Enabling Mobile Virtual Reality with Open 5G, Fog Computing and Reinforcement Learning

Yaohua Sun, Jianmin Chen, Zeyu Wang, Mugen Peng, and Shiwen Mao

Abstract

Virtual reality (VR) is an emerging technology reshaping interactive experience and can be widely applied in gaming, operation training, and so on. Despite its great potentials, most existing VR solutions suffer from low mobility support, high latency, quick battery drain, as well as high cost of user devices. To overcome these issues, a mobile VR system is designed and implemented. The basic idea is to take advantage of 5G and fog computing to realize high bandwidth and low latency VR service. Meanwhile, our design features the integration of an open 5G base station (BS) and an intelligent controller. With the help of artificial intelligence (AI) and the interfaces provided by the BS and fog VR servers, the controller can wisely adjust both the BS-level and application-level parameters to enhance system performance. Using our prototype system, the impact of various parameters, the superiority of fog computing over cloud computing in supporting VR, and the effectiveness of AI in optimizing system performance are all demonstrated. In addition, multi-dimensional resource optimization for VR delivery and strategy design for VR service migration are identified as two promising future research directions.

Introduction

As a disruptive technology that brings new interactive experience, virtual reality (VR) has attracted considerable attention from content providers, mobile network operators, vertical industry customers, and so on. According to Goldman Sachs, the VR and augmented reality (AR) ecosystem will grow to an $80 billion market by 2025 [1]. Current applications of VR include entertainment, education, and skill training, and so on.

There are three common product types of VR. The first one uses a head mounted display (HMD) connected to a personal computer (PC) or a game console in a wired manner. The second one relies on an HMD with a mobile phone attached to it, while the third one adopts an all-in-one HMD. For all the existing solutions, customers need to buy high-performance but expensive devices to enjoy excellent VR quality; meanwhile, the first solution also restricts users to a small and fixed area. To achieve light, affordable, and mobile VR, the concept of cloud VR has been proposed. The core idea is to move computation-intensive rendering operations to powerful cloud servers, and then send rendered frames to HMDs by video streaming.

In [2], a cloud VR system is presented to optimize rotation latency and interaction latency. Rotation latency refers to the time elapsed from a head movement to the view update in the HMD, while interaction latency means the time elapsed after moving an object until that movement is observed. By mainly rendering the front-facing view with high resolution and leaving the rendering of small objects to local devices, both latency can be effectively reduced. Moreover, performance measurements on a cloud VR gaming platform called Air Light VR (ALVR) are conducted in [3]. Following the trend of cloud VR, some researchers go one step further to consider VR transmission in fog computing enabled cellular networks. In [4], a Field of View (FoV) rendering scheme deployed at fog computing infrastructures is proposed for VR video delivery and the test result reveals that the traffic in the core and radio access links can be reduced by over 80 percent. In [5], the authors implement a VR solution also utilizing rendering with fog computing, where a margin around FoV is streamed back as well to achieve better adaptation to different network latency conditions.

Although the existing works [2–5] have built their own VR systems and tested their performance, the radio transmission is still based on the fourth-generation (4G) mobile communication system or WiFi. Particularly, the data rate of 4G cannot support the satisfactory experience of strongly interactive VR whose rate requirement can reach over 260 Mb/s per user as indicated in [6], while WiFi suffers low communication range and potentially severe interference due to the use of unlicensed band. In addition, to the best of our knowledge, the potential of artificial intelligence (AI) in optimizing VR performance on a real-world testbed has not been investigated before, which, however, is essential when considering efficient resource utilization, the dynamic radio environment, and the co-existence of VR and other services. To achieve a superior VR experience, *this article designs and implements an (possibly the first) open 5G and AI empowered mobile VR system.* Owing to the interfaces offered by the open 5G base station (BS) and the fog VR server,

Digital Object Identifier:
10.1109/MNET.010.2100481

Yaohua Sun (corresponding author), Jianmin Chen, Zeyu Wang, and Mugen Peng are with Beijing University of Posts and Telecommunications, China; Shiwen Mao is with Auburn University, USA.

both network and application parameters can be flexibly adjusted by an intelligent controller. Particularly, an off-line deep reinforcement learning (DRL) based multi-level parameter optimization approach is proposed for the intelligent controller. With the prototype system, the impacts of various parameters on the VR performance and the efficacy of the DRL based approach are evaluated and verified.

The remainder of this article is organized as follows. The recent trends in mobile VR are briefly introduced in the next section. Following that, our proposed mobile VR system and its implementation are presented. The impacts of BS-level and application-level parameter settings on the VR performance are tested and the performance improvement brought by rendering with fog computing is numerically demonstrated. After that, an offline DRL approach is proposed for system performance optimization. We then discuss two open issues and conclude this article.

State-of-the-Art of Mobile VR

Separating Rendering from Local Devices

Previously, the implementation of VR commonly adopts local computing based solutions that rely on PCs, gaming consoles, or smart phones as external rendering devices or just uses standalone HMDs. These solutions have the following issues:

- High cost devices: For highly interactive VR services, local computing devices have to execute computation-intensive graphic rendering, which requires high-performance processors. This unavoidably raises the threshold for enjoying VR experience.
- Poor user experience: To support high VR quality, standalone VR HMDs can be heavy and easy to get hot, which make users feel uncomfortable.
- High power consumption: Standalone HMDs and mobile phones have to be frequently recharged due to the significant power consumption incurred by graphics rendering.
- Limited mobility: When HMDs are cable connected to PCs or game consoles, user's mobility is greatly constrained.

