Millimeter Wave meets Edge Computing for Mobile VR with High-Fidelity 8K Scalable 360° Video

Sabyasachi Gupta*, Jacob Chakareski*, Petar Popovski†
*The University of Alabama, Tuscaloosa, US, †Aalborg University, Denmark

***Abstract*—We investigate a novel multiple user scalable 8K 360° video mobile virtual reality arcade streaming system that enables high reliability and immersion f delity, and low interactive latency, by the synergistic integration of scalable 360° content, expected VR user viewport modeling, millimeter wave (mmWave) communication and network edge computation capability. The high data rate mmWave link is used to transmit the video content of the expected user 360 viewport at enhanced quality. To compensate for the dynamic nature of mmWave links and prospective expected viewport characterization error, we integrate a fall back transmission based on Wi-Fi broadcast of a baseline representation of the 360 panorama to all users. In our proposed transmission strategy, the expected viewport content can be sent as raw or encoded at different qualities, which enhances the end-to-end performance, by exploiting effective trade-offs between communication and computation latency at the receiving user. With the aim of maximizing the minimum VR immersion f delity across all users, we investigate the joint optimization of the mmWave access point (AP) to user association, the data rate for the encoded portion of the 360 viewport content that is to be transmitted, and computation resource allocation. Our experimental results demonstrate that the proposed system can achieve signif cant improvement in delivered VR user immersion f delity and quality of experience relative to a state-of-the-art reference method that leverages Wi-Fi transmission only.**

I. Introduction

Virtual reality (VR) technologies are increasingly popular, as they can enable a range of societal applications, spanning education and training, health-care, telepresence and remote work, search and rescue, disaster relief, etc. It is anticipated that VR will represent a 120 billion market by 2022 [1]. 360° video is an integral part of Virtual Reality. Therefore, the delivery of high quality 360° video is extremely important to adopt VR widely. 360° videos can provide an immersive experience by immersing viewers at the center of a remote scene. The latency in 360° video streaming system needs to be restricted to 10-20 milliseconds. Relative to traditional videos, 360° require much more bandwidth to deliver the entire scene. Presently, only low-quality low-resolution 360 videos can be streamed over wired networks [2], [3].

A user, at any moment can watch only a limited region of the 360 spatial panorama determined by the f eld of view (FOV) of the VR head mounted display (HMD) of the user and his navigation actions. VR server may predict the future viewport of the user. Therefore, only a part of the 360 video content which covers the predicted future viewport of the user is delivered by the VR server to the user. Once the encoded tiles of the video are received at the user, they need to be decoded and rendered at the processing unit of the VR devices. The PC-based VR devices, e.g., Oculus Rift [2], HTC Vive [3], are tethered to a powerful gaming computer and perform decoding locally on the computer/console. Therefore, these devices lack portability. Mobile VR, untethered from a PC/console, but attached to a smartphone, such as Samsung Gear VR [4], Google Daydream [5], etc., has portability. However, it exhibits limited computation capability which may result in high computation delay and therefore negatively affect the user experience.

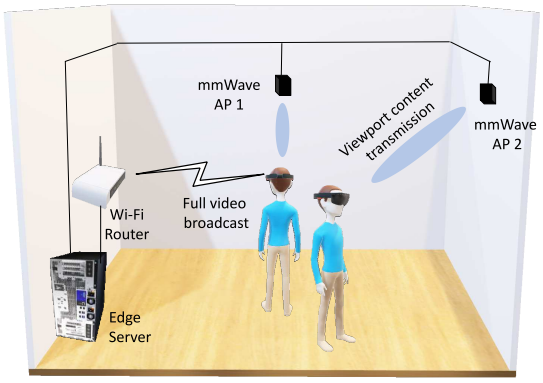

Fig. 1. The proposed next generation VR arcade system that we investigate.

In this paper, we investigate a novel streaming system for a next generation VR arcade that enables high VR immersion f delity, and low interactive latency via a novel integration of high f delity 8K 360-degree scalable video, mobile-edge computing (MEC) and mmWave. By enabling high computation capability at the network edge and high transmission rate using mmWave technology, it becomes possible to decode part of the 360 video content at the server and transmit the raw data using mmWave with low overall system delay. This approach helps to reduce the decoding delay of the 360 video content at the user and improves the system eff ciency, in the mobile setting, where the user device has inferior computing capabilities. In our system, each user is assigned an mmWave AP and the video content of the expected user 360 viewport is transmitted using high data rate mmWave link at enhanced quality. Among the 360 video content which is sent through mmWave link to an user, a part of the video content is sent as a raw data while the remaining part is sent encoded. To compensate for the uncertainty of mmWave links and prospective expected viewport characterization error, a baseline representation of the 360 panorama is broadcasted to all users using Wi-Fi. Our system is illustrated in Fig. 1. We investigate the joint optimization of the mmWave AP to user assignment, data rate for the transmission of the encoded 360 content, and computation resource allocation. The aim is

This work has been supported in part by NSF Awards CCF-1528030, ECCS-1711592, CNS-1836909, and CNS-1821875.

978-1-7281-1817-8/19/$31.00

to maximize the minimum VR immersion fidelity across all users, subject to overall latency constraints, and computational resource constraints at the edge server and mobile user devices.

