mmWave Networking and Edge Computing for Scalable 360° Video Multi-User Virtual Reality

Sabyasachi Gupta, Jacob Chakareski, *Senior Member, IEEE*, and Petar Popovski, *Fellow, IEEE*

***Abstract*— We investigate a novel multi-user mobile Virtual Reality (VR) arcade system for streaming scalable 8K 360° video with low interactive latency, while providing high remote scene immersion fidelity and application reliability. This is achieved through the integration of embedded multi-layer 360° tiling, edge computing, and wireless multi-connectivity that comprises sub-6 GHz and mmWave (millimeter wave) links. The sub-6 GHz band is used for broadcast of the base layer of the entire 360° panorama to all users, while the directed mmWave links are used for high-rate transmission of VR-enhancement layers that are specific to the viewports of the individual users. The viewport-specific enhancements can comprise compressed and raw 360° tiles, decoded first at the edge server. We aim to maximize the smallest immersion fidelity for the delivered 360 content across all VR users, given rate, latency and computing constraints. We characterize analytically the rate-distortion trade-offs across the spatiotemporal 360° panorama and the computing power required to decompress 360° tiles. The proposed solution consists of geometric programming algorithms and an intermediate step of graph-theoretic VR user to mmWave access point assignment. The results reveal a significant improvement (8 – 10 dB) in delivered VR user immersion fidelity and spatial resolution (8K vs. 4K) compared to a state-of-the-art method based on sub-6 GHz transmission only. We also show that an increasing number of raw 360° tiles are sent, as the mmWave network link data rate or the edge server/user computing power increase. Finally, we demonstrate that in order to hypothetically deliver the same immersion fidelity, the reference method would incur a much higher (2.5-4.5x) system latency.**

***Index Terms*— Multi-user virtual reality, scalable 360° video, mmWave, edge computing, resource allocation.**

I. Introduction

VIRTUAL reality (VR) technologies are becoming increasingly popular in entertainment and gaming, education and training, healthcare, advertising, and social media. It is expected that VR technology will represent a 120 billion market by 2022 [1]. 360° video is an integral part of virtual reality systems and can enable remote scene immersion for a VR user experiencing it. Relative to traditional video streaming, VR based 360° video streaming has the following challenging requirements: ultra high data rate, ultra low response latency, and intensive computing. Thus, at present, only low-quality and low-resolution 360° videos can be streamed, and only over wired networks [2], [3]. The quality of experience is even worse over mobile devices, due to the much lower wireless bandwidth and computing capabilities of the user's mobile device to which the VR headset is attached. On the other hand, seamless untethered VR applications integrating high-fidelity real remote scene 360° content are expected to have the highest societal impact, advancing quality of life, the global economy, and energy conservation [4]. Enabling such applications is the objective we pursue within the present paper.

The latency in 360° video streaming systems comprises communication and computing delays that need to be constrained to 10-20 milliseconds end-to-end [5]. Relative to traditional 2D video, 360° video has a much higher resolution and temporal frame rate, and thus requires much more bandwidth to deliver the entire 3D 360° look-around remote scene it represents. High quality 360° video of frame rate of 100 frames per second and spatial resolution of 12K, as recommended by MPEG, can easily consume bandwidth of multiple Gigabits-per-second (Gbps) [6].

A. Proposed Framework and Contributions

We explore a novel streaming system for next-generation untethered VR that enables high immersion fidelity and low interactive latency, illustrated in Figure 1. It integrates synergistically, for the first time, high fidelity 8K scalable 360° video, edge computing, and wireless millimeter wave (mmWave) transmission. Firstly, using statistical characterization of user viewport navigation, we identify the subset of all tiles of the 360° content that have a high likelihood of being navigated by the user. Next, leveraging high computing capability at the network edge and using mmWave technology, our system enables transmitting a subset of viewport specific tiles as decompressed (therefore, reducing decoding delay at a user) and the rest of the viewport specific tiles as encoded at high data rates via mmWave. Moreover, to augment the system's reliability, and compensate for prospective mmWave link uncertainty and viewport prediction error, the base layer

Manuscript received 1 July 2021; revised 16 March 2022 and 14 November 2022; accepted 22 November 2022. Date of publication 15 December 2022; date of current version 21 December 2022. The work of Sabyasachi Gupta and Jacob Chakareski was supported by NSF under Award CCF-2031881, Award ECCS-2032387, Award CNS-2040088, Award CNS-2032033, and Award CNS-2106150; in part by NIH under Award R01EY030470; and in part by the Panasonic Chair of Sustainability at New Jersey Institute of Technology (NJIT). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Aline Roumy. *(Corresponding author: Sabyasachi Gupta.)*

Sabyasachi Gupta is with the Nokia Standards Department, Nokia Networks, Bengaluru 560045, India (e-mail: sabyasachi.gupta@nokia.com).

Jacob Chakareski is with the Ying Wu College of Computing, New Jersey Institute of Technology (NJIT), Newark, NJ 07102 USA (e-mail: jacob.chakareski@njit.edu).

Petar Popovski is with the Department of Electronic Systems, Aalborg University, 9220 Aalborg, Denmark (e-mail: petarp@es.aau.dk).

Digital Object Identifier 10.1109/TIP.2022.3228521

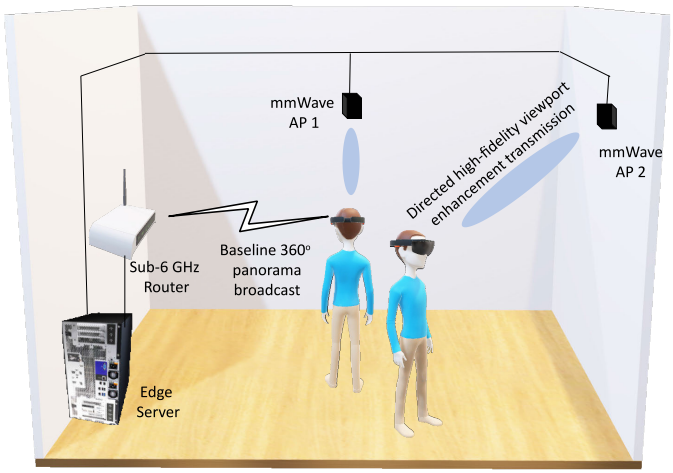

Fig. 1. A next generation mobile VR arcade with integrated edge computing and millimeter wave capability.

of the entire 360° panorama is broadcast over sub-6 GHz to all the users. These advances result in a significant improvement in VR user immersion fidelity.

The major technical contributions of this paper are the following:

- We propose a novel next generation mobile VR system with integrated edge computing and millimeter wave capability, 8K space-time scalable 360° multi-layer tiling, and multi-connectivity based one sub-6 GHz and mmWave links, and investigate the benefits of raw 360° tile mmWave transmission and the induced trade-offs between communication and computing latency. Viewport-adaptive scalable 360° tiling is integrated for efficient resource utilization.
- Joint optimization of raw 360° tile selection for mmWave transmission, allocation of remaining mmWave link data rate across the rest of the compressed 360° tiles, and allocation of edge server and user device computing resources. The objective is to maximize the delivered immersion fidelity, subject to end-to-end delay and transmission/computing constraints.
- We investigate the computing requirements of 360° tiles and formulate an accurate analytical polynomial model that captures the dependence between the number of CPU computing cycles required to decompress a tile and the transmission data rate assigned to it.
- To overcome the mixed-integer programming nature of the problem formulation, we formulate a lower-complexity solution that comprises multiple geometric programming algorithms and an intermediate step of graph-theoretic VR user to mmWave access point assignment. We rigorously characterize the computational complexity of our solution.
- Our experimental results demonstrate that the proposed 360° VR streaming system enables for the first time streaming high-fidelity 8K 360° videos to mobile VR clients. Significant performance gains in delivered immersion fidelity and interactive latency are demonstrated over a state-of-the-art reference method that relies on sub-6 GHz transmission.

B. Organization of the Paper

In Section II, we first discuss related work. Formulation of our system models is carried out in Section III. We present the problem formulation in Section IV and the respective optimization solution in Section V. We carry out comprehensive simulation experiments to assess the performance of our framework in Section VI. Finally, we conclude in Section VII.

II. Related Work

Tiling-based viewport-adaptive streaming [7], [8], [9], [10] is a popular scheme to efficiently utilize communication resources. In this approach, each frame of the 360° video is spatially divided into a number of rectangular regions called tiles, as illustrated in Figure 2. Each tile is encoded independently at a fixed data rate. At any time, a VR user experiencing the streaming content can only watch a limited spatial portion of the wide 360° panorama on its head-mounted display (HMD), known as the viewport. The streaming server, anticipating the user's viewing direction, sends then a smaller subset of the video tiles that overlap with the user's viewport over short periods of time (1-2s) into the future. Once the compressed tiles are received, they need to be decoded and rendered at the processing unit of the VR device. PC-based VR devices, e.g., Oculus Rift [2] and HTC Vive [3], are tethered to a powerful gaming computer and perform decoding locally on the computer/console. Hence, they lack mobility and present potential tripping hazzard. On the other hand, mobile VR solutions such as Samsung Gear VR [11] and Google Daydream [12] are attached to a mobile device, with limited computing capabilities and wireless bandwidth, which adversely impact the quality of experience.

Resource allocation for scalable 360° video transmission in sub-6 GHz networks has been investigated recently [13] and [14]. Sun et al. [13] proposed a two layer system for

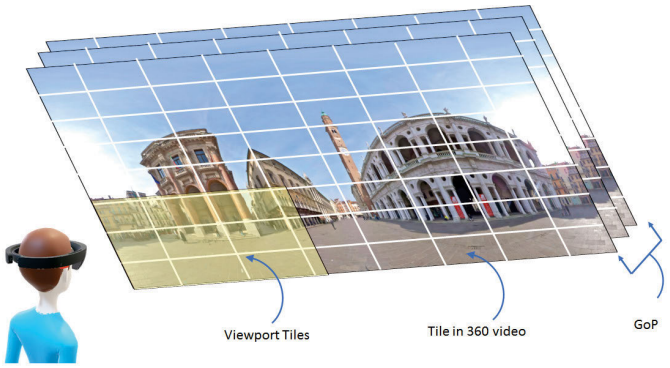

Fig. 2. Spatiotemporal tiling and Group of Pictures (GOP) of a 360° video.

360 video streaming, which can dynamically adapt the bitrates of the base layer and the enhancement layer according to the network throughput variations and viewport prediction errors. In [14], the authors proposed a cooperative streaming scheme of scalable viewport specific 360° content in which base station multicasts viewport specific 360° content to all users and users locally share part of the received viewport specific 360° content using mobile ad hoc network. Aerial 360° video streaming for remote scene immersion in next generation UAV-IoT applications has been explored in [4]. Cooperative mobile-edge multi-user 360° video delivery in 5G small-cell cellular networks has been investigated in [15].

In [16], a terahertz network with multiple reconfigurable intelligent surfaces (RISs) and users, and a single transmitter is considered for VR applications, and the problem of RIS to user association is investigated with the aim of optimizing the rate and reliability of the network. Chen et al. [17] propose a federated learning based user to base station (BS) association strategy with the aim of minimizing "breaks in presence" in VR, where the content is transmitted from the BSs to the users using mmWave communication. However, bitrate allocation for the transmitted VR content is not considered in this study. Moreover, efficient signal processing techniques such as scalable video coding and 360° video tiling are not considered. Therefore, the above strategy may be sub-optimal for maximizing the VR users' experienced immersion fidelity.