To support mobile and high performance VR applications with substantially lower equipment cost, *a recent trend is to move heavy computing tasks such as graphics rendering to cloud or edge servers and then send VR contents via video streaming to HMDs over wireless networks*, while the HMDs will focus on the functions such as sensory data uploading, video frame decoding, head motion rendering, and image displaying. In this way, HMDs can be lighter, much cheaper, and with lower energy consumption. More importantly, users can fully enjoy portable and mobile VR experience, and application developers do not have to struggle with heterogeneous HMD operation systems anymore.

Trends from the Perspective of Communication and Computing

With the separation of heavy rendering tasks from local devices to cloud or edge servers, communication and computing both play key roles in ensuring the quality of experience (QoE) of VR users.

To support mobile and high performance VR applications with substantially lower equipment cost, a recent trend is to move heavy computing tasks such as graphics rendering to cloud or edge servers and then send VR contents via video streaming to HMDs over wireless networks, while the HMDs will focus on the main functions such as sensory data uploading, video frame decoding, head motion rendering, and image displaying.

The Communication Perspective: Since non-local servers need to transmit the rendered VR contents to HMDs within a very short time, the wireless links should be able to support a high peak data rate. For example, according to the test results of China Mobile on a cloud VR game, the wireless network needs to transmit 0.5–1 Mbits of data within several milliseconds [7], hence leading to a peak data rate of several hundred Mb/s. Facing this challenge, researchers propose to use millimeter wave (mmWave) [8] and even Terahertz (THz) communications for wireless VR transmission.

The Computing Perspective: First, fog computing infrastructures, such as fog access points [9] or dedicated fog servers, can be exploited for graphics rendering, which are usually much closer to users compared to cloud servers, thus shortening the end-to-end transmission latency. Second, at fog VR servers, strategic rendering can be exploited to reduce the rendering latency. For example, graphics can be pre-rendered based on the prediction of user's head rotation and servers can also choose to render only the FoV image. Third, a prerequisite for offloading rendering tasks to a server is that the server has pre-cached the necessary application codes and original VR contents. This is termed service caching and various caching policies can be adopted by leveraging the statistics of VR application requests.

An Intelligent Mobile VR Gaming System Based on Open 5G and Fog Computing

Motivated by the trends described earlier, we have designed and implemented a mobile VR system based on 5G and fog computing, whose components will be elaborated in this section.

System Overview

Different from the existing systems, the novelty of our design, as shown in Fig. 1, lies in the integration of an open 5G BS and an intelligent controller. In recent years, mobile network operators have paid much attention to the openness of radio access networks (RANs), which features softwarized RAN functions running on commodity hardware with various open interfaces available to external controllers, and it is envisioned that open RANs can help reduce the capital expenditure and enable more network flexibility as well as intelligence. The open 5G BS in our system directly connects a user plane function (UPF) close to it, which is responsible for processing GTP packets and re-direct VR traffic to a fog VR server. The server is capable of realizing elastic VR service provisioning, for example, based on the docker virtualization technique. Via the interfaces to the open 5G BS and the fog server, the intelligent controller can collect rich data related to radio transmission and VR application, with