The main contributions of this work is as follows: (1) In our proposed transmission scheme, efficient use of the available computational and communication resource is enabled, due to the synergistic integration of scalable 360 video content with MEC and mmWave communication, (2) We have analyzed the CPU cycles required to decode the tiles of the 360° video for different data rate, and modeled CPU cycles for a tile as polynomial function of data rate, (3) We introduce rate-distortion analysis to characterize analytical problem under investigation. (4) Our experimental results demonstrate that the proposed 360° VR streaming system can enable significant improvement in delivered VR immersion fidelity and quality of experience relative to a state-of-the-art reference method that employs Wi-Fi transmission only.

Only two related studies [6, 7] have appeared to date that examined the integration of mmWave and edge computing for mobile VR, however, with very limited contributions. In [6], a user clustering strategy is investigated to maximize the user FoV frame request admission. To minimize the traffic volume for VR gaming, proactive computing and caching of synthesized interactive VR video frames have been proposed in [7]. However, the proposed heuristic strategies in [6, 7] do not necessarily enable improvement in the delivered user immersion fidelity, which is a critical factor in deciding the quality of experience in mobile VR. Also, only low-quality 4K video content delivery is considered in these studies. Hence, the delivered quality of experience is low. The present paper considerably advances the state-of-the-art since we rigorously analyze system design parameters, e.g., the allocation of computing resources and video tile data rates, based on their rate-distortion characteristics, with the objective to maximize the delivered VR user immersion fidelity.

II. VR Arcade System Model

Fig. 1 illustrates the VR arcade system model under consideration. There exist $U = \{1, 2, .., M\}$ set of mobile VR users and the users can receive a 360° video with wireless VR headset. There are $A = \{1, 2, .., N\}$ set of mmWave AP and a Wi-Fi router with an edge cloud available in the network. The 360 video content is streamed to the users using mmWave communication and Wi-Fi. The 360 video to be streamed to all the users $u \in U$, is encoded at frame rate $1/T_f$ where T_f is the time between frames. The frames from the spherical video are unwrapped into a 2D projection as illustrated in Fig. 2. The 2D projection of the video is divided into $\mathcal{L}$ tiles arranged in a $L_H \times L_V$ regular grid. Therefore, we have $\mathcal{L} = L_H L_V$. Subsequent p frames in time constitute a group of pictures (GoP). Similar to the frame partitioning, each GoP is partitioned spatially to $\mathcal{L}$ GoP-tiles and the set of GoP-tiles is $L = \{1, 2, .., \mathcal{L}\}$. We encode every GOP into multiple layers of increasing immersion fidelity, for enhanced operational efficiency, as illustrated in Fig. 2. Thus, every GOP tile will feature one base layer and multiple enhancement layers that effectively trade-off delivered immersion fidelity for assigned transmission data rate. Next, we discuss our scalable 360 video transmission strategy and its effective integration into our multi-connectivity VR streaming arcade.

A. Scalable 360 Video Transmission

We introduce a scalable scheme which synergistically integrate with mmWave and Wi-Fi communication for efficient resource utilization. Using state-of-the-art scalable video coding [8], each GoP-tile $l \in L$ is encoded to a base layer and multiple enhancement layers as illustrated in Fig. 2. The base layer of all tiles $l \in L$ provides minimum low quality representation of the video. An enhancement layer alongwith the base layer of a tile $l \in L$ and lower enhancement layers provide high quality representation of the video content in the GoP-tile. Therefore, for each GoP-tile, the base layer and enhancement layer of each GoP-tile can be sent separately. A user receives the base layer of each GoP-tile through Wi-Fi while it receives the enhancement layer of a specific part of the GoP through mmWave AP as shown in Fig. 1. The edge server can process/decode the 360° video and also it provides facility to cache the layered GoP-tiles of the 360 video content.

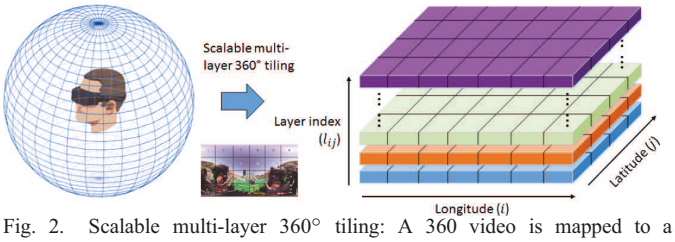

Fig. 2. Scalable multi-layer 360° tiling: A 360 video is mapped to a wide equirectangular panorama encoded into tiles featuring multiple layers of immersion fidelity.