The role of edge computing and caching for wireless VR has been discussed recently [18], [19], [20], [21], and [22]. In particular, the study in [18] discusses the possibility of improving the overall system delay by the use of edge computing for VR gaming. Similarly, for 360° video delivery and VR gaming, the work in [19] examines the data rate and latency requirement for enabling wireless VR with the help of edge computing. Similarly, [23] has shown that high data rate mmWave link can support uncompressed video transmission. Sukhmani et al. [20] shown that the distributed cache can improve the service performance of VR applications. The users can cache and share their local content resources between neighbors which can greatly shorten the service latency. [21] proposes an efficient VR framework in which selection of device for rendering and caching of the viewport content for each user is made based on available computation and cache availability at the users and the edge server. In [22], the authors investigate the communication resource allocation and base station selection to optimize VR video delivery delay and tracking accuracy in an orthogonal frequency division multiple access (OFDMA) network. Uplink and downlink resources are used to transmit cellular user's tracking information and VR scenes, respectively. Edge computing helps to generate VR scenes when erroneous tracking information is received due to poor uplink wireless channel.

The integration of mmWave and edge computing for mobile VR has been considered in [19], [24], and [25], however, with limited contributions. [19] studies a user clustering strategy to maximize the user field-of-view frame request admission. [25] studies proactive computing and caching of synthesized interactive VR video frames, to minimize the traffic volume of VR gaming. [24] introduces a parallel rendering and streaming mechanism to reduce the addon streaming latency, by pipelining the rendering, encoding, transmission and decoding procedures. The strategies considered in [19] and [25] are heuristic and do not necessarily enable higher VR immersion fidelity for the user. Similarly, only low-quality low-resolution (4K) 360° content has been considered. Both of this shortcomings considerably penalize the delivered quality of experience. On the other hand, we considerably advance the state-of-the-art by a rigorous end-to-end system analysis and optimization that aim to maximize the delivered immersion fidelity subject to various system constraints. Moreover, we synergistically integrate raw 360° tile transmission, 8K scalable 360° multi-layer tiling, and dual sub-6 GHz and mmWave transmission, and provide rigorous analysis of the enabled benefits and induced performance trade-offs. Our experimental results demonstrate the considerable performance advances enabled by our framework.

The present work is based on our preliminary studies in [26] and [27]. In [26], we investigated the joint optimization of mmWave access point to user assignment, the allocation of mmWave data rate for a given set of viewport-specific enhancement tiles, and allocation of computing resources at the edge server and user devices for the proposed streaming system. In this work and [27], along with the above mentioned design variables, we jointly decide the choice of viewport-specific enhancement tiles to be transmitted decompressed to provide a complete analysis of our proposed system. Compared to our previous work in [27], the major improvements in this present study are as follows: (*i*) Complete mathematical analysis of the proposed optimization problem is provided, (*ii*) Complexity of the proposed solution is investigated, and (*iii*) New analysis of the dependency CPU computing cycles versus data rate is obtained.

III. VR Arcade System Models

A. General Aspects

In our VR arcade system, illustrated in Figure 1, there are M users receiving the 360° video content through wireless VR headsets. There are N mmWave APs and a sub-6 GHz router interlinked with a collocated edge server, where the compressed content resides. A user is assigned to a mmWave access point (AP) and viewport-specific enhancement layers of

TABLE I
MAJOR NOTATION USED IN THE PAPER

Parameters	Definition
U, A	Set of users and set of mmWave APs
L,	Set of GOP-tiles and size of $\mathcal{L}$
L_u,	GOP-tile set in user u's expected viewport
$\mathcal{L}_u$,	No. of GOP-tiles in user u's expected viewport
$L_{u,r}$	Set of GOP-tiles in L_u to be sent as raw data
$\mathcal{L}_{u,r}$,	No. of GOP-tiles in L_u to be sent as raw data
$L_{u,c}$	Set of GOP-tiles in L_u to be sent as encoded
$\mathcal{L}_{u,c}$	No. of GOP-tiles in L_u to be sent as encoded
R_μ	Data rate of sub-6 GHz communication
R_l	Data rate of each encoded GOP-tile $l \in L_{u,c}$
b_r	Size of the enhancement layer of GOP-tile $l \in L_{u,r}$ after decoding
f_u	Processing power of user u
$f_{u,1}$	User u's CPU power applied to decode the GOP-tiles that are received over sub-6 GHz
$f_{u,2}$	User u's CPU power applied to decode the GOP-tiles $l \in L_{u,c}$.
β_{l,k_r}	No. of CPU cycles required to decode GOP-tile $l \in L_{u,r}$
F_u	Computation resource allocated at the edge server to user u.
Π	Set of all possible AP to user assignments
p_l^u	Navigation likelihood of GOP tile $l \in L_u$
r'	Downlink broadcast transmission rate of sub-6 GHz
$r_{a,u}$	mmWave transmission rate through the AP a to user u link
τ	Maximum tolerable end-to-end delay

the scalable 360° content are sent to this user using mmWave transmission, comprising both raw and compressed tiles. Furthermore, the base layer of the entire 360° panorama is broadcast over sub-6 GHz to all the users. Table I summarizes the main notation used throughout the development of our system models, problem formulation, and optimization techniques, over the following three sections of the paper.

B. Sub-6 GHz and mmWave Communication Model

Here, we describe the two transmission media used in our system. The bandwidth allocated for sub-6 GHz communication is B_w. Therefore the sub-6 GHz rate is $r' = B_w \log(1 + \gamma_{br})$ where γ_{br} is the broadcast SNR using sub-6 GHz.

The mmWave channel is based on measurement results of line of sight (LoS) or non-line of sight (NLoS) paths for the 60-GHz indoor channels, and includes both pathloss attenuation l_{au} and small scale Nakagami fading with coefficient g_{au} [28]. The channel gain h_{au} from AP a to user u is thus given by $|h_{au}|^2 = l_{au}|g_{au}|^2$. The corresponding Nakagami shape factor m_{au} will take value m_L for LoS and m_N for NLoS paths; it is further assumed that g_{au} is i.i.d. and not temporally correlated. The pathloss value is $l_{au} = C_L d^{-\alpha_L}$, if there is a LoS path between AP a and user u. Otherwise, $l_{au} = C_N d^{-\alpha_N}$. Here, C_L and C_N are constants that depend on LoS or NLoS channel characteristics, respectively.

The radiation pattern of actual directional antennas is approximated with a 2D sectored antenna model [29]. Let g_{au}^{Tx} and g_{au}^{Rx} denote the transmission and reception antenna gains from AP a to the user u. They are defined as $g_{au}^{\bullet} = \frac{2\pi-(2\pi-\phi_a)g_{sl}}{\phi_a}$, if $v_{au}^{\bullet} \leq \frac{\phi_a}{2}$, and $g_{au}^{\bullet} = g_{sl}$, otherwise. Here, $\bullet \in \{Tx, Rx\}$, $v_{au}^{\bullet}$ stands for the angular deviation from the boresight directions, and g_{sl} is the constant sidelobe gain with $g_{sl} \ll 1$. Thus, the AP a to user u SNR is $\gamma_{au} = \frac{p_a|h_{au}|^2 g_{au}^{Tx} g_{au}^{Rx}}{N_0 B}$, where p_a is the transmit power by the AP a, B is the bandwidth for mmWave communication, and N_0 is the noise power spectral density. Therefore, the AP a to user u rate is $r_{a,u} = B \log(1+\gamma_{au})$. We use the approximation that mmWave interference among the user links is absent due to the directed pencil-shaped mmWave beams between users and APs. Also, the transmission rate $r_{a,u}$, for each link between AP $a \in A$ and user $u \in U$ is available at the centralized controller, by continuous monitoring of the beam direction [30].

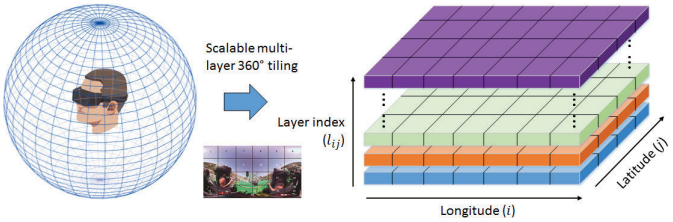

Fig. 3. Scalable multi-layer 360° tiling. Equirectangular projection is carried out first.

C. Scalable 360° Video Encoding and Decoding Model

We introduce a scalable 360° video representation method that synergistically integrates with mmWave and sub-6 GHz communication for efficient resource utilization. First, using an equirectangular projection, the raw spherical video frames are mapped to respective 2D panoramas, as illustrated in Figure 3 below, to enable the application of state-of-the-art video coding [31], [32]. Then, each panoramic 360° video frame is partitioned into $L_H \times L_V$ tiles. Finally, the tiled frames are grouped temporally into blocks of subsequent frames, each denoted as Group of Pictures (GOP) and compressed separately. Here, we denote the sequence of tiles at the same spatial location (h, v) in a GOP as a GOP-tile and construct K scalable layers of increasing immersion fidelity for each GOP-tile, as illustrated in Figure 3, by applying the scalable extension of the latest video compression standard denoted as SHVC [33] independently to each GOP-tile [33]. The first layer of a compressed GOP-tile is known as the base layer, and the remaining $K-1$ layers are denoted as enhancement layers. The reconstruction fidelity of a GOP-tile improves incrementally as more layers are being decoded progressively starting from the base layer. Let L denote the set of all GOP-tiles. Each GOP-tile $l \in L$ exhibits an immersion reconstruction distortion D_l related to the encoding data rate R_l of the GOP-tile as $D_l = a_l R_l^{b_l}$ where a_l and b_l are constants [34].

For each user, a subset of GOP-tiles are decoded at the edge server and rest of the GOP-tiles are decoded at the user. The time required to decode a GOP-tile depends upon the data rate of the GOP-tile, which in turn depends on the number of scalable layers from which the tile is decoded/reconstructed. To find the time delay induced by decoding GOP-tiles at the edge server or a user, we analyze the number of CPU computing cycles β required to decode a GOP-tile as a function of its data rate R. Our empirical results shown in

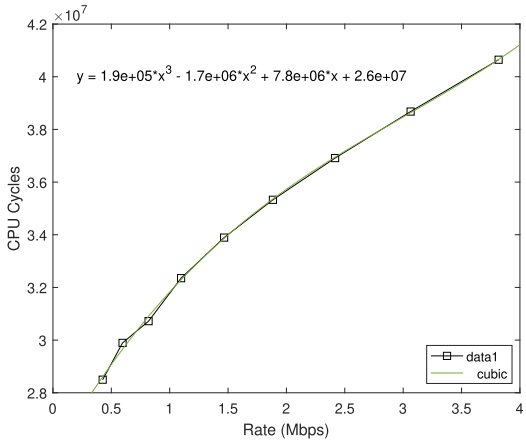

Fig. 4. # CPU computing cycles versus data rate dependency using polynomial modeling.

Figure 4 demonstrate a polynomial relationship that we can capture as $\beta = cR^3 - dR^2 + eR + g$, where c, d, e, and g are positive constants. We indicate the values of these constants in Figure 4. The computation delay of a GOP-tile can be expressed as the ratio of required CPU cycles to decode and CPU frequency of the equipment at which the tile is decoded.

