Understanding the Impact of Wi-Fi Configuration on Volumetric Video Streaming Applications

Umakant Kulkarni*
Hewlett Packard Labs
ukulkarn@purdue.edu

Khaled Diab
Hewlett Packard Labs
khaled.diab@hpe.com

Shivang Aggarwal
Hewlett Packard Labs
shivang.aggarwal@hpe.com

Lianjie Cao
Hewlett Packard Labs
lianjie.cao@hpe.com

Faraz Ahmed
Hewlett Packard Labs
faraz.ahmed@hpe.com

Puneet Sharma
Hewlett Packard Labs
puneet.sharma@hpe.com

Sonia Fahmy
Purdue University
fahmy@purdue.edu

ABSTRACT

Emerging multimedia applications often use a wireless LAN (Wi-Fi) infrastructure to stream content. These Wi-Fi deployments vary vastly in terms of their system configurations. In this paper, we take a step toward characterizing the Quality of Experience (QoE) of volumetric video streaming over an enterprise-grade Wi-Fi network to: (i) understand the impact of Wi-Fi control parameters on user QoE, (ii) analyze the relation between Quality of Service (QoS) metrics of Wi-Fi networks and application QoE, and (iii) compare the QoE of volumetric video streaming to traditional 2D video applications. We find that Wi-Fi configuration parameters such as channel width, radio interface, access category, and priority queues are important for optimizing Wi-Fi networks for streaming immersive videos.

CCS CONCEPTS

• **Networks** → **Network measurement**; **Wireless local area networks**; • **Information systems** → **Multimedia streaming**.

KEYWORDS

Wi-Fi; Volumetric Video Streaming; Quality of Service; Quality of Experience; Access Category; Channel Width; Radio Interface; Priority Queues

ACM Reference Format:
Umakant Kulkarni, Khaled Diab, Shivang Aggarwal, Lianjie Cao, Faraz Ahmed, Puneet Sharma, and Sonia Fahmy. 2023. Understanding the Impact of Wi-Fi Configuration on Volumetric Video Streaming Applications. In *Emerging Multimedia Systems (EMS '23), September 10, 2023, New York, NY, USA*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3609395.3610599

*Also with Purdue University.

EMS '23, September 10, 2023, New York, NY, USA

ACM ISBN 979-8-4007-0303-4/23/09.
https://doi.org/10.1145/3609395.3610599

1 INTRODUCTION

Streaming immersive videos over the Internet has been gaining popularity over the past decade. Examples of immersive videos include panoramic videos [3, 8, 10, 25] and, more recently, *volumetric videos* [9, 19, 35]. This steady growth in popularity can be attributed to several factors. First, immersive videos enable users to *interact* with scenes by navigating to different perspectives and angles. Second, substantial progress has been made in accessible rendering devices, such as head-mounted displays (HMDs), since the creation of the Sword of Damocles VR system in 1968. Third, the Internet, including its core and access networks, has significantly improved since its infancy, in terms of hardware, protocols, and implementations.

One key area of improvement has been in wireless local area network technologies such as Wi-Fi. Since the ratification of the first IEEE 802.11 standard in 1997 [15], which offered a maximum data rate of 2 Mbps, there have been tremendous advances in Wi-Fi link capabilities. Wi-Fi links have had a multi-fold increase in their capacities from 802.11n (WiFi 4) in 2009, introducing the simultaneous use of multiple antenna (MIMO), to 2013, when 802.11ac (WiFi 5) was introduced with wider channels and beamforming, to most recently the advent of 802.11ax (WiFi 6) in 2019 which offered denser modulation and coding schemes (MCSes) and OFDMA, offering theoretical PHY data rates of several Gbps. Today, most video streaming, both traditional and immersive, is performed indoors where a client is connected to a server via a Wi-Fi access point. Specifically, in the context of immersive video streaming applications, due to the extensive use of HMDs, having a wired connection from the HMD to the server can be detrimental to the user experience, not only because it limits the user's mobility, but also because the wire can actually be a tripping hazard [1, 20].

Streaming immersive videos over Wi-Fi is continuing to impose significant requirements, however. This is due to the high bandwidth demand, stringent latency constraints, and the dynamic and heterogeneous interactions with these videos. For example, volumetric videos provide 6 degrees of freedom (6DoF) to a user, three translational and three rotational, compared to only rotational 3DoF (yaw, pitch, and roll) for panoramic videos. A scene in a video is often encoded as a polygon mesh (e.g., a mesh of triangles), or as a *point cloud* (i.e., a set of 3D point coordinates and attributes such as color). Streaming raw point clouds at 30 frames per second (fps) can consume more than 4.8 Gbps of bandwidth per session [30]. Interacting with a scene (via translation and rotation) may require fetching and rendering additional point clouds, which increases the perceived delay.