Each user may look at different part of the video at any given time. Navigation traces are collected and based on the proposed strategy in [9], we obtain the expected viewport for the next GoP for each user $u \in U$. Let the expected viewport of user $u \in U$ during a GoP be expressed as a GoP-tile subset $L_u \subset L$. Each GoP-tile $l \in L_u$ is transmitted using an mmWave AP $a \in A$ to the user $u \in U$ in the following manner: Among the L_u set of GoP-tiles, the $L_{u,r} \subset L_u$ set of GoP-tiles are decoded at the edge cloud; the decoded bits of the $L_{u,r}$ set of GoP-tiles along with encoded enhancement layer of $L_u \setminus L_{u,r}$ set of GoP-tiles are sent to the AP a using a backhaul wired link. Then, this data is transmitted to the user u using the AP a by mmWave communication. The proposed novel integrated mmWave communication and network edge computation capability allows to decode part of the expected viewport tiles and send them as raw data with low latency. This results in reducing the delay of decoding the viewport content at the user and improving the system efficiency. A baseline representation of all the GoP-tiles $l \in L$ is broadcasted over Wi-Fi. The baseline and enhancement representations of each GoP-tile $l \in L_u$ which are received through Wi-Fi and mmWave communication, respectively, enable high VR immersion fidelity delivered to the user. Furthermore, sending the baseline representation of the GoP-tiles $l \in L$ over Wi-Fi compensate for the uncertainty of mmWave links and prospective expected viewport characterization error. Note that each user is associated with only one AP and each AP is associated with at most one user.

B. Wi-Fi Communication

The full set of GoP-tiles are broadcasted using Wi-Fi communication. Each GoP-tile $l \in L$ has immersion distortion D_l related to the data rate of the GoP-tile as $D_l = a_l R_l^{b_l}$ where a_l and b_l are constants [10]. Let the immersion distortion of each GoP-tile $l \in L$ which is sent over Wi-Fi communication is D_μ and the corresponding data rate of the GoP-tile is R_μ. The bandwidth allocated for Wi-Fi communication is B_w. Therefore the delay in transmitting all the GoP-tiles $l \in L$ using broadcast is $\tau_c = \mathcal{L}R_\mu / B_w \log(1+\gamma_{br})$ where γ_{br} is the broadcast SNR using Wi-Fi [1]. Once the encoded GoP-tiles are received at the VR headset of each user $u \in U$, it decodes all the GoP-tiles using the available computing equipment.

C. mmWave commmunication

Among the L_u set of GoP-tiles which are to be sent using mmWave AP a, the $L_{u,r} \subset L_u$ set of GoP-tiles are sent raw while $L_u \setminus L_{u,r}$ are sent encoded. Let the size of the enhancement layer of each GoP-tile $l \in L_{u,r}$ after decoding be b_r in number of bits and the data rate of each encoded GoP-tile $l \in L_u \setminus L_{u,r}$ is R_l. The user already receives the base layer of each GoP-tile $l \in L_u$ (of data rate R_μ) in Wi-Fi communication. The number of bits required to be sent for each compressed GoP-tile $l \in L_u \setminus L_{u,r}$ is $(R_l - R_\mu)$ using mmWave communication. Therefore, the delay in transmitting all the GoP-tiles $l \in L_u$ using mmWave communication is

$$\tau_{a,u} = \frac{|L_{u,r}| b_r + \sum_{l \in L_u \setminus L_{u,r}} (R_l - R_\mu)}{r_{a,u}} \tag{1}$$

where $r_{a,u}$ is the mmWave transmission rate through the link between AP a and user u and it depends upon transmit power, channel gain, noise power, and directional antenna gain [11, 12]. Interference is not present among the users due to the directional pencil beam communication between AP and user. Once the GoP-tiles are received at the VR headset of user $u \in U$, the tiles $l \in L_u \setminus L_{u,r}$ are decoded at the VR headset.

D. CPU Time analysis

Here, we analyze the time delay to decode the GoP-tiles at the edge cloud and the user. The time required to decode a GoP-tile depends upon the data rate of the GoP-tile. We assume that each GoP-tile $l \in L_{u,r}$ is decoded from the highest data rate (best quality) $R_{l,max}$ of the GoP-tile available at the edge. The GoP-tiles $l \in L_{u,r} \setminus L_u$ of data rate R_l and the GoP-tiles $l \in L$ of data rate R_μ are decoded at the user. To find the decoding time for these GoP-tiles, we first analyze the required CPU cycles to decode these GoP-tiles. Our experimental results demonstrate that the relationship between required CPU cycles for decoding a GoP-tile, denoted by β, and the data rate of the GoP-tile R can be established by a polynomial function as $\beta = cR^3 - dR^2 + eR + f$ where c, d, e and f are positive constants.

Let the processing capability of the VR headset of each user $u \in U$ is f_u and let $f_{u,1}$ and $f_{u,2}$ be the processing power allocated by the user to process the GoP-tiles that are received over Wi-Fi and mmWave, respectively, where $f_{u,1} + f_{u,2} \leq f_u$. Therefore, the CPU cycles required to decode the GoP-tiles $l \in L$ of quality R_μ at the user is $\mathcal{L}(cR_\mu^3 - dR_\mu^2 + eR_\mu + f)$ and the delay in decoding is $T_c = \mathcal{L}(cR_\mu^3 - dR_\mu^2 + eR_\mu + f)/f_{u,1}$. The total decoding delay of $L_{u,r}$ set of GoP-tiles at the edge cloud is $T_{u,1} = \sum_{l \in L_{u,r}} \beta_{l,k_r} / F_u$ where, $\beta_{l,k_r} = cR_{l,max}^3 - dR_{l,max}^2 + eR_{l,max} + f$ is the number of CPU cycles required to decode each of these GoP-tile and F_u is the computation resource allocated at the edge server to user u. The decoding delay of all the GoP-tiles $L_u \setminus L_{u,r}$ at the VR headset of user u is $T_{u,2} = \sum_{l \in L_u \setminus L_{u,r}} (cR_l^3 - dR_l^2 + eR_l + f)/f_{u,2}$. Therefore, the total delay to receive all the GoP-tiles $l \in L$ in Wi-Fi communication is $T_c + \tau_c$ and the total delay to receive all the GoP-tiles $l \in L_u$ using mmWave communication is $T_{u,1} + T_{u,2} + \tau_{a,u}$.