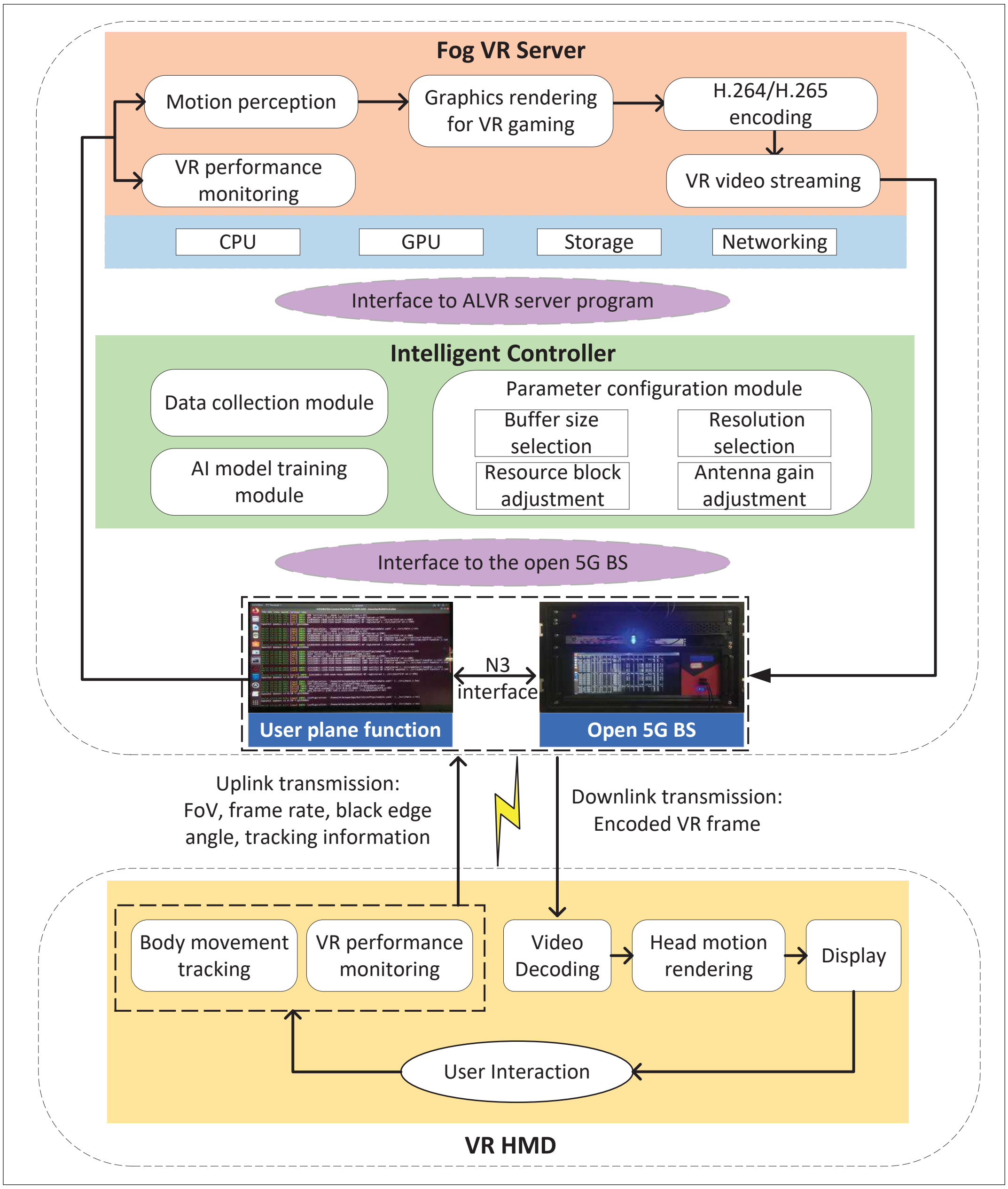

FIGURE 1. The architecture of the proposed mobile VR gaming system.

which AI models are trained and utilized to intelligently configure the BS-level and application-level parameters to enhance system performance. In the remainder of this section, the details of each component in Fig. 1 will be introduced.

The Open 5G BS

In the system implementation, the 5G SA protocol stack software from Amari is installed on a general purpose server equipped with an Intel CPU (9900k) together with 16 GB memory and a Linux operating system. The software provides rich interfaces that allow flexible configuration on radio parameters. Moreover, a hardware acceleration card based on a field-programmable gate array (FPGA) is integrated to speed up baseband processing. The 5G radio signal is transmitted/received by a four-antenna radio unit supporting 100 MHz bandwidth that connects to the BS via a 40G fiber, and the UPF communicates with the BS as well as other control plane functions of the core network coming from Open5GS.

The Fog VR Server

To perform dynamic and low latency rendering, the fog VR server is constructed with a personal computer with a gtx1080 graphics card. Meanwhile, the server consists of Steam VR, the ALVR server program, and pre-downloaded VR game applications to reduce the service loading time. ALVR provides an open-source solution to VR streaming, which invokes the OpenVR API to register VR HMD's information in SteamVR, and ALVR also extracts rendered VR frames from SteamVR. The frames are further encoded by the ALVR server program using H.264 or H.265 and streamed to VR HMDs based on UDP transmission. Moreover, various VR performance metrics can be fetched from the ALVR program, such as VR frame transmission latency, frame rate at

HMDs, and packet loss rate. The performance metrics that are unavailable, such as black edge angle and the uplink transmission latency of user action information, can be acquired by manipulating the ALVR program. In addition, unlike the traditional 360° video service, our fog VR server is designed for interactive VR gaming that requires real-time rendering based on the tracking information of user movements, and hence it would be difficult to use multi-casting in the multi-user case due to individualized viewpoint of each user. However, there are still other ways of enhancing transmission efficiency, such as using an advanced encoding/compression technique and advanced transport layer protocol like QUIC.

The VR HMD

In our system, we use Oculus quest as the VR HMD and its screen refresh rate can be up to 90 Hz. With Qualcomm Snapdragon XR inside, the HMD with ALVR client installed is capable of decoding VR frames received from the fog VR server with a high performance. After decoding, the HMD has to further deal with head motion rendering to fit the image into current user's orientation [10]. In addition to establishing the connection with the ALVR server program, the ALVR client program also allows users to appoint the IP address of a target VR server by some code manipulation. Other functions of the ALVR client program include HMD/user movement state acquisition and uploading VR frame rate as well as black edge angle. Furthermore, since the HMD is not equipped with a 5G communication chip, it accesses the 5G BS with the help of a 5G CPE that translates 5G signal to WiFi signal.

The Intelligent Controller

The controller is responsible for configuring VR application parameters and BS parameters to optimize a certain objective. Given the system complexity, it is endowed with decision-making capability by taking advantage of AI. Specifically, the controller consists of a data collection module, an AI model training module, a parameter configuration module, and interfaces to the open 5G BS and the ALVR server program. The interfaces are implemented based on Web socket communication and the data going through the interfaces is in the JavaScript Object Notation (JSON) format. Via these interfaces, the data collection module can gather current parameter configurations and various performance metrics, including the number of available resource blocks (RBs), the selection of modulation and coding scheme, VR frame transmission latency, packet loss rate, and so on. By feeding such rich data into the AI model training module, different AI models can be created, such as models for intelligent system parameter configuration and models for the prediction of VR user experience.

VR Performance Evaluation

In this section, with the wireless VR prototype system, the impacts of several key parameters on the VR performance are evaluated and the advantages of rendering by fog computing is also demonstrated.