Recent work has proposed several optimizations to the delivery of volumetric videos to users. For example, Vivo [9] is a client-side solution that reduces the volume of volumetric video data requested by exploiting three visibility-aware optimizations according to viewport, occlusion, and distance. Another recent work is YuZu [35], which employs the concept of super resolution for volumetric video streaming. Despite this recent research activity, a significant gap still exists in *our understanding of the quality of experience (QoE) of volumetric video streaming over a variety of Wi-Fi network configurations.*

In this paper, we conduct the first in-depth measurement study that characterizes the QoE of volumetric video streaming over a variety of Wi-Fi network configurations. The type of Wi-Fi deployment, coverage, and population density can vary significantly, and we believe that understanding performance with different Wi-Fi configuration parameter values is extremely important. The objectives of our study include: (i) understanding the impact of a number of Wi-Fi configuration parameters on user QoE, (ii) analyzing the relation between quality of service (QoS) metrics of Wi-Fi networks and application QoE, and (iii) comparing the QoE of volumetric video streaming applications to baselines that include traditional 2D video streaming and conferencing applications. Our results determine the Wi-Fi configuration parameters that have the highest impact on application QoE. We believe that our work will aid designers of next-generation Wi-Fi systems in optimizing immersive video streaming application performance.

We encountered a number of challenges during our study. First, we needed to construct a testbed that represents realistic configurations, so that our results are meaningful. Second, volumetric video streaming offers 6DoF interaction and Wi-Fi networks have several configuration parameters. Therefore, there can be a combinatorial explosion of inputs with which to experiment. Third, identifying the root causes of our observations has proven to be a complex task. This is because the relationship between the QoE of applications and the QoS metrics of Wi-Fi access points is not straightforward.

We make the following key observations from our results:

(1) Increasing the number of spatial streams and channel width improves the QoE of our volumetric video streaming applications.
(2) The air time and the duration of contention-free channel access have a significant impact on the QoE of volumetric video streaming applications.
(3) Fine-grained control over data scheduling enhances the QoE of volumetric video streaming applications.
(4) Certain Wi-Fi QoS metrics, such as the available transmission buffer and number of restramsmitted frames, are highly correlated with the QoE of volumetric video streaming applications.
(5) The QoE of adaptive 2D multimedia applications is less sensitive to Wi-Fi configuration parameters than the QoE of non-adaptive volumetric video streaming applications.

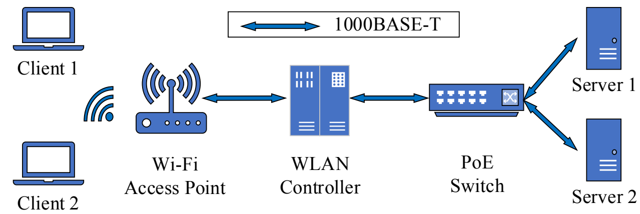

Figure 1: Experimental testbed

2 MEASUREMENT METHODOLOGY

We conduct our experiments on an enterprise-grade Wi-Fi testbed, hosting streaming clients and servers that are built by extending several open-source software tools.

2.1 Testbed

We set up a controller-based Wi-Fi testbed that mimics a small campus deployment. Our testbed consists of an Aruba 555 access point (AP) [11] which connects users to an egress network via an Aruba 7010 Mobility Controller [12], as shown in Figure 1.

We use two Acer Aspire A515-55 laptops as end-user devices. Each laptop is equipped with a Wi-Fi 6E Intel AX210 wireless card. The laptops maintain a clear line of sight to the AP. We restrict Internet access on these laptops to avoid interference from other traffic. We also deploy two servers, HPE ProLiant DL360 Gen9 (Server 1) and HPE ProLiant DL60 Gen9 (Server 2), to send videos and background traffic to Clients 1 and 2, respectively. All hosts run Ubuntu 20.04 with kernel version 5.15.0-71-generic or later.

2.2 Wi-Fi Control Parameters

Commercial Wi-Fi systems offer a variety of configuration options to meet user requirements, especially in an enterprise setting spanning a large area with a large numbers of users and applications. For example, network engineers may assign certain frequency bands to certain APs to mitigate interference. PHY and MAC enhancements in the Wi-Fi specifications also include control parameters to support real-time applications [2]. We vary the following Wi-Fi parameters in our experiments (Figure 2):

(1) *Channel Width.* Wi-Fi spectrum around 2.4 GHz and 5 GHz is divided into multiple channels. The width of each channel impacts the maximum data rate that can be transmitted. We experiment with channel widths of 20, 40, 80, and 160 MHz.
(2) *PHY/MAC Mode.* The IEEE 802.11 standards [14] define PHY/MAC modes, or *radio interfaces*, such as the 802.11n High Throughput (HT), 802.11ac Very High Throughput (VHT), and 802.11ax High Efficiency (HE). These technologies define the number of sub-carriers, type of modulation, supported channel widths, sub-carrier spacing, symbol duration, number of spatial streams, and guard intervals.
(3) *Access Category.* The 802.11e standards [14] define a QoS mechanism that assigns each traffic flow a priority level. Each priority is referred to as an access category (AC). The ACs specify the minimum and maximum contention window values as well as the length of the Transmit Opportunity (TXOP) periods. We experiment with the four available ACs: Background (AC_BK), Best Effort (AC_BE), Video (AC_VI), and Voice (AC_VO).