III. Problem Formulation

Let Π denotes the set of all possible AP to user assignments, for the AP set A and user set U, such that every member set $\pi \in \Pi$ features AP to user assignments comprising $|U|$ disjoint AP user pairs. For example, with $A = \{a_1, a_2\}$ and $U = \{u_1, u_2\}$, we have two different AP user assignments partitions $\{(a_1, u_1), (a_2, u_2)\}$, and $\{(a_1, u_2), (a_2, u_1)\}$ and $\Pi = \{\{(a_1, u_1), (a_2, u_2)\}, \{(a_1, u_2), (a_2, u_1)\}\}$.

Leveraging our recent advances in [9], we are able to characterize the likelihood of every GOP tile appearing in the user viewport over that GOP. We can then leverage this characterization to find expected user viewport as a set of GoP-tiles $l \in L_u$ and analytically formulate the expected immersion fidelity (or distortion) experienced by the user over the GOP. In particular, let w_l^u, $l \in L_u$ denote this navigation likelihood of GOP tile $l \in L_u$. Then, we can formulate the delivered expected immersion distortion of viewport of user u is $\sum_{l \in L_{u,r}} w_l^u a_l R_{l,max}^{b_l} + \sum_{l \in L_u \setminus L_{u,r}} w_l^u a_l R_l^{b_l}$. Our aim is to minimize the maximum expected immersion distortion of viewport among all the users $u \in U$ in the network. Therefore, the optimization problem of interest can be expressed as

$$\min_{\substack{\pi \in \Pi, \boldsymbol{F}, \boldsymbol{R} \\ f_{u,1}, f_{u,2}}} \max_{u \in U} \sum_{l \in L_{u,r}} w_l^u a_l R_{l,max}^{b_l} + \sum_{l \in L_u \setminus L_{u,r}} w_l^u a_l R_l^{b_l} \tag{2}$$

$$\text{s.t. } \frac{\mathcal{L}(cR_\mu^3 - dR_\mu^2 + eR_\mu + f)}{f_{u,1}} + \frac{\mathcal{L}R_\mu}{r'} \leq \tau \quad u \in U$$

$$\frac{\sum_{l \in L_{u,r}} \beta_{l,k_r}}{F_u} + \frac{\sum_{l \in L_u \setminus L_{u,r}} cR_l^3 - dR_l^2 + eR_l + f}{f_{u,2}}$$

$$+ \frac{|L_{u,r}| b_r + \sum_{l \in L_u \setminus L_{u,r}} (R_l - R_\mu)}{r_{a,u}} \leq \tau \quad u \in U$$

$$\sum_{u \in U} F_u \leq F, \quad f_{u,1} + f_{u,2} \leq f_u, \quad u \in U$$

where $r' = B_w \log(1+\gamma_{br})$, $\boldsymbol{F}$ is the vector of all values of F_u, $u \in U$, $\boldsymbol{R}$ is a set that contains all possible R_l, $\in L_u \setminus L_{u,r}$, $u \in U$, and τ is the maximum tolerable delay within which each GoP needs to be sent such that users do not experience any lag. The first constraint in (2) imposes that the total delay of receiving all baseline GoP-tiles $l \in L$ at

[1]The broadcast SNR is the minimal SNR among all the links. We assume channel state information is available at the transmitter to adapt its transmission rate accordingly.

the user, via Wi-Fi communication, be bounded by τ. The second constraint imposes that the total delay of receiving all enhancement GoP-tiles $l \in L$ at the user via mmWave communication be bounded by τ. The computation resource allocations at the edge cloud is restricted by the total available computation resource F as shown in the third constraint. The restriction on computation resource allocation for each user $u \in U$ is given by the fourth constraint. Minimizing the distortion is the same as maximizing the immersion fidelity due to the one-to-one mapping between them. Since (2) is a mixed-integer programming, it is hard to solve optimally in practice. First, we consider fixed cloud resource allocation, for which the problem in (2) decomposes into independent data rate and user computing power allocation for each user u to AP. The solution to each of these independent problems is obtained in Section V. Based on this solution, the user to AP assignment is obtained in Section IV. Finally, the joint computation resource allocation of the edge cloud and the users, and data rate selection of the part of the expected viewport is obtained in Section VI.

IV. AP to User Assignment

The optimal solution of AP to user assignment can be obtained by searching over all possible assignments Π. However, this requires searching over $(M+N)!/M!$ user assignments where $M = |U|$ is the number of users and $N = |A|$ is the number of APs. Alternatively, AP to user assignment problem reduces to the simpler user assignment problem which we proceed to solve optimally by means of a graph-theoretic matching algorithm.