D. Proposed Video Transmission Framework

The overall delivery process of the proposed 360° streaming and associated delay is shown in Fig. 5. Multiple enhancement layers of viewport-specific GOP-tiles are sent to the individual users over mmWave links. Specifically, we leverage our ongoing work on statistical characterization of user viewport navigation [34], [35], to identify as the subset of 360° GOP-tiles that overlap with the VR viewport of user u over that GOP. Essentially, it comprises the GOP-tiles that exhibit a non-zero likelihood of being navigated by the user during that GOP of the 360° content. Let L_u denote the set of GoP-tiles in expected viewport of user u. Enhancement layers of tiles $l \in L_u$ will be sent to user u using mmWave transmission from one select AP, to augment the enabled immersion fidelity. In particular, from the L_u set of GOP-tiles to be sent via mmWave AP a, the $L_{u,r} \subseteq L_u$ subset of GOP-tiles are sent raw, while the remaining $L_{u,c} = L_u \setminus L_{u,r}$ tiles are sent compressed. When carried out at the edge server, GOP-tile $l \in L_{u,r}$ is decoded from the highest available data rate (best quality) $R_{l,max}$ of the GOP-tile. This corresponds to reconstructing the tile from all K scalable layers into which it has been encoded using our approach. Let data rate of each encoded GOP-tile $l \in L_{u,c}$ is R_l. To augment the system's reliability, and compensate for prospective mmWave link uncertainty and viewport prediction error, the base layer of every GOP-tile $l \in L$ is broadcast over sub-6 GHz to all users. Let the immersion distortion of each GOP-tile $l \in L$ which is sent over sub-6 GHz communication be D_μ and the corresponding data rate of the GOP-tile be R_μ. Since we construct/encode the 360° content in scalable manner, the number of bits required to be sent for the enhancement layers of each compressed GOP-tile $l \in L_{u,c}$ is $(R_l - R_\mu)$. Therefore, via mmWave channel, the GOP-tiles $l \in L_{u,c}$ is transmitted to the user in time $\tau^1_{a,u}$, which is expressed as

$$\tau^1_{a,u} = \frac{\sum_{l \in L_{u,c}} (R_l - R_\mu)}{r_{a,u}} \tag{1}$$

Simultaneously, $L_{u,r}$ GOP-tiles is decoded in time $T_{u,2}$, which is expressed as

$$T_{u,1} = \frac{\sum_{l \in L_{u,r}} \beta_{l,k_r}}{F_u}, \tag{2}$$

where $\beta_{l,k_r} = cR^3_{l,max} - dR^2_{l,max} + eR_{l,max} + g$ is the number of CPU cycles required to decode tile $l \in L_{u,r}$, F_u is the edge server's computing resource allocated to user u. Let the size of each GOP-tile $l \in L_{u,r}$ after decoding be b_r. After completion of $L_{u,c}$ GOP-tiles' delivery and $L_{u,r}$ GOP-tiles' decoding in time $\max(T_{u,1}, \tau^1_{a,u})$, the raw GOP-tileset is sent via mmWave. The delay for transmitting this information is given by

$$\tau^2_{a,u} = \frac{\mathcal{L}_{u,r} b_r}{r_{a,u}} \tag{3}$$

where $\mathcal{L}_{u,r} = |L_{u,r}|$. Therefore, overall delay associated with decoding and transmission of $L_{u,r}$ GoP tiles is $\max(T_{u,1}, \tau^1_{a,u}) + \tau^2_{a,u}$.

Let $\tau_{c,1}$ denote the delay in transmitting the GOP-tiles $l \in L_{u,c}$ over sub-6 GHz to all users, *i.e.*, $\tau_{c,1} = \frac{\mathcal{L}_{u,c} R_\mu}{r'}$, where $\mathcal{L}_{u,c} = |L_{u,c}|$. The $L_{u,c}$ GoP-tiles can be decoded at the user after the enhancement layer information as well as the base layer information of all $L_{u,c}$ GoP-tiles are received at the user, *i.e.*, after the delay of $\max(\tau^1_{a,u}, \tau_{c,1})$. Let the processing capability of the VR headset of user u be f_u, and let $f_{u,1}$ and $f_{u,2}$ be the processing power allocated by the user to decode the GOP-tiles received over sub-6 GHz and mmWave, respectively, where $f_{u,1} + f_{u,2} \leq f_u$. Thus, the number of CPU computing cycles required to decode the GOP-tiles $l \in L_{u,c}$ of data rate R_l at the user is $(cR^3_l - dR^2_l + eR_l + g)$. Hence, the induced decoding delay can be formulated as:

$$T_{u,2} = \frac{\sum_{l \in L_{u,c}} cR^3_l - dR^2_l + eR_l + g}{f_{u,2}}. \tag{4}$$

Therefore, the overall delay associated with transmission and decoding of $L_{u,c}$ GoP tiles is $\max(\tau^1_{a,u}, \tau_{c,1}) + T_{u,2}$.

Let the delay delay in transmitting all the GOP-tiles $l \in L$ over sub-6 GHz to all users be τ_c which can be expressed as

$$\tau_c = \frac{\mathcal{L} R_\mu}{r'} \tag{5}$$

where $\mathcal{L} = |L|$. Therefore, we have $\tau_c > \tau_{c,1}$. After delivery of the GOP-tiles $l \in L$ in time τ_c over sub-6 GHz communication, each user u can compute $L \setminus L_{u,c}$ GoP-tiles with processing power $f_{u,1}$. The number of CPU computing cycles required to decode the GOP-tiles $l \in L \setminus L_{u,c}$ of data rate R_μ at the user is $(\mathcal{L} - \mathcal{L}_{u,c})(cR^3_\mu - dR^2_\mu + eR_\mu + g)$. Hence, the induced decoding delay can be formulated as:

$$T_{u,3} = \frac{(\mathcal{L} - \mathcal{L}_{u,c})(cR^3_\mu - dR^2_\mu + eR_\mu + g)}{f_{u,1}}. \tag{6}$$

Therefore, the overall delay associated with transmission and decoding of $L \setminus L_{u,c}$ GoP-tiles is $\tau_c + T_{u,3}$.

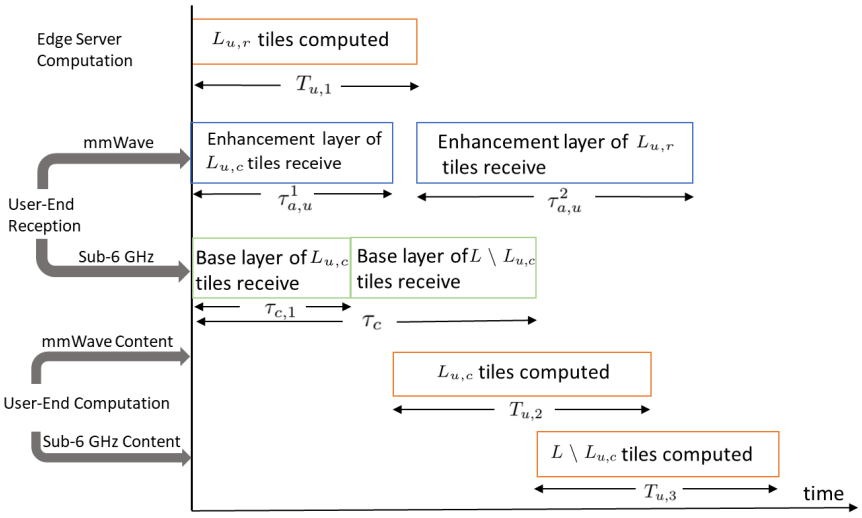

Fig. 5. Operations at different devices for delivery of 360° video to user u.

Therefore, overall delay associated at the user u to receive and decode base layer content of all GoP-tiles L and enhanced layer content of GoP-tiles L_u is $\max(\max(T_{u,1}, \tau^1_{a,u}) + \tau^2_{a,u}, \max(\tau^1_{a,u}, \tau_{c,1}) + T_{u,2}, \tau_c + T_{u,3})$

IV. 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 $\Pi = \{\{(a_1, u_1), (a_2, u_2)\}, \{(a_1, u_2), (a_2, u_1)\}\}$. Furthermore, let $\boldsymbol{L}_u$ be the power set of the set L_u which is the set of all subsets of L_u, including the empty set and L_u itself.

Leveraging our recent advances in [34] and [36], we can characterize the likelihood of every GOP-tile appearing in the user viewport over that GOP and the expected immersion distortion experienced by the user, given a data rate allocation across the GOP-tiles. In particular, let p^u_l, $l \in L_u$, denote this navigation likelihood of GOP tile $l \in L_u$. Then, we can formulate the expected immersion distortion experienced by user u in our system as $\sum_{l \in L_{u,r}} p^u_l a_l R^{b_l}_{l,max} + \sum_{l \in L_{u,c}} p^u_l a_l R^{b_l}_l$. Our aim is to minimize the maximum expected immersion distortion over all users, for given system and application constraints. Thus, the optimization problem of interest can be expressed as:

$$\min_{\substack{L_{u,r} \in \boldsymbol{L}_u, \boldsymbol{R}_{L_{u,r}}, \\ f_{u,1}, f_{u,2}, \pi \in \Pi, \boldsymbol{F}}} \max_{u \in U} \sum_{l \in L_{u,r}} p^u_l a_l R^{b_l}_{l,max} + \sum_{l \in L_{u,c}} p^u_l a_l R^{b_l}_l,$$

$$\begin{aligned} \text{s.t.} \quad & \max(T_{u,1}, \tau^1_{a,u}) + \tau^2_{a,u} \leq \tau \quad u \in U, \\ \text{s.t.} \quad & \max(\tau^1_{a,u}, \tau_{c,1}) + T_{u,2} \leq \tau \quad u \in U, \\ \text{s.t.} \quad & \tau_c + T_{u,3} \leq \tau \quad u \in U, \\ & \sum_{u \in U} F_u \leq F, \quad f_{u,1} + f_{u,2} \leq f_u, \quad u \in U, \end{aligned} \tag{7}$$

where $\boldsymbol{R}_{L_{u,r}}$ is a set that contains all R_l, $l \in L_{u,c}$, for a given choice of $L_{u,c} \in \boldsymbol{L}_u$, $\boldsymbol{F}$ is the vector of all values of F_u, $u \in U$, 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 (8) imposes that the total delay of decoding and receiving GOP-tiles $l \in L_{u,r}$ at the user, be bounded by τ, the maximum tolerable end-to-end delay. The second constraint imposes that the total delay of receiving enhancement layer GOP-tiles $l \in L_{u,c}$ at the user and decoding these GOP-tiles be bounded by τ. Similarly, the third constraint imposes that the total delay of receiving and decoding base layer GOP-tiles $l \in L$ at the user be bounded by τ. The computation resource allocation at the edge server is restricted by the total available computation resource F, as shown in the fourth constraint. Similarly, the restriction on computation resource allocation for each user $u \in U$ is given by the fifth constraint. We aim to select the optimal $L_{u,r}$ from $\boldsymbol{L}_u$, find the set of optimal data rate $\boldsymbol{R}_{L_{u,r}}$ for the selected tile set $L_{u,r}$, computing resource allocation, and AP to user assignment to minimize immersion distortion across all the users. Minimizing the immersion distortion is the same as maximizing the immersion fidelity, due to the one-to-one mapping between them [34]. The proposed statistical characterization of user navigation captures as navigation likelihoods the expected overlap of a tile with the user viewport over a time duration and the factor that equatorial tiles are more likely navigated than polar tiles. Therefore, our expected viewport quality formulation can correspond to a tile-level Weighted Spherical PSNR (WS-PSNR) [37].