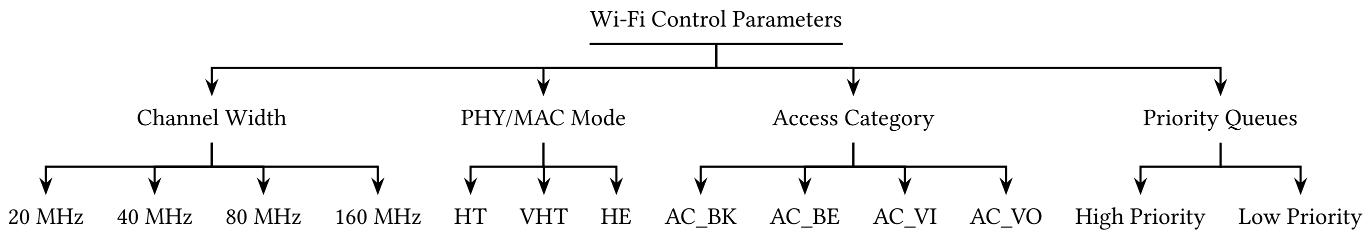

Figure 2: Wi-Fi control parameters.

(4) *Priority Queues.* Some Wi-Fi chip manufacturers enable custom prioritization of traffic flows. For example, the Qualcomm chips deployed in our APs support a feature named the *drain ratio*, which controls the ratio of the packet dequeuing rates from the different hardware queues.

2.3 Applications

Our experiments stream volumetric videos, represented as point clouds or as volumetric 3D textures. In addition, we deploy two 2D video applications, an adaptive bitrate video streaming application and a video conferencing application, to serve as baselines for comparison. The applications we consider in our study are:

(1) *Still Point Clouds (sPC).* This application renders the single-frame soldier point cloud dataset [18] at a single spatial resolution using a cube of 4096 × 4096 × 4096 voxels with a depth of 12 (Figure 3). Since the user interacts with a single frame, the maximum required bandwidth is 14 Mbps.

(2) *Moving Point Clouds (mPC).* This application renders *sequences* of point clouds from the soldier dataset [4], where each sequence is a cube of 1024 × 1024 × 1024 voxels and a depth of 10. The volumetric video is rendered at 30 fps. Thus, the maximum required bandwidth of this application is 400 Mbps. In both the sPC and mPC applications, we set the point cloud budget to 50K.

(3) *Volumetric 3D Textures (v3T).* We employ WebGL-based volume rendering [32] to display volumetric videos. Specifically, we execute a web application that uses glTF [16] to stream volumetric videos [4] using textures in the WebP [6] image format. The application renders each scene using vertex and fragment shaders executed in the rendering pipeline of the GPU (Figure 4). The maximum bandwidth required by this application is 60 Mbps.

(4) *Adaptive Bitrate Video (ABR).* We leverage the dynamic adaptive streaming over HTTP (DASH) technology. Specifically, we deploy an HTTP server that transmits a 180-second video using four-second video chunks to a client over HTTP. The video (big buck bunny [24]) is encoded at 24 fps using the H.264 and AAC codecs. We employ the buffer-based rate adaptation algorithm [13] at the client to decide the quality of each chunk. In this application, the bandwidth varies from 3.4 Mbps to 47 Mbps based on network conditions.

(5) *Video Conferencing (vCon).* We use the WebRTC [7] framework to emulate a video conferencing application between two users. The two users send a pre-recorded video [34] of a "talking head" containing audio and video streams for a duration of 180 seconds at a variable frame rate using the H.264 and Opus codecs. We also configure a TURN server to facilitate the transmission of media packets. The maximum bandwidth of the vCon application is 9.5 Mbps.

Figure 3: sPC and mPC rendering using Potree.

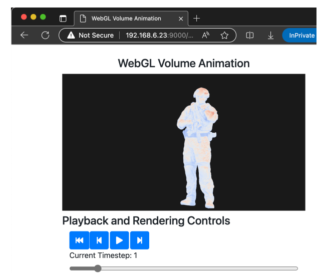

Figure 4: v3T rendering.

We selected applications (1)–(3) as example open-source immersive video streaming applications because they are easy to deploy and use on commodity GPU-enabled handheld devices, such as cellphones or tablets, using modern web browsers that support WebGL [17] and glTF [16]. Although these applications are not adaptive or optimized [9, 19, 35], our goal is to understand the impact of Wi-Fi control parameters, and these applications make it straightforward to isolate the impact of Wi-Fi parameters.