We begin by reviewing some concepts of bipartite graph theory matching [13, 14]. A graph G comprising a vertex set $\mathcal{V}$ and an edge set $\mathcal{E}$ is bipartite if $\mathcal{V}$ can be partitioned into $\mathcal{V}^1$ and $\mathcal{V}^2$ (the bipartition), such that every edge in $\mathcal{E}$ connects a vertex in $\mathcal{V}^1$ to one in $\mathcal{V}^2$. A matching in G is a subset of $\mathcal{E}$ such that every vertex $v \in \mathcal{V}$ is incident to at most one edge of the matching. A maximum matching in G contains the largest possible number of edges. For the bipartite graph in Fig. 3(b), the two possible maximum matchings are $\{(v_a^1, v_u^1), (v_a^2, v_u^2)\}$ and $\{(v_a^1, v_u^2), (v_a^2, v_u^1)\}$.

To solve the AP to user assignment, first the network is represented as a weighted bipartite graph in which each AP $a \in \{1, .., N\}$ and each user $u \in \{1, .., M\}$ are represented by vertices $v_a^1 \in \mathcal{V}^1$ and $v_u^2 \in \mathcal{V}^2$, respectively, and the weight of the edges (v_a^1, v_u^2) is expressed as $\omega_{(v_a^1, v_u^2)} = D_{a,u}^*$ where $D_{a,u}^*$ is the immersion distortion when user u is assigned AP a. In section V, the procedure to obtain the value of each $D_{a,u}^*$ is described. Fig. 3 shows graph construction for a network with two users and two APs in which the immersion distortion between each user and APs are obtained from Section V as $D_{1,1}^* = 0.08$, $D_{1,2}^* = 0.05$, $D_{2,1}^* = 0.07$, and $D_{2,2}^* = 0.02$

The user selection problem in (2) can be expressed as a bottleneck matching (BM) problem of the graph defined by the maximum matching whose the largest edge weight is as small as possible, i.e.,

$$\min_{\phi \in \Phi} \max_{(v_a^1, v_u^2) \in \phi} \omega_{(v_a^1, v_u^2)} \tag{3}$$

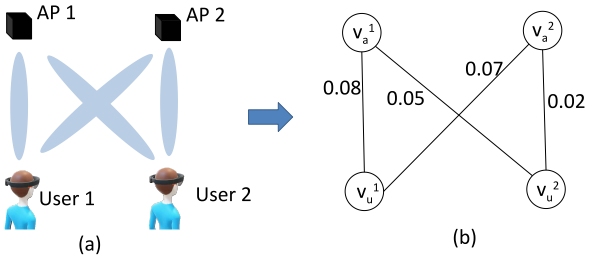

Fig. 3. Example of a bipartite graph for the network with 2 APs and 2 users.

where Φ contains all possible maximum matchings. For the graph in Fig. 3(b), the bottleneck matching is $\{(v_a^1, v_u^2), (v_a^2, v_u^1)\}$ and so the corresponding assignment is: AP 1 is assigned to user 2, and AP 2 is assigned to user 1. The user assignment problem can be solved optimally using the BM algorithm proposed in [14] with complexity $\mathcal{O}(N^{2.5})$.

V. Calculation of AP to User Weight

To find the AP to user weight, we begin by considering fixed cloud resource allocation. Here, we consider equal cloud resource allocation, i.e., $F_u = F/M$. Due to fixed cloud resource allocation, (2) can be separated into the following subproblems

$$\begin{aligned} \min_{\boldsymbol{R}, f_{u,1}, f_{u,2}} \quad & \sum_{l \in L_u \setminus L_{u,r}} w_l^u a_l R_l^{b_l} \\ \text{s.t.} \quad & \frac{\mathcal{L}(cR_\mu^3 - dR_\mu^2 + eR_\mu + f)}{f_{u,1}} + \frac{\mathcal{L}R_\mu}{r'} \le \tau \\ & \frac{\sum_{l \in L_{u,r}} \beta_{l,k_r}}{F_u} + \frac{\sum_{l \in L_u \setminus L_{u,r}} cR_l^3 - dR_l^2 + eR_l + f}{f_{u,2}} \\ & + \frac{|L_{u,r}| b_r + \sum_{l \in L_u \setminus L_{u,r}} (R_l - R_\mu)}{r_{a,u}} \le \tau \\ & f_{u,1} + f_{u,2} \le f_u, \end{aligned} \tag{4}$$

for each user u and AP a selection in an assignment π. Note that the first term in the objective function of (2) is constant for a fixed assignment π. Hence, we omit this term here. The above problem is nonconvex. It can be converted to a GP problem via the single condensation method [15]. According to this method, for a constraint which is a ratio of posynomials, the denominator posynomial (say $\mathrm{f}(\mathbf{x})$) can be approximated into a monomial using the following inequality:

$$\mathrm{f}(\mathbf{x}) = \sum_\ell \mathrm{f}_\ell(\mathbf{x}) \ge \hat{\mathrm{f}}(\mathbf{x}) = \prod_\ell \left[\frac{\mathrm{f}_\ell(\mathbf{x})}{\delta_\ell} \right]^{\delta_\ell}, \tag{5}$$

where $\delta_\ell > 0$ and $\sum_\ell \delta_\ell = 1$. Then, for $\delta_\ell = \mathrm{f}_\ell(\hat{\mathbf{x}})/\mathrm{f}(\hat{\mathbf{x}})$, $\hat{\mathrm{f}}(\hat{\mathbf{x}})$ is the best monomial approximation of $\mathrm{f}(\mathbf{x})$ near $\mathbf{x} = \hat{\mathbf{x}}$.