Using (1), (2), (3), (4), (5), and (6), we can rewrite problem (7) as follows

$$\min_{\substack{L_{u,r} \in \boldsymbol{L}_u, \boldsymbol{R}_{L_{u,r}}, \\ f_{u,1}, f_{u,2}, \pi \in \Pi, \boldsymbol{F}}} \max_{u \in U} \sum_{l \in L_{u,r}} p^u_l a_l R^{b_l}_{l,max} + \sum_{l \in L_{u,c}} p^u_l a_l R^{b_l}_l,$$

$$\text{s.t.} \quad \frac{\sum_{l \in L_{u,r}} \beta_{l,k_r}}{F_u} \leq V_{u,1}, \quad \frac{\sum_{l \in L_{u,c}} (R_l - R_\mu)}{r_{a,u}} \leq V_{u,1} \quad u \in U,$$

$$\begin{aligned}
&\text{s.t.}\ \frac{\sum_{l\in L_{u,c}}(R_l-R_\mu)}{r_{a,u}} \le V_{u,2},\quad \frac{\mathcal{L}_{u,c}R_\mu}{r'}\\
&\le V_{u,2}\ \ u\in U,\\
&\text{s.t.}\ V_{u,1}+\frac{\mathcal{L}_{u,r}b_r}{r_{a,u}} \le \tau\ \ u\in U,\\
&\text{s.t.}\ V_{u,2}+\frac{\sum_{l\in L_{u,c}} cR_l^3-dR_l^2+eR_l+g}{f_{u,2}}\\
&\le \tau\ \ u\in U,\\
&\text{s.t.}\ \frac{\mathcal{L}R_\mu}{r'}+\frac{\mathcal{L}'_u(cR_\mu^3-dR_\mu^2+eR_\mu+g)}{f_{u,1}}\\
&\le \tau\ \ u\in U,\\
&\sum_{u\in U}F_u\le F,\quad f_{u,1}+f_{u,2}\le f_u,\ \ u\in U, \qquad (8)
\end{aligned}$$

where $V_{u,1}$, $V_{u,2}$, $\forall u \in U$, are slack variables, and $\mathcal{L}'_u = (\mathcal{L}-\mathcal{L}_{u,c})$, which has a non-negative value since $\mathcal{L}\ge\mathcal{L}_{u,c}$.

V. Optimization Solution

A. Solution Discussion and Outline

The above optimization problem is mixed-integer programming, which is hard to solve optimally in practice, due to its complexity. Thus, we investigate a lower-complexity solution framework that comprises three steps applied sequentially. First, we fix the edge server's computing resource allocation, e.g., a uniform allocation across all mobile VR users, to be able to tackle the integer variables in the optimization problem. For a given edge server computing resource allocation, the problem in (8) decomposes into a joint mmWave data rate allocation, user computing resource allocation, and raw tile selection, for each user u to AP a pairing in a given assignment. We compute solutions to each of these independent problems, for any prospective user to AP assignment π, in Section V-B. These solutions will produce edge weights for any user u to AP a prospective pairing, which we will then leverage in Section V-C to compute the optimal user to AP assignment π^* using a graph-theoretic solution. Finally, given this assignment, and considering again $\boldsymbol{F}$ as variable, we resolve (8), to jointly identify/update the optimal computing resource allocation at the edge server and the users, and the mmWave data rate allocation. This step is carried out in Section V-D and completes our optimization strategy. Figure 6 illustrates our optimization framework and its major steps.

B. Computing the Optimal Edge Weights for AP to User Pairings

We set the edge server's computing resource allocation to be constant and uniform across all users, i.e., $F_u = F/M$. Due to the fixed allocation, (8) decouples into the following subproblems:

$$\begin{aligned}
&\min_{\substack{L_{u,r}\in\boldsymbol{L}_u,\boldsymbol{R}_{L_{u,r}},\\ f_{u,1},f_{u,2}}} \sum_{l\in L_{u,r}} p_l^u a_l R_{l,max}^{b_l}+\sum_{l\in L_u\setminus L_{u,r}} p_l^u a_l R_l^{b_l},\\
&\text{s.t.}\frac{\sum_{l\in L_{u,r}}\beta_{l,k_r}}{F_u}\le V_{u,1},\ \frac{\sum_{l\in L_{u,c}}(R_l-R_\mu)}{r_{a,u}}\le V_{u,1},\\
&\text{s.t.}\ \frac{\sum_{l\in L_{u,c}}(R_l-R_\mu)}{r_{a,u}}\le V_{u,2},\ \ \frac{\mathcal{L}_{u,c}R_\mu}{r'}\le V_{u,2},\\
&\text{s.t.}\ V_{u,1}+\frac{\mathcal{L}_{u,r}b_r}{r_{a,u}}\le\tau,\\
&\text{s.t.}\ V_{u,2}+\frac{\sum_{l\in L_{u,c}} cR_l^3-dR_l^2+eR_l+g}{f_{u,2}}\le\tau,\\
&\text{s.t.}\ \frac{\mathcal{L}R_\mu}{r'}+\frac{\mathcal{L}'_u(cR_\mu^3-dR_\mu^2+eR_\mu+g)}{f_{u,1}}\le\tau,\\
&f_{u,1}+f_{u,2}\le f_u, \qquad (9)
\end{aligned}$$

for each user u and AP a pairing in an assignment π. To solve (9), we first consider a fixed set of raw enhancement GOP-tiles $L_{u,r}$, and formulate an optimization strategy to solve the allocation of mmWave data rate and user computing resource. We then show how to integrate the selection of $L_{u,r}$ into our optimization strategy, by reformulating (9) accordingly.

1) Fixed Set of Raw GOP-Tiles: We first solve (9), for a given $L_{u,r}\in\boldsymbol{L}_u$. We can show that this problem is not convex. It can be reformulated into geometric programming (GP) via the single condensation method [38]. 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:

$$\mathrm{f}(\mathbf{x})=\sum_\ell \mathrm{f}_\ell(\mathbf{x})\ge\hat{\mathrm{f}}(\mathbf{x})=\prod_\ell\left[\frac{\mathrm{f}_\ell(\mathbf{x})}{\delta_\ell}\right]^{\delta_\ell}, \qquad (10)$$

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 (9) in this case. In particular, at each iteration t, the second constraint in (9) is converted into a posynomial using (10) as:

$$\sum_{l\in L_{u,c}} R_l(t)\left(\frac{\mathcal{L}_{u,c}R_\mu}{\delta_1(t)}\right)^{-\delta_1(t)}\left(\frac{V_{u,1}(t)r_{a,u}}{\delta_2(t)}\right)^{-\delta_2(t)} \qquad (11)$$

where $\delta_1(t)$, and $\delta_2(t)$, are obtained from the solution at the $(t-1)$-th iteration as: $\delta_1(t)=\frac{\mathcal{L}_{u,c}R_\mu}{\mathcal{L}_{u,c}R_\mu+V_{u,1}(t-1)r_{a,u}}$ $\delta_2(t)=\frac{V_{u,1}(t-1)r_{a,u}}{\mathcal{L}_{u,c}R_\mu+V_{u,1}(t-1)r_{a,u}}$. Similarly, at every iteration t, we can convert the third, sixth and seventh constraints in (9) into a posynomial using (10) as:

$$\sum_{l\in L_{u,c}} R_l(t)\left(\frac{\mathcal{L}_{u,c}R_\mu}{\delta_3(t)}\right)^{-\delta_3(t)}\left(\frac{V_{u,2}(t)r_{a,u}}{\delta_4(t)}\right)^{-\delta_4(t)}\le 1 \qquad (12)$$

$$\begin{aligned}
&(V_{u,2}f_{u,2}+\sum_{l\in L_{u,c}}(cR_l^3+eR_l+g))\prod_{l\in L_{u,c}}\left(\frac{dR_l(t)^2}{\delta_{5l}(t)}\right)^{-\delta_{5l}(t)}\\
&\cdot\left(\frac{\sum_{l\in L_{u,c}}\tau f_{u,2}(t)}{\delta_6(t)}\right)^{-\delta_6(t)}\le 1 \qquad (13)
\end{aligned}$$

$$\begin{aligned}
&\left(\frac{\tau r' f_{u,1}(t)}{\delta_7(t)}\right)^{-\delta_7(t)}\left(\frac{r'\mathcal{L}'_u dR_\mu^2}{\delta_8(t)}\right)^{-\delta_8(t)}\\
&\cdot\left(r'\mathcal{L}'_u(cR_\mu^3+eR_\mu+g)+f_{u,1}(t)\mathcal{L}R_\mu\right)\le 1, \qquad (14)
\end{aligned}$$

where $\delta_3(t)$, $\delta_4(t)$, $\delta_{5l}(t)$, $\delta_6(t)$ $\delta_7(t)$ and $\delta_8(t)$ are obtained from the solution at the $(t-1)$-th iteration

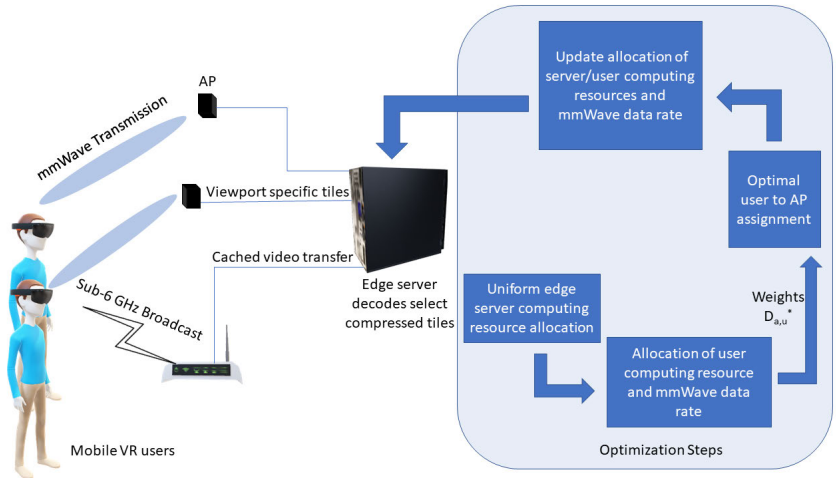

Fig. 6. Optimization framework for resource allocation in our next generation multi-user mobile VR system.