To automate our sPC and mPC application experiments, we use Potree [28], a WebGL-based large point cloud renderer that leverages octrees to efficiently load point cloud datasets. By default, the Potree client stores all downloaded point clouds in the browser cache. We found that this operation uses > 60% of memory and GPU resources on the laptop clients, and reduces application performance. To improve performance, we modify the Potree javascript web client to disable this caching functionality such that the web browser does not store any previously downloaded point clouds in memory and always fetches the requested point clouds from the server.

During each sPC and mPC application session of duration 180 seconds, the user interacts with point cloud objects every second (i.e., 180 times) using different 6DoF motions. By default, the Potree client makes multiple HTTP requests to the server every second to fetch all the frames and nodes corresponding to that interaction. We observed delay in rendering the point clouds due to the setting up and tearing down of separate TCP connections. Therefore, we captured each interaction trace and analyzed the requests made to the server, and developed tools that send the exact same requests to the server to simulate the point cloud interaction and streaming without rendering, where multiple HTTP requests corresponding to different frames are combined in a single TCP connection using the `byteranges` HTTP header.

The primary difference between the sPC and mPC applications is that mPC fetches frames from the server at a specified number of fps. The default Potree web client does not support any motion-based point cloud rendering. We therefore extended the Potree web client to add loading and unloading of point cloud frames at a given fps. The entire viewing scene is kept constant and only the moving portion of the point cloud is updated at 30 fps.

As with the sPC and mPC applications, we modified the v3T application to disable rendering. We mimic the network activity of an existing open-source browser-based client [32], by requesting a set of WebP images corresponding to the frames that are required at a particular time instance. Therefore, all our experiments focus on the impact of the network, assuming the availability of ample rendering capabilities.

2.4 QoE Metrics

Although subjective metrics are the ultimate indicators of user experience, they are often difficult to obtain, especially for a large number of system configurations, as in our study. Objective quality metrics provide a reasonable approximation of perceived quality since they are based on features such as latency, reconstruction quality, and structural similarity. Thus, We compute the objective QoE metrics for our applications as follows:

(1) *Volumetric Video Streaming.* YuZu [35] proposed a QoE model for point-cloud-based volumetric videos as a function of the point density, viewing distance, stall time, and viewport. We keep the viewing distance constant and fix the point density to 50K in our experiments. This makes the quality of each viewport constant and hence the variation between frames and patches becomes zero in equation (8) in [35]. Therefore, the user-perceived quality of each point cloud interaction in our case becomes a linear function of stall time. We use the time-to-load as a QoE metric for the sPC, mPC, and v3T applications, which is calculated as the time between when a client interacts with a scene and when it receives the last byte of the response.

(2) *Adaptive Bitrate Video.* We utilize a popular QoE metric for the ABR application, which considers the video quality of a chunk, variation in chunk quality with respect to the previous chunk, rebuffering time, and startup delay [33]. QoE is computed for each video chunk and normalized against the best possible QoE value. We average the normalized QoE for all chunks to obtain the QoE for a given session.

(3) *Video Conferencing.* We capture the videos of a WebRTC session in WebM format at the sender and receiver. Since some frames may arrive out-of-order at the receiver, we add a sequence number to each frame at the sender to represent its expected rendering order at the receiver. We use the extracted frame numbers to calculate the following reference-based quality metrics: VMAF [22], PSNR, and SSIM [31]. We use SSIM in the remainder of this paper, since it has been shown to be highly correlated to subjective mean opinion scores in scenarios with packet loss and jitter [5].

3 MEASUREMENT RESULTS

The primary goal of our experiments is to characterize multimedia application performance in Wi-Fi networks under different configurations. In the default configuration of our deployment, we set the Wi-Fi frequency band to 5 GHz, channel width to 40 MHz, access category to *Best Effort*, and PHY/MAC mode to *High Efficiency* (HE).

During each application session, we record several Wi-Fi QoS metrics on the controller every four seconds. Application QoE metrics are computed as described in Section 2.4 for each 4-second interval. These objective QoE metrics are normalized against the highest possible QoE value for each specific application. We only compare the relative values of QoE metrics across experiments with different Wi-Fi control parameter values. The QoS and QoE metrics are mapped according to their respective timestamps.

Our experimental results constitute a dataset of more than 18,000 records, where each record includes the values of 19 Wi-Fi QoS metrics and one QoE metric corresponding to a 4-second time interval. Table 1 shows an example dataset for a single experiment. We execute five iterations of each experiment for each of the five applications and each combination of Wi-Fi control parameters that we are evaluating. We show the 95% confidence intervals for our results.

Table 1: Sample dataset of a single video streaming session.