We formulate an iterative technique to optimally solve (4). At each iteration t, the first constraint in (4) is converted into a posynomial using (5) as

$$\begin{aligned} & \left(\frac{\tau r' f_{u,1}(t)}{\delta_1(t)} \right)^{-\delta_1(t)} \left(\frac{r' \mathcal{L} d R_\mu^2}{\delta_2(t)} \right)^{-\delta_2(t)} \\ & \cdot \left(r' \mathcal{L}(cR_\mu^3 + eR_\mu + f) + f_{u,1} \mathcal{L} R_\mu \right) \le 1 \end{aligned} \tag{6}$$

where $\delta_1(t)$, and $\delta_2(t)$ are obtained as

$$\delta_1(t) = \frac{\tau r' f_{u,1}(t-1)}{\tau r' f_{u,1}(t-1) + r' \mathcal{L} d R_\mu^2},$$
$$\delta_2(t) = \frac{r' \sum_{l \in L} d R_\mu^2}{\tau r' f_{u,1}(t-1) + r' \mathcal{L} d R_\mu^2}. \quad (7)$$

Similarly, at each iteration t, the second constraint in (4) is converted into a posynomial using (5) as

$$\left(\frac{\tau F_u r_{a,u} f_{u,2}(t)}{\delta_3(t)}\right)^{-\delta_3(t)} \left(\frac{F_u f_{u,2}(t) \sum_{l \in L_u \setminus L_{u,r}} R_\mu}{\delta_4(t)}\right)^{-\delta_4(t)}$$
$$\cdot \prod_{l \in L_u \setminus L_{u,r}} \left(\frac{F_u r_{a,u} d R_l(t)^2}{\delta_{5l}(t)}\right)^{-\delta_{5l}(t)} \Bigg(r_{a,u} f_{u,2}(t) \sum_{l \in L_{u,r}} \beta_{l,k_r}$$
$$+ F_u r_{a,u} \sum_{l \in L} (c R_l(t)^3 + e R_l(t) + f) + F_u f_{u,2}(t)$$
$$\left(|L_{u,r}| b_r + \sum_{l \in L_u \setminus L_{u,r}} R_l(t)\right)\Bigg) \leq 1 \quad (8)$$

where $\delta_3(t)$, $\delta_4(t)$ and $\delta_{5l}(t)$ are obtained from the solution at the $(t-1)$-th iteration as $\delta_3(t) = c_1/(c_1+c_2+c_3)$, $\delta_4(t) = c_2/(c_1+c_2+c_3)$, $\delta_{5l}(t) = c_3/(c_1+c_2+c_3)$ with $c_1 = \tau F_u r_{a,u} f_{u,2}(t-1)$, $c_2 = F_u f_{u,2} \sum_{l \in L_u \setminus L_{u,r}} R_\mu$, and $c_3 = F_u r_{a,u} d R_l(t-1)^2$. Let $D_{a,u}(t) = \sum_{l \in L_u \setminus L_{u,r}} w_l^u a_l R_l(t)^{b_l}$. Thus, the overall optimization to be solved at iteration t is

$$\min_{\boldsymbol{R}, f_{u,1}, f_{u,2}} \; D_{a,u}(t) \quad (9)$$
$$\text{s.t. } (6), (8), \;\; f_{u,1}(t) + f_{u,2}(t) \leq f_u,$$

The above optimization problem is GP problem and can be solved optimally. The iterative optimization is carried out until $|D_{a,u}(t) - D_{a,u}(t-1)| \leq \epsilon$ with $0 \leq \epsilon \ll 1$. An algorithmic implementation is included in Algorithm 1, which converges to the global solution [15]. The optimal value of the optimization problem (4) is obtained as $D_{a,u}(t)$ after termination of the Algorithm. Therefore, the optimal immersion f delity when user u assigned to AP a is $D^*_{a,u} = \sum_{l \in L_{u,r}} w_l^u a_l R_{l,max}^{b_l} + D_{a,u}(t)$.

Algorithm 1 GP based solution for (4).

1: Set $t = 1$, $f_{u,1}(t) = f_{u,2}(t) = f_u/2$, Initialize $R_l(t)$
2: **while** true **do** ▷ inf nite loop
3: $t = t + 1$
4: Calculate $\delta_1(t)$, $\delta_2(t)$
5: Find the optimum $f_{u,1}(t)$, $f_{u,2}(t)$ $D_{a,u}(t)$, $R_l(t)$ solving (9) using GGPLAB [16]
6: **if** $|D_{a,u}(t) - D_{a,u}(t-1)| \leq \epsilon$ **then**
7: Break
8: **end if**
9: **end while**
10: Assign $D^*_{a,u} = \sum_{l \in L_{u,r}} w_l^u a_l R_{l,max}^{b_l} + D_{a,u}(t)$.