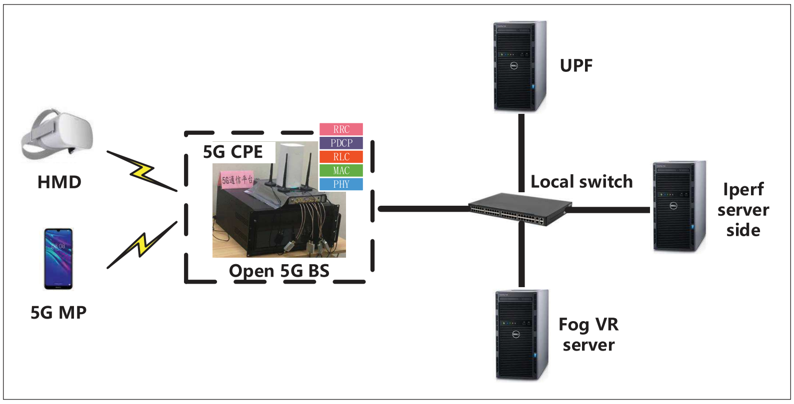

FIGURE 2. The network topology for performance evaluation. Note that AMF and SMF from Open5GS as well as the intelligent controller are also connected to the switch, which are omitted in the figure.

Test Scenario, Performance Metrics and System Parameters

Given the co-existence of extensive differentiated services in future wireless networks, we consider the test scenario in Fig. 2, where there is an Oculus HMD accessing the open 5G BS via a 5G CPE and there is also a commercial 5G mobile phone (MP) associated with the 5G BS directly. The MP is inserted with a dedicated subscriber identification module (SIM) card that records information for authentication at Access & Mobility Management Function (AMF). To simulate a high data rate service, we continuously send UDP packets to the MP from a PC using an `iperf` command and the sending rate is 1Gb/s.

In the subsequent tests, the following performance metrics are considered.

Downlink Data Rate: It measures the data transmission speed in the downlink. Data rate information can be collected from the open 5G BS on a per-user basis.

Packet Loss Rate: It is calculated every second and is defined as the number of frame data packets not correctly received by the HMD divided by the total number of frame data packets sent by the fog VR server. This metric is automatically calculated by the ALVR server program. A high packet loss rate will lead to strong mosaic effect on the HMD screen, which degrades the view clarity.

The Angle of Black Edge: In interactive VR applications, usually users frequently rotates their heads. During the rotation, black domains or smears can appear at the edge of the FoV, which are called black edge. *Clearly, the angle of black edge directly affects the immersing experience, which, however, has been seldom measured in prior work*. Following the definition given by [10], we have modified the code of the ALVR client for online calculation of the black edge angle.

Frame Rate: It refers to the actual rate at which frames are displayed at the HMD. To achieve a satisfactory QoE, frame rate should be close to 60 frames per second [5]. The value of this metric is periodically uploaded by the ALVR client to the ALVR server.

Peak Signal-to-Noise Ratio (PSNR): PSNR is a common metric to measure the similarity between two pictures. This metric will be adopted to intuitively compare the performance of fog rendering with that of cloud rendering. The details related to PSNR calculation will be introduced later.

		Data rate (Mb/s)			Featured performance metrics for VR		
		Total rate	The rate of the MP	The rate of VR HMD	Packet loss rate	Black edge angle	Frame rate
VR frame resolution: 1536 × 768, receiving buffer size: 200KB							
Number of RBs	50	159.47	121.81	37.66	47.00 %	12.65°	43.00
	75	238.35	200.23	38.12	24.10 %	10.47°	42.00
	100	332.69	293.13	39.56	23.20 %	9.31°	43.00
	125	439.97	398.75	41.22	24.60 %	7.25°	45.00
	150	626.88	583.20	43.68	25.10 %	7.01°	46.00
	175	676.91	629.02	47.89	9.33 %	5.31°	44.00
	200	744.10	692.20	51.90	0.72 %	3.96°	44.00
VR frame resolution: 1536 × 768; number of RBs: 150							
Receiving buffer size (KB)	200.00	627.48	584.55	42.93	24.41 %	7.22°	46.00
	400.00	602.56	557.95	44.61	14.25 %	6.13°	45.00
	800.00	638.09	593.06	45.03	8.20 %	8.22°	43.00
	1200.00	612.93	568.12	44.81	1.14 %	4.10°	40.00
	1600.00	624.14	581.48	42.66	0.00 %	9.04°	41.00
	2000.00	613.41	572.83	40.58	0.00 %	10.81°	40.00
The number of RBs: 150; receiving buffer size: 200KB							
VR frame resolution	1024 × 512	611.05	581.13	29.92	7.88 %	4.05°	47.00
	1536 × 768	606.22	560.42	45.80	25.19 %	7.19	43.00
	2048 × 1024	612.32	548.91	63.41	26.80 %	9.28	41.00
	2560 × 1280	602.71	528.79	73.92	25.74 %	9.59	37.00

TABLE 1. Test results.

The Impacts of Key Parameters

As for parameters to be adjusted, we mainly focus on the number of RBs available to the open 5G BS, receiving buffer size at the HMD, and the resolution of VR frames. Note that these parameters can be set by the intelligent controller via implemented interfaces mentioned before.

First, the impacts of the number of RBs are evaluated under a fixed buffer size and VR frame resolution, which are set to 200KB and 1536 × 768, respectively, while adaptive coding rate for VR streaming is adopted. From Line 5 to Line 11 of Table 1, it can be seen that the downlink date rate of the MP increases significantly with increased number of RBs. As for the VR HMD, the number of RBs mainly affects its packet loss rate that is reduced from 47 percent to nearly 0 percent, and meanwhile a larger number of RBs also contributes to the improvement of black edge angle. From Line 13 to 18, we keep the number of RBs at 150 and show the impacts of buffer size on the user performance under the resolution of 1536 × 768. It is observed that the data rate of both VR HMD and MP only slightly fluctuate under various buffer sizes. However, the buffer size has a considerable influence on the packet loss rate of VR service. At last, the impacts of VR frame resolution are demonstrated from Line 20 to 23. It is intuitive that a larger frame resolution leads to a much higher data rate of VR service. However, due to the limited radio resource, a higher resolution setting also results in a lower VR experience, namely a lower frame rate, a higher packet loss rate, and a higher black edge angle.