Time Interval	QoS Metric 1	QoS Metric 2	...	QoS Metric 19	QoE
0–4	363	12	...	23812	0.2037
4–8	444	26	...	24250	0.2579
8–12	416	31	...	23706	0.3981
...	...	...	...	...	...
176–180	883	4	...	23346	1.000

3.1 Impact of Wi-Fi Control Parameters

To observe the impact of a particular Wi-Fi control parameter on application QoE, we vary a single control parameter while keeping all other parameters fixed at the default configuration in each set of experiments.

3.1.1 Impact of Channel Width and Radio Interfaces. Figures 5 and 6 plot the QoE of the five applications for different Wi-Fi channel widths and physical radio interfaces. The figures show the difference between traditional and volumetric video streaming applications. The QoE for adaptive video streaming and conferencing applications remains largely unaffected by changes in the channel width or physical interface. In contrast, the QoE for volumetric video streaming applications (sPC, mPC, v3T) mostly show a gradual increase as the channel width increases or physical interface changes from HT to VHT to HE. A channel of width of 20 MHz or 40 MHz appears to be insufficient for streaming immersive multimedia content, and both HT and VHT do not perform well for the moving point cloud streaming application.

This behavior is explained by the way our applications request data from the server. The adaptive 2D video streaming application fetches a stored video chunk when the buffer space allows. Similarly, the 2D video conferencing application streams video at a

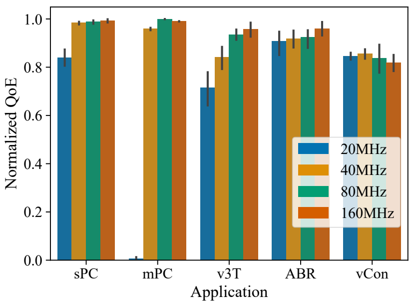

Figure 5: QoE for different channel widths.

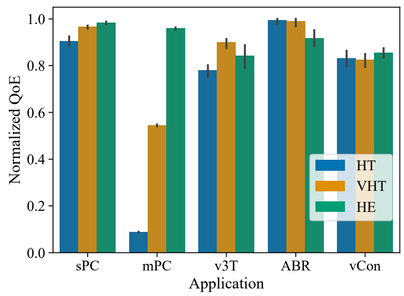

Figure 6: QoE for different PHY/MAC modes.

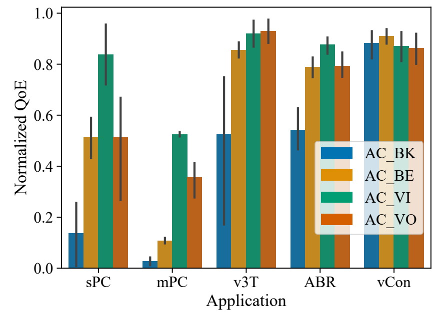

Figure 7: QoE for different access categories.

variable resolution. In contrast, our non-adaptive volumetric video streaming applications fetch large amounts of data over TCP based on user actions. User action events in immersive video streaming applications occur frequently and are not easy to predict. The time between a user action and receiving the response is highly affected by the underlying number of spatial streams and channel width. Therefore, changing the mode from HT to VHT to HE or increasing the channel width improves the QoE of immersive applications.

The two Wi-Fi control parameters discussed here, channel width and PHY/MAC mode, depend on the type of hardware in the APs. Wi-Fi APs are the type of devices that residential or even commercial customers may not upgrade very frequently. If an AP deployment does not support the latest Wi-Fi standards, users will experience lower quality while streaming volumetric video content. Therefore, it is essential for HMD and immersive application vendors to clearly specify the recommended technical specifications from the Wi-Fi deployment perspective.

3.1.2 Impact of Access Category. IEEE 802.11e defines WMM (Wi-Fi Multimedia) certification to provide QoS for devices in Wi-Fi networks. WMM creates four buckets, AC_VO, AC_VI, AC_BE, AC_BK that represent Voice, Video, Best effort and background services, respectively. Wi-Fi APs classify incoming application traffic into each bucket, and the scheduler schedules packets from these buckets at different rates. This is accomplished using the TXOP feature at the MAC layer, which provides contention-free channel access to the selected bucket for a period of time.

We conduct a preliminary experiment with Zoom, Skype, and browser-based volumetric streaming services, and find that Wi-Fi APs do not accurately classify these applications into their respective access categories. This is because the classification requires vendor-side changes in the flow classification module which is difficult to update manually for each new application. Thus, we explore which access category works best for each multimedia application.

We generate TCP background traffic using *iperf3* to observe the gain in QoE. The background traffic is streamed between Client 2 and Server 2 in Figure 1, and is scheduled over the *Best Effort* access category. Each multimedia application is then scheduled over each of the four access categories, one at a time, and the QoE metrics are recorded. As shown in Figure 7, all applications have improved QoE when the access category changes from background, to best effort, to video. Other than video conferencing and volumetric 3D textures, all applications see degradation in QoE with the Voice access category. This degradation is likely due to the fact that AC_VO has half of the TXOP duration of the AC_VI access category. Therefore, in addition to the priority of the flow, the air time and the duration of contention-free channel access to the applications play a major role in the QoE of multimedia applications.

