L}{v_{FU}}}. \quad (28)$$

Therefore, the delay requirement during each interval is equal to $C^R_{\tau,qi}$. Since the device rate requirement is $\delta_{S_i,n}$, the rate of RRH cluster q transmitting content n to user i, $C^{\mathrm{H}}_{\tau,qi}$ must satisfy $C^{\mathrm{H}}_{t,qi} \geq \max\left\{C^R_{\tau,qi}, \delta_{S_i,n}\right\}$. This completes the proof.

C. Proof of Theorem 2

Since the delay requirement $C^R_{\tau,ki,n}$ depends on the contents at the UAV cache, it can be written as $\delta^R_{i,n} = \max\left\{C^R_{\tau,ki,n(n\in\mathcal{C}_k)}, C^R_{\tau,ki,n(n\notin\mathcal{C}_k)}, \delta_{S_i,n}\right\}$. Let $P^{\min}_{\tau,ki} = \sum_{t=1}^{F_{\tau,i}} P^{\min}_{j,\tau,t,ki}$. Then the reduction of UAV transmit power by content caching during time slot τ of period j will be:

$$\Delta P_{j,\tau,ki,n} = \begin{cases} P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)_{n\notin\mathcal{C}_k} - P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)_{n\in\mathcal{C}_k}, C^R_{\tau,ki,n\notin\mathcal{C}_k} \geq \delta_{S_i,n}, \\ P^{\min}_{\tau,ki}(\delta_{S_i,n})_{n\notin\mathcal{C}_k} - P^{\min}_{\tau,ki}\left(C^R_{\tau,ki}\right)_{n\in\mathcal{C}_k}, \quad \delta_{S_i,n} > C^R_{\tau,ki,n\notin\mathcal{C}_k}. \end{cases}$$

Considering the fact that the content request distribution changes once every H time slots, the power minimization problem for UAV k during a period that consists of H time slots is:

$$\begin{aligned} &\min_{\mathcal{C}_k} \sum_{\tau=1}^{T} \sum_{i\in\mathcal{U}_{\tau,k}} P^{\min}_{\tau,ki} \min_{\mathcal{C}_k} = \sum_{j=1}^{T/H} \sum_{\tau_j=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} P^{\min}_{\tau_j,ki} \\ &= \min_{\mathcal{C}_k} \sum_{j=1}^{T/H} \sum_{\tau=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} P^{\min}_{j,\tau,ki} \overset{(a)}{\Leftrightarrow} \max_{\mathcal{C}_k} \sum_{j=1}^{T/H} \sum_{\tau=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} \Delta P_{j,\tau,ki,n}, \\ &\overset{(b)}{=} \max_{\mathcal{C}_k} \sum_{j=1}^{T/H} \sum_{\tau=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} \left(\sum_{n\in\mathcal{C}_k} (p_{j,in}\Delta P_{j,\tau,ki,n}) + \sum_{n\notin\mathcal{C}_k} (p_{j,in}\Delta P_{j,\tau,ki,n}) \right), \\ &= \max_{\mathcal{C}_k} \sum_{j=1}^{T/H} \sum_{\tau=1}^{H} \sum_{i\in\mathcal{U}_{\tau,k}} \sum_{n\in\mathcal{C}_k} (p_{j,in}\Delta P_{j,\tau,ki,n}), \end{aligned}$$

where (a) follows the fact that minimizing the transmit power of the UAVs is equivalent to maximizing the reduction of the UAVs' transmit power caused by caching, and (b) is obtained by computing the average power reduction using content request probability distribution of each user. This completes the proof.

D. Proof of Theorem 3

At very low altitudes, $h^2_{\tau,k} \ll (x_{t,i} - x_{\tau,k})^2 + (y_{t,i} - y_{\tau,k})^2$, $\frac{h_{\tau,k}}{d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k},\boldsymbol{w}_{\tau,t,i}\right)} \approx 0$ leading to $\phi_t = 0^\circ$, and, hence, $\Pr\left(l^{\mathrm{NLoS}}_{t,ki}\right) = 1$. Thus, we have $\bar{l}_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right) = l^{\mathrm{NLoS}}_{t,ki}$ and (16) can be rewritten as $P^{\min}_{\tau,t,ki} = \left(2^{\delta^R_{i,n}/B} - 1\right)\sigma^2 10^{\left(L_{FS}(d_0)+\chi_{\sigma_{\mathrm{NLoS}}}\right)/10} d_{t,ki}\left(\boldsymbol{w}_{\tau,t,k}, \boldsymbol{w}_{\tau,t,i}\right)^{\mu_{\mathrm{NLoS}}}$.