Performance Comparison between Fog Rendering and Cloud Rendering

We adopt PSNR to illustrate the performance improvement brought by moving graphics rendering to the fog VR server. By using PSNR, the similarity between the frame viewed by the user and the original frame at the VR server is measured quantitatively. To simulate the transmission condition for cloud rendering, we manually increase the round trip time and packet loss rate between the UPF and the fog VR server by 40ms and 3 percent, respectively. In Fig. 3, the blue and orange curves correspond to the PSNR values for different frame indexes in the fog VR case and cloud VR case, respectively, while the green and purple curves correspond to the average PSNR values achieved in the fog VR case and cloud VR case, respectively. First, with the increment of frame index, the VR user's head always rotates at an angular speed of around 60°/s, and hence both curves fluctuate due to the fast and frequent change of user viewpoint. Second, owing to the lower latency and packet loss rate in the fog VR case, the blue curve is above the orange curve for most of the frames and hence the average PSNR is significantly improved by fog VR as indicated by the gap between the green and purple curves. Meanwhile, it can be seen from Fig. 3 that the view quality when PSNR is relatively lower has been significantly enhanced as well.

System Performance Optimization Based on Reinforcement Learning

Joint Optimization of Multi-Level Parameters

Considering the scarcity of radio resource and the diverse performance requirements of the VR HMD and MPs, we propose a batch constrained offline DRL based approach to system performance optimization. The aim is to balance the RB usage and VR performance while guaranteeing the target data rate of MPs. Our approach is developed by extending an open-source offline RL algorithm called batch constrained offline deep RL. The state space, action space, and reward function are re-defined to adapt it to our problem. Interested readers are referred to [11] for more details. Here, we mainly introduce the definition of state, action, and reward as well as the underlying environment where learned policies will be deployed.

Environment: Instead of using a simulated environment, the prototype system used for performance evaluation above is taken as the underlying environment with one more MP, from which we collect enough transition data for offline DRL model training. After training is finished, the trained model is applied to the environment to optimize system parameter configuration in an online fashion.

State: In DRL, the learning agent selects an action based on currently observed state. In our experiment, state is composed of the state associated with the two MPs expressed as

$$\{dl^1_{mcs}, ul^1_{mcs}, snr^1, dl^2_{mcs}, ul^2_{mcs}, snr^2\}$$

as well as the state related to the VR HMD

$$\{dl^3_{mcs}, ul^3_{mcs}, snr^3, FrameLatency, ActionLatency\},$$

where dl_{mcs} and ul_{mcs} represent the modulation and coding schemes (MCS) adopted in the downlink and uplink, respectively, and snr is the signal-to-noise ratio (SNR) measured in the uplink by the open 5G BS. In addition to MCS and SNR information, the state of VR HMD also incorporates *FrameLatency* and *ActionLatency*. The former refers to the latency of delivering a VR frame from the fog VR server to the VR HMD, while the latter measures the latency induced by uploading the movement information of the HMD to the VR server.

Action: Earlier, the impacts of several system parameters were evaluated, including the number of RBs at the open BS, receiving buffer size at the HMD, and the VR frame resolution. Since these parameters all affect user performance, our DRL model intends to optimize them jointly. Then, each action of the learning agent represents a possible tuple of these parameters denoted by {*RBNum*, *FrameResolution*, *BufferSize*}. Specifically, $RBNum \in \{25, 50, ..., 250\}$, $FrameResolution \in \{1024 \times 512, 1536 \times 768, 2048 \times 1024\}$, and $BufferSize \in \{0.1MB, 1MB, 2MB\}$. Therefore, there are 90 different actions in total.

Reward: Reward is a signal fed back by the underlying environment, which guides RL agents to adjust their action selection policies. In this article, our goal is to improve the QoE level of the VR user while reducing the resource consumption at the open BS and satisfying the data rate requirement of the other two users. Therefore, the reward function is expressed as follows.

$$r = 1 - \left[\alpha_1 \frac{RBNum}{RBNum_{max}} + \alpha_2\left(\frac{\theta}{360°} + PLR + \frac{|f - f_{target}|}{f_{target}}\right) + \alpha_3 \sum_{i=1}^{3} \frac{|R^i - R^i_{target}|}{R^i_{target}}\right], \quad (1)$$

where $RBNum_{max}$ is the maximal number of available RBs at the open 5G BS, which is 273 under 100MHz bandwidth, θ is the black edge angle perceived by the VR HMD, PLR is the packet loss rate for end-to-end VR transmission, f and f_{target} represent the actual frame rate at the HMD and target frame rate, respectively, R^i and R^i_{target} represent the actual downlink rate and target downlink rate of MP i, while α_1, α_2 and α_3 are weight factors, indicating the importance of resource usage at the open BS, the importance of VR performance, and the importance of the QoS of users requesting high speed UDP service, respectively. According to the test results in Table 1, a lower VR frame resolution will lead to higher MP data rate under fixed number of RBs. Hence, with the above reward design, our DRL agent is encouraged to choose a relatively high VR resolution to improve user experience, since a too low resolution will lead to a potentially large

$$\alpha_3 \sum_{i=1}^{3} \frac{|R^i - R^i_{target}|}{R^i_{target}},$$

and then the total reward will be degraded. Finally, it is worth noting that other reward setting or VR