3.1.3 Impact of Priority Queues. Packets belonging to flows within the same access category are dequeued using a FIFO mechanism. Since the network footprint of volumetric video streaming applications is different from that of traditional 2D multimedia applications, prioritizing immersive applications over standard multimedia applications that are within the same access category gives fine-grained control for scheduling, which can ultimately improve the QoE.

We validate this hypothesis by utilizing the Qualcomm priority queue feature using root access on the *ArubaOS* running on the AP. We generate background traffic as described in Section 3.1.2. The multimedia application is scheduled via the same access category as that of background traffic, which is AC_BE. The drain ratio of the priority queue is configured to four. This results in dequeuing four packets from the application queue when only one packet is dequeued from the background traffic queue. We find that this approach improves the overall throughput, latency, and QoE for the multimedia applications. As shown in Figure 8, mPC and v3T, which require the highest bandwidth, see the highest gain in QoE, compared to the scenario when priority queues are not enabled.

We note that cellular networks also employ wireless spectrum access mechanisms. Real-time multimedia application users who are located in the same radio access network (RAN) region may experience application quality degradation [26]. We posit that scheduling these application flows using priority queues implemented at a cell tower (eNodeB or gNodeB) can improve the QoE for wide-area wireless network users, much like we have observed in the case of Wi-Fi networks. We plan to explore this idea in our future work.

3.2 Wi-Fi QoS Metrics and QoE

We now use the datasets of QoS to QoE mapping, such as those given in Table 1, to analyze the relationship between instantaneous QoS metrics and the corresponding application QoE during an interval of time. We compute the Pearson correlation coefficient between each available Wi-Fi QoS metric and the normalized QoE value. For this analysis, we only vary the channel width and we only

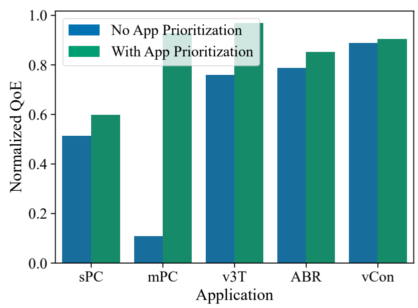

Figure 8: QoE with and without priority queues.

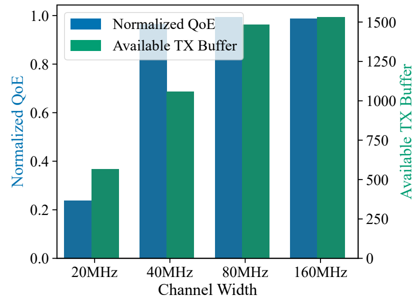

Figure 9: Available TX buffer and QoE of mPC for different channel widths.

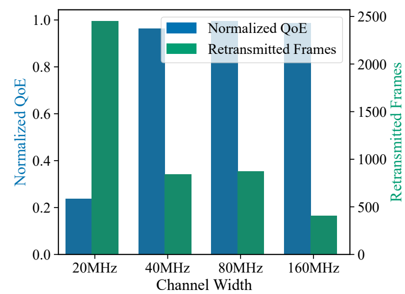

Figure 10: Retransmitted frames and QoE of mPC for different channel widths.

show the results for the mPC application since Figures 5 to 7 show that the mPC application experiences the highest variance in QoE when the values of Wi-Fi control parameters are changed. We find that the Wi-Fi QoS metrics available TX buffer, `tx_avail_buffer`, and number of retransmitted frames, `tx_retry_frames`, are the two metrics that have the highest correlation with the application QoE.

Figure 9 shows that the available TX buffer (`tx_avail_buffer`) is directly proportional to the QoE of the moving Point Clouds (mPC) application, with a correlation of +0.84. Available TX buffer refers to the amount of memory available at the Wi-Fi access point for storing outgoing frames. If the the value of this QoS metric is consistently low, it can lead to packet loss and increased latency. As we increase the channel width, the access point can drain the buffer at a higher rate, reducing the number of frames in the TX buffer. This increases the throughput and reduces the latency, and thus increases the QoE. We observe a similar relationship between the available TX buffer and the QoE for all other applications as we vary the channel width.

We also analyze the number of retransmitted frames (`tx_retry_frames`), which represents the number of frames that were successfully transmitted after being retried due to a missing acknowledgement from the client. This number has a correlation of -0.61 with QoE. At narrower channel widths, the client transmission capabilities are restricted, resulting in either packet loss or timeouts of ACK messages. This triggers the Wi-Fi access point to retransmit frames which increases the frame retransmission counts at smaller channel widths. Figure 10 shows that as channel width increases, the number of frame retransmissions decreases, reducing the latency, which ultimately increases the QoE. We also analyze the number of missed acknowledgements, and confirm these findings by calculating its correlation with both the number of retransmitted frames and the QoE.