Algorithm 1 GP Based Solution for (9), for a Given $L_{u,r}$

Set $t = 1$,
Initialize $f_{u,1}(t) = f_{u,2}(t) = f_u/2$, Initialize $R_l(t)$
while true **do** ▷ infinite loop
  $t = t + 1$
  Calculate $\delta_1(t)$, $\delta_2(t)$
  Find the optimum $f_{u,1}(t)$, $f_{u,2}(t)$ $D(t)$, $R_l(t)$ solving (15) using GGPLAB [39]
  if $|D(t) - D(t-1)| \leq \epsilon$ **then**
    Break
  end if
end while
The optimal value of the optimization problem (9) is $D(t)$

as: $\delta_3(t) = \frac{\mathcal{L}_{u,c}R_\mu}{\mathcal{L}_{u,c}R_\mu + V_{u,2}(t-1)r_{a,u}}$ $\delta_4(t) = \frac{V_{u,2}(t-1)r_{a,u}}{\mathcal{L}_{u,c}R_\mu + V_{u,2}(t-1)r_{a,u}}$ $\delta_{5l}(t) = \frac{dR_l(t-1)^2}{\sum_{l\in L_{u,c}} dR_l(t-1)^2 + \tau f_{u,2}(t-1)}$ $\delta_6(t) = \frac{\tau f_{u,2}(t-1)}{\sum_{l\in L_{u,c}} dR_l(t-1)^2 + \tau f_{u,2}(t-1)}$, $\delta_7(t) = \frac{\tau r' f_{u,1}(t-1)}{\tau r' f_{u,1}(t-1) + r'\mathcal{L}'_u dR_\mu^2}$, $\delta_8(t) = \frac{r'\mathcal{L}'_u dR_\mu^2}{\tau r' f_{u,1}(t-1) + r'\mathcal{L}'_u dR_\mu^2}$.

Let $D(t) = \sum_{l\in L_{u,r}} p_l^u a_l R_{l,max}^{b_l} + \sum_{l\in L_u \setminus L_{u,r}} p_l^u a_l R_l(t)^{b_l}$. Then, the overall optimization to be solved at iteration t is

$$\begin{aligned}
&\min_{\boldsymbol{R}_{L_{u,r}}(t), f_{u,1}(t), f_{u,2}} (t)\ \ D(t)\\
&\text{s.t. } (12),\ \ ,(13),\ \ (14),\ \ ,\ \ f_{u,1}(t) + f_{u,2}(t) \leq f_u.\\
&\text{s.t.} \frac{\sum_{l\in L_{u,r}} \beta_{l,k_r}}{F_u} \leq V_{u,1},\ \ \frac{\mathcal{L}_{u,c}R_\mu}{r'} \leq V_{u,2},\\
&\text{s.t. } V_{u,1} + \frac{\mathcal{L}_{u,r}b_r}{r_{a,u}} \leq \tau, \qquad (15)
\end{aligned}$$

The above optimization problem is GP and can be solved optimally. The iterative optimization is carried out until $|D(t) - D(t-1)| \leq \epsilon$ with $0 \leq \epsilon \ll 1$. An algorithmic implementation is included in Algorithm 1, which converges to the global solution [38].

Let the optimal expected immersion distortion obtained by Algorithm 1 be $D_{a,u}(L_{u,r}) = D(t)$, which depends on of $L_{u,r}$. Next, we consider integrating the optimal selection of $L_{u,r}$.

2) Raw GOP-Tile Selection: The optimization problem in (9) can be solved optimally in the following manner: In an inner loop, for each possible tile set $L_{u,r}$ in $\boldsymbol{L}_u$, find the mmWave data rate optimization and user computing resource allocation using the procedure described in Section V-B.1, and then in an outer loop, find the best tile set $L_{u,r}$ for which the expected immersion distortion of viewport is smallest. However, this scheme requires to search over $2^{|\boldsymbol{L}_u|}$ possible tile sets. To solve (9) with low complexity, first, we reformulate our problem as follows: Let x_l be an indicator function that denotes whether a tile is sent as encoded or raw, where $x_l = 1$, if tile l is sent as raw, and $x_l = 0$, if the tile is sent as encoded. Thus, the optimization problem in (9) can be reformulated as

$$\begin{aligned}
&\min_{\substack{x_l, R_l, l\in L_u\\ f_{u,1}, f_{u,2}}} \sum_{l\in L_u} x_l p_l^u a_l R_{l,max}^{b_l} + \sum_{l\in L_u} (1-x_l) p_l^u a_l R_l^{b_l},\\
&\text{s.t. } \frac{\sum_{l\in L_u} x_l \beta_{l,k_r}}{F_u} \leq V_{u,1},\ \ \frac{\sum_{l\in L_u}(1-x_l)(R_l - R_\mu)}{r_{a,u}}\\
&\leq V_{u,1},\\
&\text{s.t. } \frac{\sum_{l\in L_u}(1-x_l)(R_l - R_\mu)}{r_{a,u}} \leq V_{u,2},\\
&\text{s.t. } \frac{R_\mu \sum_{l\in L_u}(1-x_l)}{r'} \leq V_{u,2},\ \ V_{u,1}\\
&+ \frac{\sum_{l\in L_u} x_l b_r}{r_{a,u}} \leq \tau,\\
&\text{s.t. } V_{u,2} + \frac{\sum_{l\in L_u}(1-x_l)(cR_l^3 - dR_l^2 + eR_l + g)}{f_{u,2}}\\
&\leq \tau,\\
&\text{s.t. } \frac{\mathcal{L}R_\mu}{r'}\\
&+ \frac{(\mathcal{L} - \sum_{l\in L_u}(1-x_l))(cR_\mu^3 - dR_\mu^2 + eR_\mu + g)}{f_{u,1}}\\
&\leq \tau,\\
&f_{u,1} + f_{u,2} \leq f_u,\ \ x_l \in \{0,1\}, l \in L_u. \qquad (16)
\end{aligned}$$

To solve the above problem efficiently, we first replace the binary constraints above with continuous equivalents

$x_l \in [0,1], l \in \boldsymbol{L}_u$. Furthermore, we introduce v_0 as a slack variable such that $v_0 \geq \sum_{l\in L_u} x_l p_l^u a_l R_{l,max}^{b_l} + \sum_{l\in L_u}(1-x_l) p_l^u a_l R_l^{b_l}$. Then, we can reformulate (16) as:

$$\begin{aligned}
&\min_{\substack{x_l, R_l, l\in L_u \\ f_{u,1}, f_{u,2}}} \quad v_0, \\
&\text{s.t.} \ \frac{\sum_{l\in L_u} x_l p_l^u a_l R_{l,max}^{b_l} + \sum_{l\in L_u} p_l^u a_l R_l^{b_l}}{v_0 + \sum_{l\in L_u} x_l p_l^u a_l R_l^{b_l}} \leq 1, \\
&\text{s.t.} \frac{\sum_{l\in L_u} x_l \beta_{l,k_r}}{F_u} \leq V_{u,1}, \ \frac{\sum_{l\in L_u}(1-x_l)(R_l - R_\mu)}{r_{a,u}} \\
&\leq V_{u,1}, \\
&\text{s.t.} \ \frac{\sum_{l\in L_u}(1-x_l)(R_l - R_\mu)}{r_{a,u}} \leq V_{u,2}, \\
&\text{s.t.} \ \frac{R_\mu \sum_{l\in L_u}(1-x_l)}{r'} \leq V_{u,2}, \quad V_{u,1} + \frac{\sum_{l\in L_u} x_l b_r}{r_{a,u}} \\
&\leq \tau, \\
&\text{s.t.} \ V_{u,2} + \frac{\sum_{l\in L_u}(1-x_l)(cR_l^3 - dR_l^2 + eR_l + g)}{f_{u,2}} \\
&\leq \tau, \\
&\text{s.t.} \ \frac{\mathcal{L} R_\mu}{r'} \\
&+ \frac{(\mathcal{L} - \sum_{l\in L_u}(1-x_l))(cR_\mu^3 - dR_\mu^2 + eR_\mu + g)}{f_{u,1}} \\
&\leq \tau, \\
&f_{u,1} + f_{u,2} \leq f_u, \ x_l \in [0,1], \quad l \in L_u. \qquad (17)
\end{aligned}$$

The above optimization problem can be solved iteratively using a GP method similar to Algorithm 1. It can be shown that the iterative solution method converges to the global solution [38]. The detailed solution process (17) is omitted due to the limited space.

The obtained optimal solution for x_l is continuous. To find the desired raw GOP-tile selection, we pursue the following rounding strategy. We first initialize $L_{u,r}$ as empty. Then, at each step: (i) We find the tile l^* with the biggest value of x_l among the available tile set $L_{u,c}$, and (ii) If the expected immersion distortion reduces, we add l^* to the raw tile set $L_{u,r}$. We continue this process as long as the immersion distortion reduces further, and finally we produce the desired tile set $L_{u,r}^*$, at the end. Then, the allocation of mmWave data rate and user computing resource can be obtained by solving (9), for the given $L_{u,r}^*$. Let $D_{a,u}^*$ denote the optimal expected immersion distortion experienced by user u, i.e., the value of the objective function in (9), enabled by the thereby produced optimal solution. We compute this quantity for every prospective pairing AP a to user u, in order to solve for the optimal AP to user assignment, as explained next. Algorithm 2 summarizes formally our optimization procedure described herein.We assume block fading to model the mmWave and sub-6 GHz channels, i.e., channel gain of each link does not change within the time length of a GoP tile delivery, i.e., in time τ, but may vary from time to time. Therefore, the optimization scheme requires to be executed only after the duration of time τ, if channel gain of any link changes.

Algorithm 2 Optimal Solution of (9), Including a Selection of $L_{u,r}$

1: Set $t = 1$, $L_{u,r} = \emptyset$
2: Solve (17) and obtain $\{x_l\}$, $l \in L_u$.
3: **for** $i = 1 : \mathcal{L}_u$ **do**
4: $l^* = \arg\max_{l\in L_{u,c}} x_l$
5: Solve (9) with raw tile set $L_{u,r} \cup l^*$ using the procedure described in Section V-B1 and obtain $D_{a,u}(L_{u,r} \cup l^*)$.
6: **if** $D_{a,u}(L_{u,r} \cup l^*) <= D_{a,u}(L_{u,r})$ **then**
7: $L_{u,r} = L_{u,r} \cup l^*$
8: **else**
9: break
10: **end if**
11: **end for**
12: Output raw tile set $L_{u,r}^*$
13: Solve (9) with fixed raw GOP-tile set $L_{u,r}^*$ using procedure described in Section V-B1. Compute the respective objective $D_{a,u}^*$.

Complexity of Algorithm 2: Here, we analyze the computational complexity of proposed Algorithm 2. CVX is used to solve the GP problems with the interior point method in steps 2 and 5. The number of required iterations to solve (17) is $\frac{\log(\mathcal{L}_u + 9/t_{0,1}\epsilon)}{\log \xi}$ where $\mathcal{L}_u + 9$ is the total number of constraints, $t_{0,1}$ is the initial point to approximate the accuracy of interior point method for solving (17) using GP, $0 < \epsilon < 1$ is the stopping criterion for interior point method, and ξ is used for updating the accuracy of interior point method [40]. Furthermore, it can be shown that for each iteration, the number of computations required to convert the non-convex constraints into convex is on the order of $\mathcal{L}_u$. Therefore, the total number of computations required to solve (17) is on the order of $\mathcal{L}_u \times \frac{\log(\mathcal{L}_u + 9/t_{0,1}\epsilon)}{\log \xi}$. The total number of constraints in (15) is 7 and the number of computations required to compute (13) is on the order of $\mathcal{L}_u$ at each iteration. Therefore, it can be observed that the total number of computations required to solve (15) is on the order of $\mathcal{L}_u \times \frac{\log(7/t_{0,2}\epsilon)}{\log \xi}$ where $t_{0,2}$ is the initial point to approximate the accuracy of interior point method for (15). The step 5 is repeated for a maximum of $\mathcal{L}_u$ times, for which the computation requirement is on the order of $\mathcal{L}_u^2 \times \frac{\log(7/t_{0,2}\epsilon)}{\log \xi}$. Therefore, the total number of computations required for Algorithm 2 is on the order of $\mathcal{L}_u \times \frac{\log(\mathcal{L}_u + 9/t_{0,1}\epsilon)}{\log \xi} + \mathcal{L}_u^2 \times \frac{\log(7/t_{0,2}\epsilon)}{\log \xi}$.