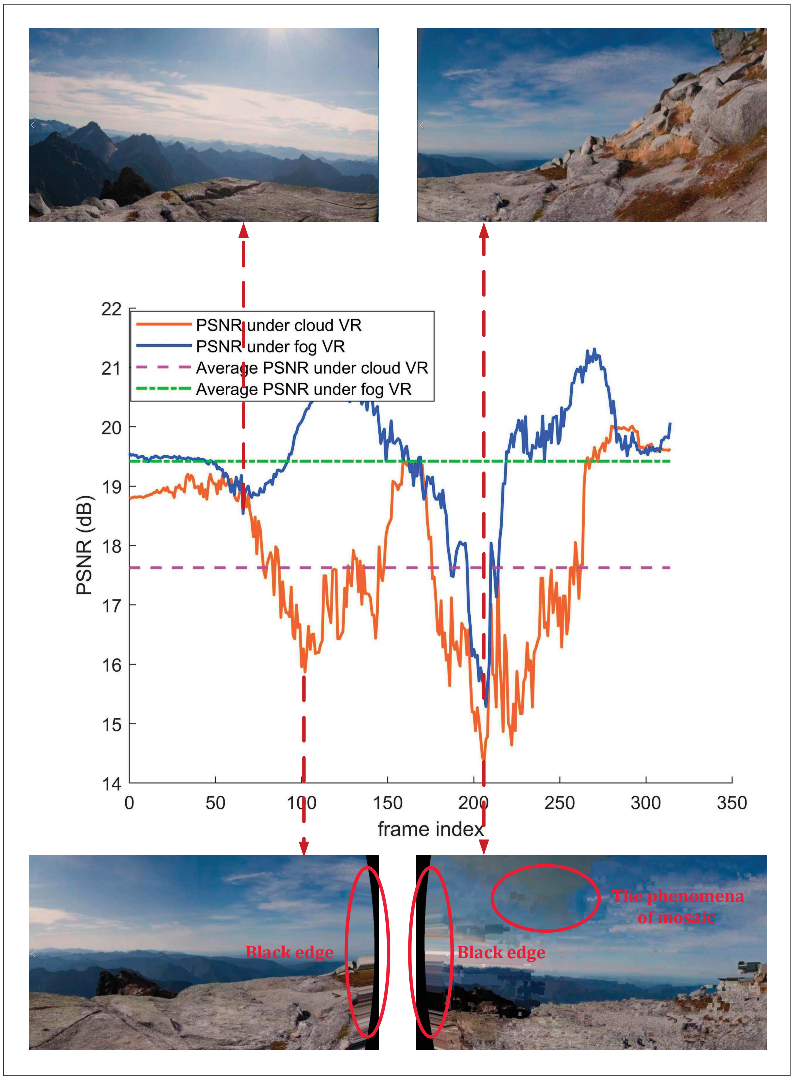

FIGURE 3. Time-varying PSNR values under fog rendering and cloud rendering: VR frame resolution: 2560 × 1280; the number of RBs: 250; receiving buffer size: 2MB; the angular velocity of user head rotation: 60°/s.