4 RELATED WORK

Optimizing immersive application performance has been an active research area. Research in this area includes improvements in different components such as rendering, storage, compression, transport, and network support. Several systems such as Flare [25], Rubiks [10], Pano [8], and PARSEC [3] proposed optimizations for increasing the QoE of panoramic video streaming. Similarly, Vivo [9], GROOT [19], and YuZu [35] proposed optimizations to improve the QoE of volumetric video streaming. LATTE [23] grouped multimedia application users within a PHY/MAC mode and optimized group video bitrates over 802.11ac/ax networks. Zhang et al. [36] optimized the QoE of multi-user volumetric video streaming over customized mmWave multicast in 802.11ac and 802.11ad networks. IEEE 802.11ad channel access methods for virtual reality applications were compared via analysis and simulations [29]. Chord [21] used Wi-Fi lower-layer information for throughput estimation and adjusting content quality in virtual reality applications. SPAR [27] improved the positioning of virtual objects in augmented reality applications under user mobility.

Unlike the majority of these prior systems, we do not modify the multimedia applications and we do not optimize the volumetric video streaming content delivery or rendering mechanisms. Since our goal is to characterize application performance and find the Wi-Fi configuration parameters with the highest impact on application QoE, we vary several control parameters available in today's Wi-Fi access points and quantify their relationship to volumetric video streaming QoE and Wi-Fi QoS metrics.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we measure the QoE of volumetric video streaming and traditional multimedia applications in a variety of Wi-Fi network settings. Our preliminary results indicate that volumetric video streaming applications can benefit from careful Wi-Fi configuration, such as selecting an appropriate radio access category and custom flow-level prioritization. We also find that several QoS metrics at the AP highly correlate with the QoE of volumetric video streaming applications.

We are currently extending our experimental framework to support new applications and devices. We plan to conduct user studies using both handheld devices, such as cellphones and laptops, and dedicated volumetric video streaming and virtual reality headsets. We also plan to experiment with different physical environment conditions, such as line-of-sight and distance between users and the AP, mobility, number of antenna, and transmission power. Another potential avenue for future work is to investigate whether machine learning techniques can be used to accurately predict the QoE of immersive applications, and whether control actions can then be automatically taken to improve the application QoE.

REFERENCES

[1] Omid Abari, Dinesh Bharadia, Austin Duffield, and Dina Katabi. 2016. Cutting the Cord in Virtual Reality. In *Proc. of ACM HotNets*.

[2] Emna Charfi, Lamia Chaari, and Lotfi Kamoun. 2013. PHY/MAC Enhancements and QoS Mechanisms for Very High Throughput WLANs: A Survey. *IEEE Communications Surveys & Tutorials* 15 (2013).

[3] Mallesham Dasari, Arani Bhattacharya, Santiago Vargas, Pranjal Sahu, Aruna Balasubramanian, and Samir R. Das. 2020. Streaming 360-Degree Videos Using Super-Resolution. In *IEEE INFOCOM 2020*.

[4] Eugene d'Eon, Bob Harrison, Taos Myers, and Philip A Chou. 2017. 8i voxelized full bodies-a voxelized point cloud dataset. (2017).

[5] Boni García, Francisco Gortázar, Micael Gallego, and Andrew Hines. 2020. Assessment of QoE for Video and Audio in WebRTC Applications Using Full-Reference Models. *Electronics* 9, 3 (2020).

[6] Google 2023. WebP: An image format for the Web. https://developers.google.com/speed/webp. (2023).

[7] Google 2023. WebRTC. https://webrtc.org/. (2023).

[8] Yu Guan, Chengyuan Zheng, Xinggong Zhang, Zongming Guo, and Junchen Jiang. 2019. Pano: Optimizing 360 Video Streaming with a Better Understanding of Quality Perception. In *Proceedings of SIGCOMM*.

[9] Bo Han, Yu Liu, and Feng Qian. 2020. ViVo: Visibility-Aware Mobile Volumetric Video Streaming. In *Proceedings of the MobiCom '20*.

[10] Jian He, Mubashir Adnan Qureshi, Lili Qiu, Jin Li, Feng Li, and Lei Han. 2018. Rubiks: Practical 360-Degree Streaming for Smartphones. In *Proceedings of MobiSys'18*.

[11] HPE Aruba Networking 2023. Aruba 550 Series Indoor Access Points. https://www.arubanetworks.com/products/wireless/access-points/indoor-access-points/550-series/. (2023).

[12] HPE Aruba Networking 2023. Aruba 7000 Series Mobility Controllers and Gateways. https://www.arubanetworks.com/products/wireless/gateways-and-controllers/7000-series/. (2023).

[13] Te-Yuan Huang, Ramesh Johari, Nick McKeown, Matthew Trunnell, and Mark Watson. 2014. A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. In *Proceedings of SIGCOMM '14*.