VI. Cloud Resource Allocation

In section V, we consider f xed cloud resource allocation to solve the user to access point assignment at lower complexity. Now, for a given user to AP assignment, we can investigate a joint data rate, edge cloud and user computing power allocation. Therefore, for a given user to AP assignment, the optimization in (2) extends to the following problem

$$\min_{\substack{\boldsymbol{F}, \forall u, \boldsymbol{R} \\ f_{u,1}, f_{u,2}}} \; \max_{u \in U} \sum_{l \in L_u \setminus L_{u,r}} w_l^u a_l R_l^{b_l} \quad (10)$$
$$\text{s.t. } \frac{\mathcal{L}(c R_\mu^3 - d R_\mu^2 + e R_\mu + f)}{f_{u,1}} + \frac{\mathcal{L} R_\mu}{r'} \leq \tau \;\; u \in U$$
$$\frac{\sum_{l \in L_{u,r}} \beta_{l,k_r}}{F_u} + \frac{\sum_{l \in L_u \setminus L_{u,r}} c R_l^3 - d R_l^2 + e R_l + f}{f_{u,2}}$$
$$+ \frac{|L_{u,r}| b_r + \sum_{l \in L_u \setminus L_{u,r}} (R_l - R_\mu)}{r_{a,u}} \leq \tau \;\; u \in U$$
$$\sum_{u \in U} F_u \leq F, \;\; f_{u,1} + f_{u,2} \leq f_u, \;\; u \in U$$

The optimization problem can be converted as a GP by using single condensation method by following similar steps as given in Section V. Therefore optimal solution is obtained by GP based iterative algorithm similar to the Algorithm 1.

VII. Experimentation

Here, we present simulation results that evaluate the performance of the proposed strategy in terms of PSNR (Peak Signal to Noise Ratio), which is the most common quality metric in video processing. It can be expressed as $PSNR = 10 \log_{10}(255^2 / \sum_{l \in L} w_l^u D_l)$ where D_l is the distortion of tile $l \in L$. PSNR for the entire video is obtained by calculating PSNR for each GoP and averaging over them. For evaluation, we used the 'Runner' 360° video sequence captured at 8K resolution and 30 frames per second frame rate, from the STJU immersive video dataset [17]. Five users are uniformly distributed in a $5m \times 5m$ square room. The mmWave APs and the Wi-Fi router linked to an edge computing server are placed at the corner of the room. The simulation parameters are $R_{l,max} = 15$ Mbps, $R_\mu = 0.2$ Mbps, $M = 5$, $N = 10$, $F = 150$ GHz, $\tau = 1$ sec., $N_0 = -147$ dBM/Hz, $\epsilon = 10^{-5}$. The set $L_{u,r}$ consists of two randomly chosen tiles from the expected viewport tiles. As a reference method, to which we compare, we consider the 'Wi-Fi only' strategy where all the encoded GoP-tiles are sent over Wi-Fi at the maximum possible data rate enabled by the Wi-Fi link, achievable under the latency constrain of τ.

Fig. 4 demonstrates the PSNR performance of the proposed strategy when the mmWave data rate for all the links is equal and it varies from 400 Mbps to 900 Mbps, and $f_u = 3$ GHz. As the mmWave data rate increases, the PSNR of the proposed strategy increases. This is because with an increase in mmWave data rate, the GoP-tiles of the expected viewport encoded at much higher data rate can be sent of much higher data rate within the latency constraint and the immersion f delity increases. For mmWave data rate 400 Mbps and 900 Mbps, the proposed strategy achieves respectively 6 and 8 dB PSNR improvement over Wi-Fi only scheme which are signif cant gains and advances relative to the state-of-the-art.

In Fig. 5, we analyze the performance of the proposed strategy when the computing power of the user's mobile device

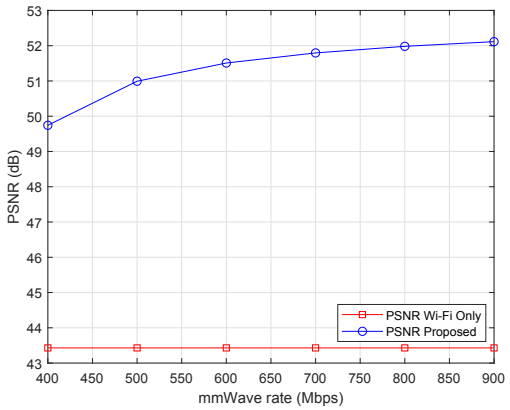

Fig. 4. PSNR versus mmWave transmission rate for 5 users and 10 APs.

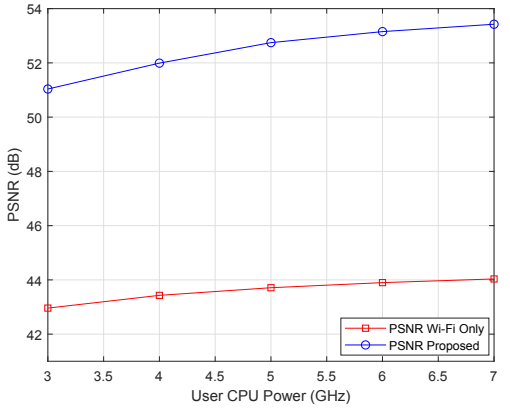

Fig. 5. PSNR versus user CPU power for 5 users and 10 APs.

increases from 3 GHz to 7 GHz. The mmWave data rate for the AP to user links are in the range of 600 Mbps to 900 Mbps. As the user's computing power increases, the decoding time at the user end decreases. Therefore, GOP-tiles encoded at much higher data rate can be sent within the latency constraint and the immersion f delity increases.