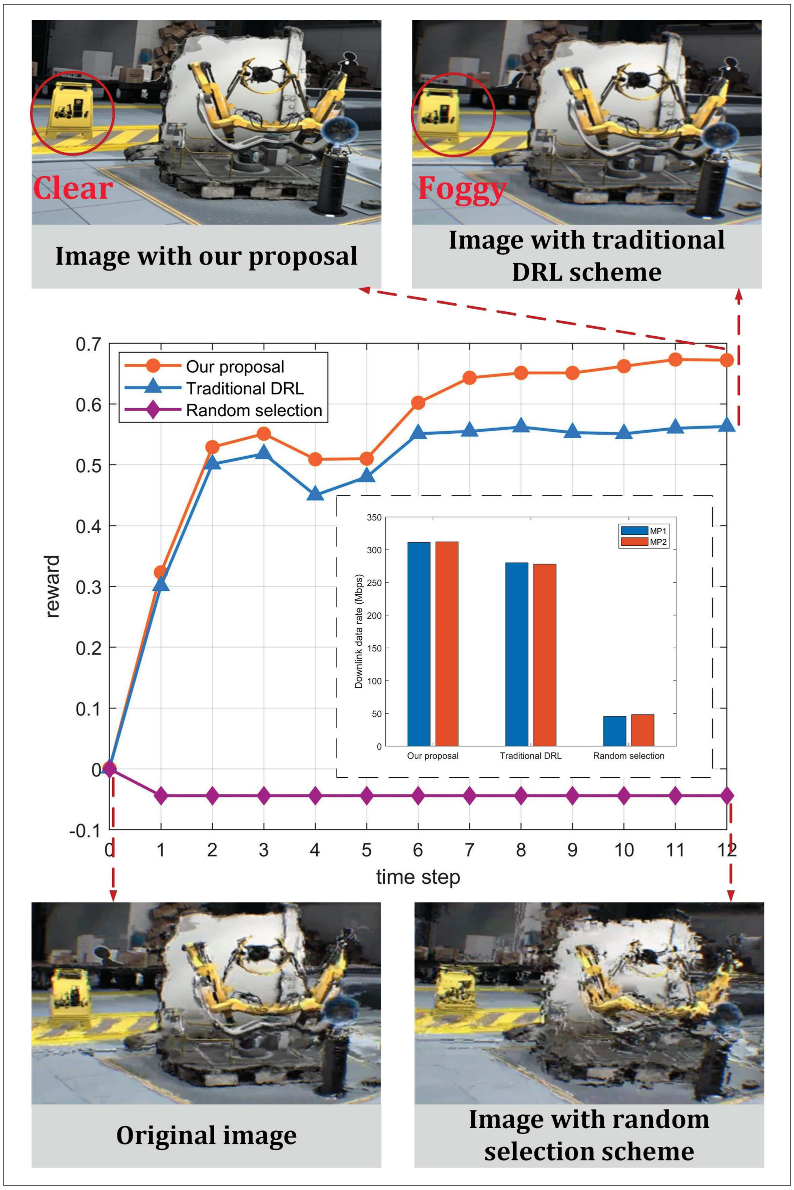

FIGURE 4. Reward and performance comparison between our proposal and the baseline schemes. (Played VR game: The Lab).

QoE models can be easily incorporated owing to the model-free nature of our proposed approach.

Evaluation Results

To show the effectiveness of the proposed system, 90,000 pieces of transition data are collected from the environment described above and used for the offline training of our DRL model, whose basic settings such as model structure and learning rate follow that in [11]. As for the parameters in the reward definition, since we care more about the QoE of the VR user than the RB usage and the QoS of the other two users, α_1, α_2, and α_3 are set to 0.3, 0.4, and 0.3, respectively. In addition, f_{target} is set to 60 FPS, while R^i_{target} is set to 320Mb/s for both MPs. At the open BS, proportional fair user scheduling policy is adopted. After training the model, the DRL model is deployed in the parameter configuration module of the intelligent controller for online system performance optimization.

In Fig. 4, the online optimization performance of our proposal, traditional DRL based parameter optimization, and random parameter selection are compared. Particularly, the traditional DRL refers to the deep RL algorithm proposed in [12], which shares the same state space, action space, and reward function as our proposal. Since our proposal is based on a modified version of the traditional DRL that better fits into offline training, the achieved reward is improved by 19 percent relative to that of the traditional DRL. Furthermore, the scenes seen by the VR user under different optimization schemes are also shown in the figure. It can be seen that our proposal leads to the best VR quality. Moreover, the data rate achieved by each MP is also the highest with our proposal. To intuitively demonstrate the necessity of conducting joint optimization of multi-level parameters, the results of optimizing only single-level parameters with our proposal are also presented in Fig. 5, from which it can be observed that joint optimization outperforms single-level optimization.

Open Issues

Resource Optimization for VR Services

To fully utilize the edge caching resources to reduce the VR latency, the authors in [13] propose a view synthesis-based VR caching scheme, which can synthesize an uncached but requested view using its adjacent views. In [14], both mmWave and sub-6 GHz links are used for VR transmission to enjoy the high bandwidth of mmWave communications while guaranteeing disruption-free transmission with sub-6GHz. Although the above works achieve good performance, the joint optimization of cache, computation, and radio resource has not been fully addressed, which is the key to further improve the VR performance. However, since cache resource is often adjusted on a larger timescale than computation and radio resource, the corresponding problem features mixed-timescales, hence being challenging to solve.

VR Service Migration Strategies

In mobile VR scenarios, users can traverse areas covered by different BSs, which incurs BS handover. When a handover event occurs for a user, the virtual machine (VM) running its requested VR application may also need to be migrated from the fog computing platform of its current BS to the platform of another BS, which is critical to ensure a stable service performance [15]. Considering the dense deployment of BSs in future networks, studying the way of identifying the appropriate target BS for handover is essential, which should not only take the wireless channel quality into account but also consider the computing resource utilization of the target BS as well as VM migration latency.

Conclusions

This article presented a wireless virtual reality (VR) service system that incorporates an intelligent controller, an open 5G base station (BS), and a fog VR server implemented with Air Light VR (ALVR). On one hand, by offloading graphics rendering from user HMDs to a fog computing platform, better user experience has been achieved compared to traditional cloud based VR. On the other hand, via interfaces to the open BS and

ALVR server program, the controller can make wise decisions on radio and application parameter configuration by utilizing reinforcement learning. Finally, towards a practical multi-VR-user scenario, due to more diversified content requests and the higher transmission rate demands, it is essential to study VR quality level selection under both storage and communication constraints to enhance the user quality-of-experience, which will be investigated in our future work.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China under Grant 62001053, 61925101, 61831002, the National Key Research and Development Program of China under Grant 2021YFB2900200, Young Elite Scientist Sponsorship Program by China Institute of Communications, S. Mao's work is supported in part by the NSF under grants CNS-2107190 and CNS-2148382.