[14] IEEE Standards for Information Technology 2020. IEEE 802.11-2020. https://standards.ieee.org/ieee/8802-11/11064/802.11/7028/. (2020).

[15] IEEE Standards for WLAN MAC and PHY specifications 1997. IEEE 802.11-1997. https://standards.ieee.org/ieee/802.11/1163/. (1997).

[16] Khronos Group 2023. GL Transmission Format. https://www.khronos.org/gltf/. (2023).

[17] Khronos Group 2023. Web Graphics Library. https://www.khronos.org/webgl/. (2023).

[18] Maja Krivokuca, Philip A Chou, and Patrick Savill. 2018. 8i voxelized surface light field (8iVSLF) dataset. *ISO/IEC JTC1/SC29/WG11 MPEG, input document m42914* , (2018).

[19] Kyungjin Lee, Juheon Yi, Youngki Lee, Sunghyun Choi, and Young Min Kim. 2020. GROOT: A Real-Time Streaming System of High-Fidelity Volumetric Videos. In *Proceedings of the MobiCom '20*.

[20] Luyang Liu, Ruiguang Zhong, Wuyang Zhang, Yunxin Liu, Jiansong Zhang, Lintao Zhang, and Marco Gruteser. 2018. Cutting the Cord: Designing a High-Quality Untethered VR System with Low Latency Remote Rendering. In *Proc. of ACM MobiSys*.

[21] Xing Liu, Christina Vlachou, Feng Qian, and Kyu-Han Kim. 2019. Supporting Untethered Multi-User VR over Enterprise Wi-Fi. In *Proceedings of the 29th ACM Workshop on NOSSDAV '19 (NOSSDAV '19)*.

[22] Netflix 2023. VMAF - Video Multi-Method Assessment Fusion. https://github.com/Netflix/vmaf. (2023).

[23] Hannaneh Barahouei Pasandi, Hadi Amirpour, Tamer Nadeem, and Christian Timmerer. 2021. Learning-Driven MU-MIMO Grouping for Multi-User Multimedia Applications Over Commodity WiFi. In *Proceedings of VisNEXT'21*.

[24] Peach Open Movie Project 2023. Big Buck Bunny. https://media.xiph.org/video/derf//y4m/big_buck_bunny_1080p24.y4m.xz. (2023).

[25] Feng Qian, Bo Han, Qingyang Xiao, and Vijay Gopalakrishnan. 2018. Flare: Practical Viewport-Adaptive 360-Degree Video Streaming for Mobile Devices. In *Proceedings of MOBICOM*.

[26] Eman Ramadan, Arvind Narayanan, Udhaya Kumar Dayalan, Rostand A. K. Fezeu, Feng Qian, and Zhi-Li Zhang. 2021. Case for 5G-Aware Video Streaming Applications. In *Proceedings of 5G-MeMU '21*. Association for Computing Machinery.

[27] Xukan Ran, Carter Slocum, Yi-Zhen Tsai, Kittipat Apicharttrisorn, Maria Gorlatova, and Jiasi Chen. 2020. Multi-User Augmented Reality with Communication Efficient and Spatially Consistent Virtual Objects. In *Proceedings of the CoNEXT '20*. 386–398.

[28] Markus Schutz. 2016. *Potree: Rendering Large Point Clouds in Web Browsers*. Master's thesis. Institute of Computer Graphics and Algorithms, Vienna University of Technology.

[29] Jakob Struye, Filip Lemic, and Jeroen Famaey. 2020. Towards Ultra-Low-Latency mmWave Wi-Fi for Multi-User Interactive Virtual Reality. In *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*.

[30] Jeroen van der Hooft, Hadi Amirpour, Maria Torres Vega, Yago Sanchez, Raimund Schatz, Thomas Schierl, and Christian Timmerer. 2023. A Tutorial on Immersive Video Delivery: From Omnidirectional Video to Holography. *IEEE Communications Surveys & Tutorials* 25, 2 (2023).

[31] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[32] Will Usher 2023. Volume Rendering with WebGL. https://www.willusher.io/webgl/2019/01/13/volume-rendering-with-webgl. (2023).

[33] Xiaoqi Yin, Abhishek Jindal, Vyas Sekar, and Bruno Sinopoli. 2015. A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. *SIGCOMM Comput. Commun. Rev.* 45, 4 (aug 2015).

[34] YouTube 2023. Talking Head. https://www.youtube.com/watch?v=hWTT4J_xNwY. (2023).

[35] Anlan Zhang, Chendong Wang, Bo Han, and Feng Qian. 2022. YuZu: Neural-Enhanced Volumetric Video Streaming. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*.

[36] Ding Zhang, Bo Han, Parth Pathak, and Haoliang Wang. 2021. Innovating Multi-User Volumetric Video Streaming through Cross-Layer Design. In *Proceedings of the 20th ACM Workshop on HotNets'21*.