C. Optimal user-to-Access Point Assignment

The optimal user to AP assignment can be identified by searching over all possible assignments Π. However, this requires searching over $(M+N)!/M!$ prospective assignments, where $M = |U|$ is the number of users and $N = |A|$ is the number of APs. We explore a lower-complexity alternative solution that leverages graph-theoretic concepts.

We begin by reviewing some concepts of bipartite graph theory matching [41], [42]. 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 $\mathcal{V}^2$. Figure 7(b) shows

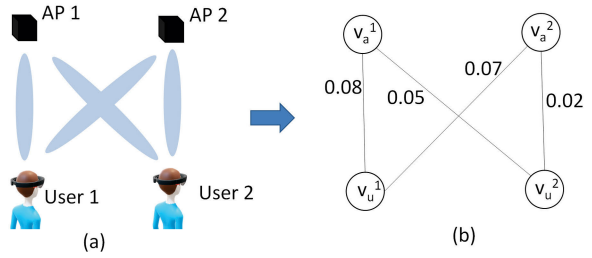

Fig. 7. Example of a weighted bipartite graph for the network with two APs and two users.

an example of a bipartite graph with two sets of vertices, $\mathcal{V}^1 = \{v_a^1, v_a^2\}$ and $\mathcal{V}^2 = \{v_u^1, v_u^2\}$, and an edge set $\mathcal{E} = \left\{(v_a^1, v_u^1), (v_a^1, v_u^2), (v_a^2, v_u^1), (v_a^2, v_u^2)\right\}$ that correspond to a user to AP assignment problem illustrated in Figure 7(a). 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 Figure 7(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 $\omega_{(v_a^1, v_u^2)} = D^*_{a,u}$. This is the minimum expected immersion distortion experienced by user u when assigned to AP a. In Section V-B, we describe how each $D^*_{a,u}$ can be obtained. Thus, we can construct the respective bipartite graph for the actual problem under consideration.[1]

Leveraging the development heretofore, we formulate the user to AP assignment subproblem from (8) as a bottleneck matching (BM) problem for the graph defined by the maximum matching whose largest edge weight is as small as possible, i.e., $\min_{\phi \in \Phi} \max_{(v_a^1, v_u^2) \in \phi} \omega_{(v_a^1, v_u^2)}$, where Φ contains all possible maximum matchings. Note that Φ is directly related to Π such that each maximum matching $\phi \in \Phi$ corresponds to an user to AP assignment in Π.

For the graph in Figure 7(b), the bottleneck matching is $\{(v_a^1, v_u^2), (v_a^2, v_u^1)\}$ and thus the corresponding assignment is: User 2 assigned to AP 1, and User 1 assigned to AP 2. The constructed bipartite graph has MN edges and $N + M$ vertices. We solve the BM problem optimally using the algorithm proposed in [42] with complexity $\mathcal{O}\left(N^{2.5}\right)$. Moreover, to construct a graph with MN edges the induced time complexity is $\mathcal{O}\left(MN\right)$. Therefore, the user to AP assignment problem can be solved optimally with complexity $\mathcal{O}\left(N^{2.5}\right)$.

D. Joint Transmission and User/Edge Server Computing Resources Allocation

In Section V-B, as part of the optimization carried out therein, we identified the optimal enhancement GOP-tile subset $L_{u,r}$ that should be transmitted as raw data over the mmWave link of a given user and AP pairing (u, a). In Section V-C, we identified the optimal user to AP assignment π^*. Given these discrete optimization developments, we can (re)solve jointly now the optimal allocation of user and edge server computing resources, and mmWave link data rate across the compressed enhancement layer GOP-tile subset $L_u \setminus L^*_{u,r}$. Concretely, we investigate the joint allocation of these three system resources by solving (8), for given π^* and $L^*_{u,r}, \forall u$.

We pursue a solution to this optimization problem by reformulating it first as GP using the single condensation method, analogously to the analytical steps carried out in Section V-B.1. We then solve the problem reformulation via an iterative optimization method that we design equivalently to Algorithm 1. A high level illustration of our overall optimization framework described throughout SectionV herein is included in Figure 6.

VI. Experimental Evaluation

We carry out a comprehensive experimental evaluation to assess the performance of our system framework. We measure the delivered VR immersion fidelity as the inverse of the respective distortion quantity, using $10 \log_{10}(255^2 / \sum_{l \in L} p_l^u D_l)$, commonly known as the Peak Signal-to-Noise ratio (PSNR). We compare the performance of the proposed strategy with the following techniques.

1) *Reference*: State-of-the-art method that integrates the latest video streaming standard MPEG DASH [43], [44], to deliver the 360° content over sub-6 GHz, given the same system constraints.
2) *Proposed-Rand*: Here, mmWave and edge computing technologies are used, but an AP is randomly assigned to each user, and raw GoP tileset is randomly selected for each user from its expected viewport tiles. The data rate allocation for encoded enhancement layer tiles and computation resource allocation at the users and the edge server are obtained by solving an iterative GP algorithm similar to Algorithm 1. Proposed-Rand can be regarded as lower complexity implementation technique for our proposed framework.

The immersion distortion modelling parameters of each GoP tile of 'Runner' and 'Basketball' 360° video sequences is experimentally calculated in our previous work [34]. The viewpoint likelihood distribution was compiled based on traces of VR user head movements that we collected [34], [36]. Oculus Rift VR headset was used and navigation actions in real time was recorded using the OpenTrack software. Modelling the relationship between CPU computing cycle to tile data rate is conducted on an Intel Core i7 computer, and modelling parameters c,d, e and g are obtained. All these parameters are then used to evaluate our system framework via numerical simulations. In our simulation experiments, five users are uniformly distributed in a $5m \times 5m$ square room VR arcade. A sub-6 GHz router and five mmWave APs serve the users. The mmWave APs and the sub-6 GHz router, linked to an edge computing server, are placed vertically along the room boundaries. For 360° content delivered to the mobile VR users, we leveraged the 'Runner' and 'Basketball' 360° video

[1]The edge weights shown in Figure 7(b) are produced in this fashion, for the example network from Figure 7(a).

TABLE II
MAJOR SIMULATION PARAMETERS

Parameter	Value
f_u	3 GHz
$R_{l,max}$	15 Mbps
M	5
N	10
τ	1 sec.
F	150 GHz
N_0	-147 dBM/Hz
ϵ	10^{-5}

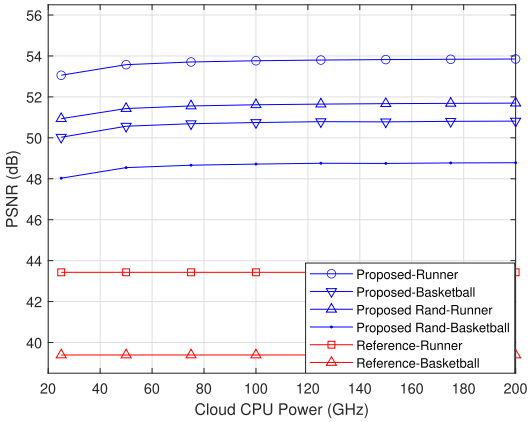

Fig. 8. PSNR performance versus edge server computing power.

sequences captured at 8K spatial resolution and 30 frames per second temporal frame rate [45]. Our major simulation parameters are summarized in Table II.

A. Delivered Immersion Fidelity Vs. Edge Server Computing Resource

In Figure 8, we show the PSNR performance of the proposed system when the edge server computing power varies from 75 GHz to 175 GHz. The mmWave network links in the system exhibit diverse data rates in the range of 600 – 800 Mbps. We can observe that as the edge server's computing power F increases and $F \leq 50$ GHz, more enhancement GOP-tiles can be decoded within a small computing latency at the edge server and can be delivered using mmWave communication, which in turn reduces the computing latency at the user. Thus, the mmWave AP can send a higher number of compressed enhancement GOP-tiles, encoded at higher data rates, which improves the PSNR. If the edge server's computing power increases beyond 50 GHz, the improvement in PSNR performance is negligible. The reason is that the mmWave rate is a bottleneck for the system, and it cannot support the delivery of more raw tiles while also transmitting compressed enhancement GOP-tiles at high rates. For $F = 75$ GHz, the proposed strategy achieves 8.8 and 10 dB PSNR improvement over the reference method for the 'Runner' and 'Basketball' 360° video sequences, respectively. Furthermore, our proposed scheme in which all variables are jointly optimized provides 2 dB PSNR improvement over Proposed-Rand method in which raw GOP-tileset selection from expected viewport GOP-tiles and AP

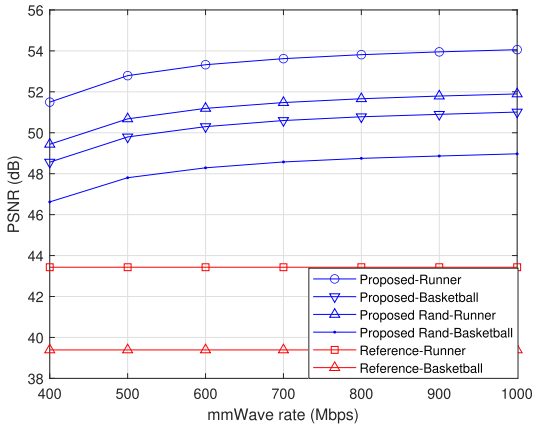

Fig. 9. PSNR performance versus mmWave network link data rate for $F = 150$ GHz.

to user assignment are random. The high-performance gain of the proposed method compared to reference method stems from the facts that scalable multi-layer tile-based 360° content enables dual sub-6 GHz and mmWave transmission which results in delivering high-quality VR content to the users and also due to the synergistic integration edge computing along with mmWave technology that allows minimizing computation delay at the VR-headset, improving further immersion fidelity at the user-end. These significant performance advances will considerably enhance the remote scene immersion fidelity and quality of experience delivered to the mobile users in our VR arcade system.

B. Delivered Immersion Fidelity Vs. mmWave Transmission Rate

Next, in Figure 9 we explore the PSNR performance of our system, when the data rate of the mmWave network links in the system is uniform, and is progressively increased from 400 Mbps to 1 Gbps. The edge server's computing power has been fixed to 150 GHz in these experiments. We can see that as the mmWave transmission data rate increases, the PSNR of the proposed system increases, as expected. In particular, as the data rate increases, more raw enhancement GOP-tiles can be transmitted. In consequence, this will reduce the computing time at the user, which together with the higher mmWave network link data rate, will enable transmitting the remaining compressed enhancement GOP-tiles, encoded at higher data rates. Both of these advances will augment the delivered immersion fidelity for the VR user. Similarly to the earlier results, we observe from Figure 9 a significant performance improvement of 7 and 8 dB over the reference method, for the 'Runner' and 'Basketball' 360° video sequences, respectively. These performance gains considerably advance the state-of-the-art.