Now, we find the optimal location $(x_{\tau,k}, y_{\tau,k})$ of UAV k during time slot τ in order to minimize $\sum_{i\in\mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} P^{\min}_{\tau,t,ki}$. In this case, the derivation of $\sum_{i\in\mathcal{U}_{\tau,k}} \sum_{t=1}^{F_{\tau,i}} P^{\min}_{\tau,t,ki}$ with respect to

$x_{\tau,k}$ is given by:

$$\frac{\partial \sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} P^{\min}_{\tau,t,ki}}{\partial x_{\tau,k}} = \frac{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} \partial P^{\min}_{\tau,t,ki}}{\partial x_{\tau,k}} = \sum_{i\in\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}} \mu_{\text{NLoS}}(x_{\tau,k}-x_{t,i})\psi_{t,ki}\left((x_{\tau,k}-x_{t,i})^2+(y_{\tau,k}-y_{t,i})^2+h^2_{\tau,k}\right)^{\frac{\mu_{\text{NLoS}}}{2}-1}. \tag{29}$$

As $\mu_{\text{NLoS}}=2$, (29) is simplified to $\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} 2(x_{\tau,k}-x_{t,i})\,\psi_{t,ki}=0$. As a result, $x_{\tau,k} = \frac{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} x_{t,i}\psi_{t,ki}}{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} \psi_{t,ki}}$. Likewise, we can show that $y_{\tau,k} = \frac{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} y_{t,i}\psi_{t,ki}}{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} \psi_{t,ki}}$.

For case b), since $h^2_{\tau,k} \gg (x_{t,i}-x_{\tau,k})^2+(y_{t,i}-y_{\tau,k})^2$, $d_{t,ki}(\boldsymbol{w}_{\tau,t,k},\boldsymbol{w}_{\tau,t,i}) \approx h_{\tau,k}$ and, hence, $\frac{h_{\tau,k}}{d_{t,ki}(\boldsymbol{w}_{\tau,t,k},\boldsymbol{w}_{\tau,t,i})} \approx 1 \rightarrow \phi_t = 90^\circ$. Consequently, $\Pr\left(l^{\text{LoS}}_{t,ki}\right)=1$. Then, we have $\bar{l}_{t,ki}(\boldsymbol{w}_{\tau,t,k},\boldsymbol{w}_{\tau,t,i}) = l^{\text{LoS}}_{t,ki}$. The derivation of $\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} P^{\min}_{\tau,t,ki}$ will be:

$$\begin{aligned}\frac{\partial \sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} P^{\min}_{\tau,t,ki}}{\partial x_{\tau,k}} &= \sum_{i\in\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}} \mu_{\text{LoS}}(x_{\tau,k}-x_{t,i})\psi_{t,ki} \\ &\quad\times\left((x_{\tau,k}-x_{t,i})^2+(y_{\tau,k}-y_{t,i})^2+h^2_{\tau,k}\right)^{\frac{\mu_{\text{LoS}}}{2}-1} \\ &\approx \sum_{i\in\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}} \mu_{\text{LoS}}(x_{\tau,k}-x_{t,i})\,\psi_{t,ki}h^{\mu_{\text{LoS}}-2}_{\tau,k}=0.\end{aligned}$$

As a result, $x_{\tau,k} = \frac{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} x_{t,i}\psi_{t,ki}}{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} \psi_{t,ki}}$ and $y_{\tau,k} = \frac{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} y_{t,i}\psi_{t,ki}}{\sum\limits_{i\in\mathcal{U}_{\tau,k}}\sum\limits_{t=1}^{F_{\tau,i}} \psi_{t,ki}}$. This completes the proof.

REFERENCES

[1] S. Chen, F. Qin, B. Hu, X. Li, and Z. Chen, "User-centric ultra-dense networks for 5g: Challenges, methodologies, and directions," *IEEE Wireless Communications*, vol. 23, no. 2, pp. 78–85, April 2016.

[2] M. Peng, Y. Sun, X. Li, Z. Mao, and C. Wang, "Recent advances in cloud radio access networks: System architectures, key techniques, and open issues," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 3, pp. 2282–2308, Thirdquarter 2016.

[3] B. B. Nagaraja and K. G. Nagananda, "Caching with unknown popularity profiles in small cell networks," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, San Diego, CA, USA, Dec. 2015.

[4] M. Chen, W. Saad, C. Yin, and M. Debbah, "Echo state networks for proactive caching in cloud-based radio access networks with mobile users," *available online: arxiv.org/abs/1607.00773*, July. 2016.

[5] J. Qiao, Y. He, and S. Shen, "Proactive caching for mobile video streaming in millimeter wave 5G networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 10, pp. 7187–7198, Oct. 2016.

[6] T. X. Tran and D. Pompili, "Octopus: A cooperative hierarchical caching strategy for radio access networks," *available online: arxiv.org/abs/1608.00067*, July 2016.

[7] Y. Guo, L. Duan, and R. Zhang, "Cooperative local caching under heterogeneous file preferences," *IEEE Transactions on Communications*, to appear, 2016.