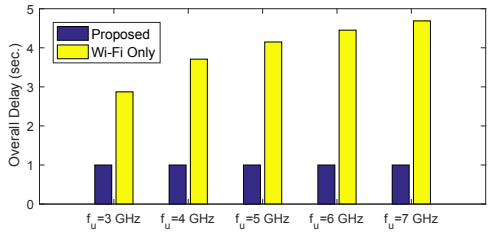

Fig. 6. End-to-end delay with variation of user CPU power.

In Fig. 6, we investigate the end-to-end delay for VR video delivery of the proposed strategy compared to the reference method when the Wi-Fi only scheme can hypothetically transmit at the same data rate and deliver the same immersion f delity as our proposed strategy. For this pupose, we consider that for Wi-Fi only scheme, all the GoP-tiles are sent encoded as the same data rate as that for our proposed strategy. As the user computing power increases from 3 GHz to 7 GHz, the immersion f delity increases for the proposed scheme while maintaining the end-to-end delay. However, a much higher end-to-end latency is induced by the Wi-Fi only scheme, due to the need to decode the received voluminous tiles encoded at high f delity in this case. Thus, the end-to-end delay considerably increases. The induced delay for the Wi-Fi only scheme becomes thereby 2.5 times to 4.5 times higher than proposed scheme, which would dramatically penalize the quality of experience of the user, as it would considerably lower the interactive nature of the VR application.

VIII. Conclusion

In this paper, we investigated a novel multiple user scalable 360° video mobile VR streaming arcade which enables high reliability and immersion f delity, and low latency, by the integration of scalable 360 content, mmWave communication and network edge computation capability. We have considered the joint optimization of the mmWave access AP to user association, data rate for the transmission of the part of the encoded 360 viewport content, and computation resource allocation to maximize the minimum VR immersion f delity across all users. The problem is mixed integer programming. Hence, we propose a low complexity solution for the problem. Simulation results show that the proposed system can achieve signif cant improvement of user immersion f delity/quality (6 dB to 8 dB) compared to a state-of-the-art approach that leverages Wi-Fi transmission only.

References

[1] Digi-Capital. Ubiquious 90 billion AR to dominate focused 15 billion VR by 2022. [Online]. Available: https://www.digi-capital.com/news/2018/01/ubiquitous-90-billion-ar-to-dominate-focused-15-billion-vr-by-2022/.
[2] Oculus. Rift. [Online]. Available: https://www3.oculus.com/en-us/dk2/
[3] HTC. Vive. [Online]. Available: https://www.htcvive.com/
[4] Samsung. Gear VR. [Online]. Available: http://www.samsung.com/us/explore/gear-vr/
[5] Google. Daydream. [Online]. Available: https://vr.google.com/
[6] C. Perfecto, M. Elbamby, J. D. Ser, and M. Bennis, "Taming the latency in multi-user VR 360°: A QoE-aware deep learning-aided multicast framework," *Arxiv*, 2018.
[7] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Edge computing meets millimeter-wave enabled VR: Paving the way to cutting the cord," in *Proc. IEEE WCNC*, April 2018, pp. 1–6.
[8] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high eff ciency video coding standard," *IEEE-Trans. CSVT*, vol. 26, no. 1, pp. 20–34, Jan. 2016.
[9] R. Aksu, J. Chakareski, and V. Swaminathan, "Viewport-driven rate-distortion optimized scalable live 360° video network multicast," in *IEEE ICMEW*, July 2018, pp. 1–6.
[10] J. Chakareski, R. Aksu, X. Corbillon, G. Simon, and V. Swaminathan, "Viewport-driven rate-distortion optimized 360° video streaming," in *Proc. IEEE ICC*, May 2018, pp. 1–7.
[11] H. Xu, V. Kukshya, and T. S. Rappaport, "Spatial and temporal characteristics of 60-GHz indoor channels," *IEEE JSAC*, vol. 20, no. 3, pp. 620–630, April 2002.
[12] J. Wildman, P. H. J. Nardelli, M. Latva-aho, and S. Weber, "On the joint impact of beamwidth and orientation error on throughput in directional wireless poisson networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 12, pp. 7072–7085, Dec 2014.
[13] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*. Philadelphia, PA, USA: SIAM, 2009.
[14] A. P. Punnen and K. Nair, "Improved complexity bound for the maximum cardinality bottleneck bipartite matching problem," *Discrete Applied Mathematics*, vol. 55, no. 1, pp. 91 – 93, 1994.
[15] G. Xu, "Global optimization of signomial geometric programming problems," *Eur. J. Oper. Res.*, vol. 233, no. 3, pp. 500–510, 2014.
[16] GGPLAB: A simple MATLAB toolbox for geometric programming. [Online]. Available: http://www.stanford.edu/boyd/ggplab/
[17] X. Liu, Y. Huang, L. Song, R. Xie, and X. Yang, "The SJTU UHD 360-degree immersive video sequence dataset," in *Proc. ICVRV*, Oct 2017, pp. 400–401.