C. Delivered Immersion Fidelity Vs. User Computing Resource

In Figure 10, we explore the PSNR performance of our system, when the computing power of the VR user's mobile device increases from 3 GHz to 7 GHz. We can observe

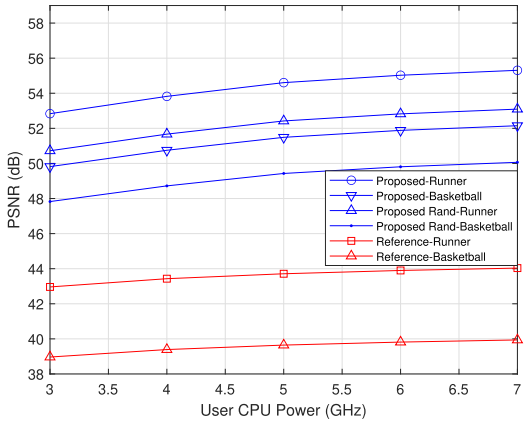

Fig. 10. PSNR performance versus user computing power for $F = 150$ GHz.

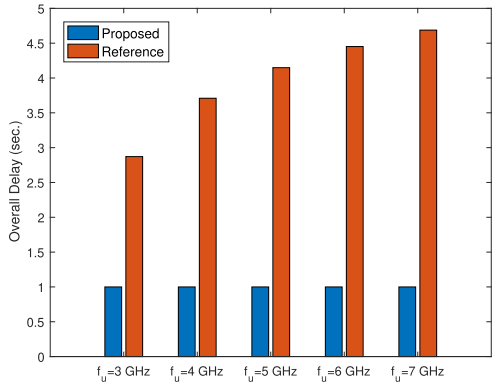

Fig. 11. End-to-end system delay vs. user computing power for $F = 150$ GHz and same PSNR performance.

that as the user's computing power increases, the time delay for decoding GOP-tiles of the 360° content at the user end decreases. Thus, higher encoding data rate compressed enhancement GOP-tiles can be transmitted within the required system latency constraints, and hence, in turn, the immersion fidelity delivered to the mobile VR user increases.

D. Latency Performance Vs. User Computing Resource

In Figure 11, we investigate the end-to-end delay induced by our system, when streaming the 'Runner' 360° video, vs. the available computing power at the user. We hypothetically consider that the reference method can deliver the same immersion fidelity, as our own system, and measure what would be its induced end-to-end delay in that case. In particular, towards this objective, we consider that the reference method can hypothetically transmit all GOP-tiles encoded at the same data rate as in the case of our system. We can observe from Figure 11 that as the user computing power increases from 3 GHz to 7 GHz, our system is able to maintain the required system latency of 1 second, while enabling increasingly higher immersion fidelity for the mobile user, by facilitating its more plentiful computing resource. On the other hand, due to the need to decompress at the user all GOP-tiles that are transmitted encoded at much higher data rate in this case, in order to provide the same immersion fidelity as our system, the reference method is struggling to maintain the required system latency. As observed from Figure 11, its end-to-end latency becomes thereby 2.5 - 4.5 times higher than the required constraint. This in turn would dramatically penalize the immersion quality of experience of the mobile VR user, as it would considerably reduce the interactive nature of the VR application.

TABLE III

EXPECTED NUMBER OF RAW ENHANCEMENT GOP-TILES SELECTED FOR MMWAVE TRANSMISSION VS. EDGE SERVER COMPUTING POWER AND MMWAVE TRANSMISSION DATA RATE. USER COMPUTING POWER IS FIXED TO $f_u = 3$ GHZ

F	400 Mbps	600 Mbps	800 Mbps	1000 Mbps
50 GHz	0	0	0.8	1.2
100 GHz	0	0	0.9	1.4
150 GHz	0	0.4	1.1	1.7
200 GHz	0	0.5	1.3	2.0

E. Expected Number of Raw Enhancement GOP-Tiles Selected for Transmission

Finally, in Table III, Table IV, and Table V, we investigate the expected number of raw enhancement GOP-tiles selected for mmWave transmission in our system, in the case of the 'Runner' 360° content. In particular, a positive non-integer valued entry in Table III, associated with a given mmWave transmission data rate and edge server computing power pair, can be explained with the following examples.

(i) if $F = 150$ GHz and mmWave rate is 600 Mbps, an expected number of transmitted raw tiles of 0.2 can occur if for one of the five users, one enhancement GOP-tile is transmitted as raw data. For the other users, all enhancement GOP-tiles are transmitted compressed. (ii) if $F = 200$ GHz and mmWave rate is 800 Mbps, an expected number of transmitted raw tiles of 1.2 can occur if for one of the five users, two enhancement GOP-tiles are transmitted as raw data. For the other four users, one enhancement GOP-tile is transmitted as raw data. Similar analogy can be drawn for the alike valued entries in Table IV and Table V.

It can be observed from Table III that as the mmWave transmission data rate and edge server computing power increase, a higher number of raw enhancement GOP-tiles are selected by our optimization framework, to augment the delivered immersion fidelity. This is because with the increase in mmWave data rate and edge server computing power, a higher number of enhancement GOP-tiles can be decompressed at the edge server and transmitted as raw data over the mmWave link, within a short time interval. In consequence, then only a smaller number of compressed enhancement GOP-tiles, encoded at higher data rate, would need to be delivered, which would lower the decoding latency induced at the user. Both of these advances will contribute to higher immersion fidelity delivered to the VR user, while maintaining the required system latency.

We can observe from Table IV that a higher number of raw enhancement GOP-tiles are likewise selected for transmission, as the mmWave transmission data rate and user computing power increase, again to augment the immersion fidelity delivered to the user. This outcome stems from

TABLE IV

EXPECTED NUMBER OF RAW ENHANCEMENT GOP-TILES SELECTED FOR MMWAVE TRANSMISSION VS. USER COMPUTING POWER AND MMWAVE TRANSMISSION DATA RATE. EDGE SERVER COMPUTING POWER IS FIXED TO $F = 150$ GHZ

f_u	400 Mbps	600 Mbps	800 Mbps	1000 Mbps
3 GHz	0	0.2	1.0	1.6
4 GHz	0	0.2	1.0	1.8
5 GHz	0	0.3	1.2	1.9
6 GHz	0	0.3	1.3	1.9

TABLE V

EXPECTED NUMBER OF RAW ENHANCEMENT GOP-TILES SELECTED FOR MMWAVE TRANSMISSION VS. EDGE SERVER COMPUTING POWER AND USER COMPUTING POWER. MMWAVE TRANSMISSION DATA RATE IS FIXED TO 800 MBPS

F	$f_u = 3$ GHz	$f_u = 4$ GHz	$f_u = 5$ GHz	$f_u = 6$ GHz
50 GHz	0.8	0.8	0.9	1.0
100 GHz	0.9	0.9	1.0	1.2
150 GHz	1.1	1.1	1.2	1.3
200 GHz	1.2	1.3	1.5	1.6

reasons equivalent to those discussed earlier in the context of the results presented in Table III. In particular, the higher user computing power enables decoding faster compressed enhancement GOP-tiles delivered to the user, i.e., with lower induced delay. This in turn will leave more of the end-to-end system delay constraint available to be consumed by mmWave transmission, which coupled with the higher mmWave transmission data rate, enables sending a higher number of raw enhancement GOP-tiles.

Similarly, Table V informs that as the edge server computing power and user computing power increase, again, a higher number of raw enhancement GOP-tiles are selected for mmWave transmission. This is because with an increase in edge server computing power and user computing power, the compressed enhancement GOP tiles can be decoded faster, thereby allowing for a higher number of raw enhancement GOP-tiles to be transmitted over the mmWave links, within the maximum tolerable system delay. Finally, it can be observed from Table III and Table IV that if the mmWave transmission data rate is limited, our optimization framework selects all enhancement GOP-tiles to be transmitted compressed.

VII. CONCLUSION

We have investigated a novel multi-user mobile VR system for streaming 8K scalable 360° video that enables high reliability and immersion fidelity, and low interactive latency, via the synergistic integration of embedded multi-layer 360° tiling, dual millimeter wave (mmWave) and sub-6 GHz transmission, and edge computing capability. High rate directed mmWave links were studied to send viewport-specific enhancement layers of the 360° content to the individual VR users, to augment the delivered remote scene immersion fidelity. Sub-6 GHz broadcast of the base layer of the entire 360° panorama to all users is carried out, to augment the application reliability. The viewport-specific enhancements could comprise compressed and raw 360° tiles, decoded first at the edge server. We explored the joint optimization of the mmWave access point to user association, the choice of 360° tiles to be transmitted decompressed, the allocation of mmWave data rate across the compressed tiles in a viewport-specific enhancement, and the allocation of computing resources at the edge server and user devices. Our aim was to maximize the minimal delivered immersion fidelity across all VR users, given transmission, latency, and computing constraints. We have introduced analytical characterizations of the rate-distortion trade-offs across the spatiotemporal 360° panorama and the computing power required to decompress 360° tiles, to facilitate our analysis and problem formulation. We explored a solution that comprises multiple geometric programming algorithms and an intermediate step of graph-theoretic VR user to mmWave access point assignment. Our results demonstrate that our framework can enable a significant improvement in delivered VR user immersion fidelity (8 dB to 10 dB) and spatial resolution (8K vs. 4K), over a state-of-the-art reference method that leverages sub-6 GHz transmission only. The high-performance gain stems from the following facts: (*i.*) Scalable multi-layer tile-based 360° content enables dual sub-6 GHz and mmWave transmission which results in delivering high quality VR content to the users (*ii.*) By synergistic integration edge computing along with the above-mentioned technologies allows minimizing computation delay at the VR-headset, improving further immersion fidelity at the user-end. We have also shown that an increasing number of raw 360° enhancement GOP-tiles are sent, as the mmWave link data rate or the edge server/user computing power increases, exploring rigorously in this context the fundamental interplay between computing/communication capabilities, end-to-end system latency, and delivered VR immersion fidelity. Finally, we demonstrated that in order to hypothetically deliver the same immersion fidelity, the reference method would incur a much higher (2.5-4.5x) system latency.

Here we discuss two different possibilities of extending our work in future. Firstly, an objective metric which is a concave and increasing function of encoding rate, may not be always modeled directly as a polynomial function of encoding rate, e.g., in [46] and [47]. Even then, the proposed geometric programming method can still be used as most of the continuous functions can be well approximated as the difference of two convex functions [48]. Secondly, there exist different quality of experiences (QoEs) for 360^o video applications, e.g., tile quality smoothness (in terms of change in encoding rate) spatially or temporally [47], [49], [50]. Spatial tile quality smoothness or temporal tile quality smoothness can be expressed as a polynomial function of the encoding rate of relevant tiles [47], [49], [50]. Therefore, if there is a specific spatial/temporal smoothness requirement, a new constraint can be added. The proposed geometric programming strategy can still be easily implemented in such a scenario, as the constraint is a polynomial function of the encoding rates of the relevant tiles. Furthermore, the results demonstrated in this paper provide an upper-bound for the cases where a constraint on video smoothness is required, and therefore the proposed solution provides a guiding insight for designing an effective system.

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] J. Chakareski, "UAV-IoT for next generation virtual reality," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5977–5990, Dec. 2019.

[5] E. Cuervo, K. Chintalapudi, and M. Kotaru, "Creating the perfect illusion: What will it take to create life-like virtual reality headsets?" in *Proc. Int. Workshop Mobile Comput. Syst. Appl. (HotMobile)*, Tempe, AZ, USA, Feb. 2018, pp. 7–12.