[8] E. Bastug, M. Bennis, M. Kountouris, and M. Debbah, "Cache-enabled small cell networks: Modeling and tradeoffs," *EURASIP J. Wireless Commun. Netw.,Special Issue Tech. Adv. Design Deployment Future Heterogeneous Netw.*, vol. 2015, no. 1, Feb 2015.

[9] Z. Ye, C. Pan, H. Zhu, and J. Wang, "Tradeoff caching strategy of outage probability and fronthaul usage in Cloud-RAN," *available online: arxiv.org/abs/1611.02660*, Nov. 2016.

[10] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Transactions on Wireless Communications*, vol. 15, no. 6, pp. 3949–3963, June. 2016.

[11] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for UAV-enabled mobile relaying systems," *IEEE Transactions on communications*, vol. 64, no. 12, pp. 4983–4996, Dec. 2016.

[12] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, Aug. 2016.

[13] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Austin, TX, USA, Dec. 2014.

[14] I. Bor-Yaliniz and H. Yanikomeroglu, "The new frontier in RAN heterogeneity: Multi-tier drone-cells," *IEEE Communications Magazine*, vol. 54, no. 11, pp. 48–55, Nov. 2016.

[15] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3D placement of drone base stations in wireless cellular networks," in *Proc. of IEEE Vehicular Technology Conference*, May 2016.

[16] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile Internet of Things: Can UAVs provide an energy-efficient mobile architecture?," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, USA, Dec. 2016.

[17] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Drone small cells in the clouds: Design, deployment and performance analysis," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, San Diego, CA, USA, Dec. 2015.

[18] N. T. Nguyen, Y. Wang, H. Li, X. Liu, and Z. Han, "Extracting typical users' moving patterns using deep learning," in *Proc. of IEEE Global Communication Conference (GLOBECOM)*, Anaheim, CA, USA, Dec. 2012.

[19] J. K. Lee and J. C. Hou, "Modeling steady-state and transient behaviors of user mobility: formulation, analysis, and application," in *proc. of ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2006, pp. 85–96.

[20] X. Wang, L. Duan, and R. Zhang, "User-initiated data plan trading via a personal hotspot market," *IEEE Transactions on Wireless Communications*, vol. 15, no. 11, pp. 7885–7898, Nov. 2016.

[21] C. Song, Z. Qu, N Blumm, and A. L. Barabasi, "Limits of predictability in human mobility.," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.

[22] E. Bastug, M. Bennis, E. Zeydan, M. A. Kader, I. A. Karatepe, A. S. Er, and M. Debbah, "Big data meets telcos: A proactive caching perspective," *Journal of Communications and Networks*, vol. 17, no. 6, pp. 549–557, Dec. 2015.

[23] D. A. Soysa, D. G. Chen, O. C. Au, and A. Bermak, "Predicting YouTube content popularity via Facebook data: a network spread model for optimizing multimedia delivery," in *Proc. of IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, Singapore, April. 2013.

[24] F. Hoppner and F. Klawonn, *Clustering with Size Constraints*, Springer Berlin Heidelberg, 2008.

[25] T. Yoo and A. Goldsmith, "On the optimality of multiantenna broadcast scheduling using zero-forcing beamforming," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 528–541, March. 2006.

[26] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Upper Saddle River, NJ: Prentice-Hall, 2002.

[27] T. S. Rappaport, F. Gutierrez, E. Ben-Dor, J. N. Murdock, Y. Qiao, and J. I. Tamir, "Broadband millimeter-wave propagation measurements and models using adaptive-beam antennas for outdoor urban cellular communications," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 4, pp. 1850–1859, April. 2013.

[28] O. Somekh, O. Simeone, Y. Bar-Ness, A. M. Haimovich, and S. Shamai, "Cooperative multicell zero-forcing beamforming in cellular downlink

channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3206–3219, June. 2009.

[29] K. Mitra, A. Zaslavsky, and C. Ahlund, "Context-aware QoE modelling, measurement and prediction in mobile computing systems," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 920–936, Dec. 2015.

[30] J. Herbert and H. Harald, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.

[31] H. Jaeger, "Controlling recurrent neural networks by conceptors," *available online: arxiv.org/abs/1403.3369*, 2014.

[32] M. Lukoševicius, *A Practical Guide to Applying Echo State Networks*, Springer Berlin Heidelberg, 2012.

[33] M. Bennis, S.M. Perlaza, P. Blasco, Z. Han, and H.V. Poor, "Self-organization in small cell networks: A reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 12, no. 7, pp. 3202–3212, June. 2013.

[34] M. Chen, W. Saad, and C. Yin, "Echo state networks for self-organizing resource allocation in LTE-U with uplink-downlink decoupling," *IEEE Transactions on Wireless Communications*, vol. 1, no. 1, Jan. 2017.

[35] M. Inaba, N. Katoh, and H. Imai, "Applications of weighted voronoi diagrams and randomization to variance-based k-clustering," in *Pro. of Annual Symposium on Computational Geometry*, 1994.