[6] M. Champel, T. Stockhammer, T. Fautier, E. Thomas, and R. Koenen, "Quality requirements for VR," in *Proc. 116th MPEG Meeting ISO/IEC JTC1/SC29/WG11*, Oct. 2016.

[7] M. Hosseini and V. Swaminathan, "Adaptive 360 VR video streaming: Divide and conquer," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2016, pp. 107–110.

[8] M. Graf, C. Timmerer, and C. Mueller, "Towards bandwidth efficient adaptive streaming of omnidirectional video over HTTP: Design, implementation, and evaluation," in *Proc. 8th ACM Multimedia Syst. Conf.*, New York, NY, USA, Jun. 2017, pp. 261–271, doi: 10.1145/3083187.3084016.

[9] A. Zare, A. Aminlou, M. M. Hannuksela, and M. Gabbouj, "HEVC-compliant tile-based streaming of panoramic video for virtual reality applications," in *Proc. 24th ACM Int. Conf. Multimedia*, New York, NY, USA, Oct. 2016, pp. 601–605, doi: 10.1145/2964284.2967292.

[10] R. Skupin, Y. Sanchez, D. Podborski, C. Hellge, and T. Schierl, "HEVC tile based streaming to head mounted displays," in *Proc. CCNC*, Jan. 2017, pp. 613–615.

[11] Samsung. *Gear VR*. [Online]. Available: http://www.samsung.com/us/explore/gear-vr/

[12] Google. *Daydream*. [Online]. Available: https://vr.google.com/daydream/

[13] L. Sun et al., "A two-tier system for on-demand streaming of 360 degree video over dynamic networks," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 43–57, Mar. 2019.

[14] X. Zhang, X. Hu, L. Zhong, S. Shirmohammadi, and L. Zhang, "Cooperative tile-based 360° panoramic streaming in heterogeneous networks using scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 1, pp. 217–231, Jan. 2020.

[15] J. Chakareski, "Viewport-adaptive scalable multi-user virtual reality mobile-edge streaming," *IEEE Trans. Image Process.*, vol. 29, pp. 6330–6342, 2020.

[16] C. Chaccour, M. N. Soorki, W. Saad, M. Bennis, and P. Popovski, "Risk-based optimization of virtual reality over terahertz reconfigurable intelligent surfaces," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[17] M. Chen, O. Semiari, W. Saad, X. Liu, and C. Yin, "Federated echo state learning for minimizing breaks in presence in wireless virtual reality networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 177–191, Jan. 2020.

[18] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Netw.*, vol. 32, no. 2, pp. 78–84, Mar. 2018.

[19] C. Perfecto, M. S. Elbamby, J. D. Ser, and M. Bennis, "Taming the latency in multi-user VR 360°: A QoE-aware deep learning-aided multicast framework," *IEEE Trans. Commun.*, vol. 68, no. 4, pp. 2491–2508, Apr. 2020.

[20] S. Sukhmani, M. Sadeghi, M. Erol-Kantarci, and A. El Saddik, "Edge caching and computing in 5G for mobile AR/VR and tactile internet," *IEEE Multimedia Mag.*, vol. 26, no. 1, pp. 21–30, Jan./Mar. 2019.

[21] Y. Sun, Z. Chen, M. Tao, and H. Liu, "Communications, caching, and computing for mobile virtual reality: Modeling and tradeoff," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 7573–7586, Nov. 2019.

[22] M. Chen, W. Saad, and C. Yin, "Virtual reality over wireless networks: Quality-of-service model and learning-based resource management," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5621–5635, Nov. 2018.

[23] H. Singh, J. Oh, C. Kweon, X. Qin, H. R. Shao, and C. Ngo, "A 60 GHz wireless network for enabling uncompressed video communication," *IEEE Commun. Mag.*, vol. 46, no. 12, pp. 71–78, Dec. 2008.

[24] L. Liu et al., "Cutting the cord: Designing a high-quality untethered VR system with low latency remote rendering," in *Proc. ACM MobiSys*, 2018, pp. 68–80.

[25] 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 Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.

[26] S. Gupta, J. Chakareski, and P. Popovski, "Millimeter wave meets edge computing for mobile VR with high-fidelity 8k scalable 360° video," in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2019, pp. 1–6.

[27] J. Chakareski and S. Gupta, "Multi-connectivity and edge computing for ultra-low-latency lifelike virtual reality," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2020, pp. 1–6.

[28] H. Xu, V. Kukshya, and T. S. Rappaport, "Spatial and temporal characteristics of 60-GHz indoor channels," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 3, pp. 620–630, Apr. 2002.

[29] 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 Trans. Wireless Commun.*, vol. 13, no. 12, pp. 7072–7085, Dec. 2014.

[30] M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, "An efficient uplink multi-connectivity scheme for 5G millimeter-wave control plane applications," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6806–6821, Oct. 2018.

[31] X. Corbillon, G. Simon, A. Devlic, and J. Chakareski, "Viewport-adaptive navigable 360-degree video delivery," in *Proc. IEEE Int. Conf. Commun.*, May 2017, pp. 1–7.

[32] X. Corbillon, A. Devlic, G. Simon, and J. Chakareski, "Optimal set of 360-degree videos for viewport-adaptive streaming," in *Proc. MM*, New York, NY, USA, 2017, pp. 943–951, doi: 10.1145/3123266.3123372.

[33] J. M. Boyce, Y. Yan, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high efficiency video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 20–34, Jan. 2016.

[34] J. Chakareski, R. Aksu, X. Corbillon, G. Simon, and V. Swaminathan, "Viewport-driven rate-distortion optimized 360° video streaming," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.

[35] X. Corbillon, F. De Simone, and G. Simon, "360-degree video head movement dataset," in *Proc. ACM Multimedia Syst. Conf.*, Taiwan, Jun. 2017, pp. 199–204.

[36] R. Aksu, J. Chakareski, and V. Swaminathan, "Viewport-driven rate-distortion optimized scalable live 360° video network multicast," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2018, pp. 1–6.

[37] Y. Sun, A. Lu, and L. Yu, "Weighted-to-spherically-uniform quality evaluation for omnidirectional video," *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1408–1412, Sep. 2017.

[38] G. Xu, "Global optimization of signomial geometric programming problems," *Eur. J. Oper. Res.*, vol. 233, no. 3, pp. 500–510, 2014.

[39] *GGPLAB: A Simple MATLAB Toolbox for Geometric Programming*. [Online]. Available: http://www.stanford.edu/boyd/ggplab/

[40] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[41] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*. Philadelphia, PA, USA: SIAM, 2009.

[42] A. P. Punnen and K. P. K. Nair, "Improved complexity bound for the maximum cardinality bottleneck bipartite matching problem," *Discrete Appl. Math.*, vol. 55, no. 1, pp. 91–93, Oct. 1994.

[43] M. Graf, C. Timmerer, and C. Mueller, "Towards bandwidth efficient adaptive streaming of omnidirectional video over HTTP: Design, implementation, and evaluation," in *Proc. 8th ACM Multimedia Syst. Conf.*, Taiwan, Jun. 2017, pp. 261–271.

[44] I. Sodagar, "The MPEG-DASH standard for multimedia streaming over the internet," *IEEE Multimedia*, vol. 18, no. 4, pp. 62–67, Apr. 2011.

[45] 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.

[46] Y. Liu, S. Dey, F. Ulupinar, M. Luby, and Y. Mao, "Deriving and validating user experience model for DASH video streaming," *IEEE Trans. Broadcast.*, vol. 61, no. 4, pp. 651–665, Dec. 2015.

[47] Z. Jiang et al., "A hierarchical buffer management approach to rate adaptation for 360-degree video streaming," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 2157–2170, Feb. 2020.

[48] C. J. Karcher, "Data fitting with signomial programming compatible difference of convex functions," *Optim. Eng.*, pp. 1–15, Apr. 2022.
[49] N. Kan, J. Zou, C. Li, W. Dai, and H. Xiong, "RAPT360: Reinforcement learning-based rate adaptation for 360-degree video streaming with adaptive prediction and tiling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 1607–1623, Mar. 2022.
[50] J. Dai, G. Yue, S. Mao, and D. Liu, "Sidelink-aided multiquality tiled 360° virtual reality video multicast," *IEEE Internet Things J.*, vol. 9, no. 6, pp. 4584–4597, Mar. 2022.

Sabyasachi Gupta received the M.Tech. degree from the National Institute of Technology, Durgapur, India, and the Ph.D. degree from the Indian Institute of Technology Delhi, New Delhi, India. He has held research positions with the University of Pompeu Fabra, Barcelona, Spain, the University of Alabama, Tuscaloosa, AL, USA, and Southern Methodist University (SMU), Dallas, USA. He is currently a Senior Simulation Specialist with the Nokia Standards Department, Nokia Networks, Bengaluru, India. His research interests include resource allocation for wireless networks, mobile-edge computing networks, investigating application of optimization technique, machine learning, and graph theory for wireless communication. He was a recipient of the Institute Gold Medal from NIT Durgapur in 2010.

Jacob Chakareski (Senior Member, IEEE) received the joint Ph.D. degree in electrical and computer engineering from Rice University and Stanford University. He is currently an Associate Professor with the College of Computing, New Jersey Institute for Technology, where he holds the Panasonic Chair of Sustainability and directs the Laboratory for AI-Enabled XR Network Systems and Societal Applications. His research interests include next generation virtual and augmented reality systems, the UAV IoT sensing and networking, fast reinforcement learning, 5G wireless edge computing and caching, ubiquitous immersive communication, and societal applications. He received the Adobe Data Science Faculty Research Award in 2017 and 2018, the Swiss NSF Career Award Ambizione (2009), the AFOSR Faculty Fellowship in 2016 and 2017, and best paper awards at ICC 2017 and MMSys 2021. He held research appointments with Microsoft, HP Laboratories, and EPFL, and served on the Advisory Board of Frame Inc. His research has been supported by NSF, NIH, AFOSR, Adobe, Tencent Research, NVIDIA, and Microsoft. For more information visit the link (www.jakov.org).

Petar Popovski (Fellow, IEEE) received the Dipl.-Ing. and M.Sc. degrees in communication engineering from the University of Sts. Cyril and Methodius in Skopje and the Ph.D. degree from Aalborg University in 2005. He is currently a Professor at Aalborg University, where he heads the Section on Connectivity and the Visiting Excellence Chair of the University of Bremen. He has authored the book *Wireless Connectivity: An Intuitive and Fundamental Guide* (Wiley, 2020). His research interests are in the area of wireless communication and communication theory. He received an ERC Consolidator Grant (2015), the Danish Elite Researcher Award (2016), IEEE Fred W. Ellersick Prize (2016), IEEE Stephen O. Rice Prize (2018), Technical Achievement Award from the IEEE Technical Committee on Smart Grid Communications (2019), the Danish Telecommunication Prize (2020), and Villum Investigator Grant (2021). He was a Member at Large at the Board of Governors in IEEE Communication Society (2019–2021). He is currently an Editor-in-Chief of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. He also serves as the Vice Chair of the IEEE Communication Theory Technical Committee and the Steering Committee of IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING. He was the General Chair for IEEE SmartGridComm 2018 and IEEE Communication Theory Workshop 2019.