References

[1] H. Bellini, "The Real Deal With Virtual and Augmented Reality," available: http://www.goldmansachs.com/our-thinking/pages/virtual-and-augmented-reality.html; accessed on Dec. 30, 2021.
[2] T. Kämäräinen et al., "CloudVR: Cloud Accelerated Interactive Mobile Virtual Reality," *Proc. 26th ACM Int'l. Conf. Multimedia*, Seoul, South Korea, Oct. 2018, pp. 1181–89.
[3] Y.-C. Li *et al.*, "Performance Measurements on a Cloud VR Gaming Platform," *Proc. 1st Workshop on Quality of Experience in Visual Multimedia Applications*, Seattle, WA, Oct. 2020, pp. 37–45.
[4] S. Mangiante *et al.*, "VR is On the Edge: How to Deliver 360 Videos in Mobile Networks," *Proc. Workshop on Virtual Reality and Augmented Reality Network*, Los Angeles, CA, Aug. 2017, pp. 30–35.
[5] S. Shi *et al.*, "Mobile VR on Edge Cloud: A Latency-Driven Design," *Proc. 10th ACM Multimedia Systems Conf.*, New York City, NY, June 2019, pp. 222–31.
[6] Huawei iLab, "Cloud VR network solution white paper," available: https://www.huawei.com/minisite/pdf/ilab/ cloud vr network solution white paper en.pdf; accessed on Dec. 30, 2021.
[7] China Mobile, "Quality of Experience Driven Networking for Cloud VR," available: http://www.chinavea.cn/ upload/Pf7a1f2f5bc8a4689bc77ee4b1651315e.pdf; accessed on Dec. 30, 2021.
[8] O. Abari *et al.*, "Enabling High-Quality Untethered Virtual Reality," *Proc. 14th USENIX Symposium on Networked Systems Design and Implementation*, Boston, MA, Mar. 2017, pp. 531–44.
[9] M. Peng *et al.*, "Fog Computing Based Radio Access Networks: Issues and Challenges," *IEEE Network*, vol. 30, no. 4, July 2016, pp. 46–53.
[10] Huawei iLab, "Black edge and network latency white paper," available: https://www-file.huawei.com/-/media/corporate/pdf/ilab/2019/cloud-vr-blackline-network-delay-en-v1.pdf; accessed on Dec. 30, 2021.
[11] S. Fujimoto *et al.*, "Benchmarking Batch Deep Reinforcement Learning Algorithms," *Proc. Deep RL Workshop at NeurIPS'19*, Vancouver, Canada, Dec. 2019.
[12] V. Mnih *et al.*, "Human-Level Control Through Deep Reinforcement Learning," *Nature*, vol. 518, no. 7540, Feb. 2015, pp. 529–33.
[13] J. Dai *et al.*, "A View Synthesis-Based 360° VR Caching System over MEC-Enabled C-RAN," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 30, no. 10, Oct. 2020, pp. 3843–55.
[14] Y. Liu *et al.*, "MEC-Assisted Panoramic VR Video Streaming Over Millimeter Wave Mobile Networks," *IEEE Trans. Multimedia*, vol. 21, no. 5, May 2019, pp. 1302–16.
[15] Z. Liang *et al.*, "Multi-Cell Mobile Edge Computing: Joint Service Migration and Resource Allocation," *IEEE Trans. Wireless Commun.*, vol. 20, no. 9, Sept. 2021, pp. 5898–5912.

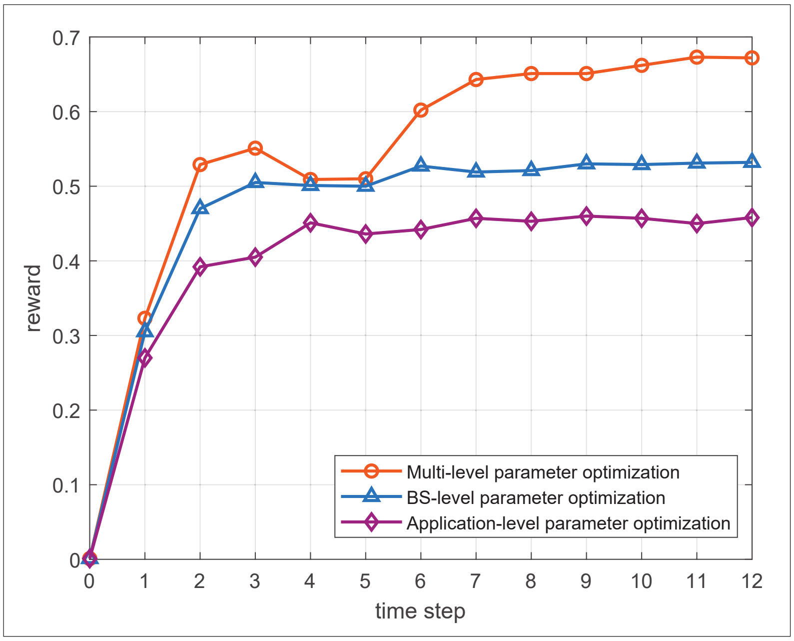

FIGURE 5. The benefit of multi-level parameter optimization.

Biographies

Yaohua Sun received the bachelors degree (with first class Hons.) in telecommunications engineering (with management) and the Ph.d. degree in communication engineering both from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2014 and 2019, respectively. He is currently an Assistant Professor with the State Key Laboratory of Networking and Switching Technology, BUPT. His research interests include intelligent radio access networks and integrated satellite-terrestrial networks. He has published over 30 papers including 3 ESI highly cited papers. He has been a Reviewer for *IEEE Transactions on Communications*, *IEEE Transactions on Mobile Computing*, etc.

Jianmin Chen received his master degree from the Beijing University of Posts and Telecommunications (BUPT) in 2022. His research interests include cloud VR and fog computing.

Zeyu Wang is currently pursuing her master degree in Beijing University of Posts and Telecommunications (BUPT). Her research interests include fog computing and resource allocation.

Mugen Peng received the Ph.D. degree in communication and information systems from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2005. Afterward, he joined BUPT, where he has been a Full Professor with the School of Information and Communication Engineering, since 2012. During 2014, he was also an Academic Visiting Fellow at Princeton University, USA. He has authored and coauthored over 90 refereed IEEE journal papers and over 300 conference proceeding papers. His main research interests include integrated satellite-terrestrial networks, fog radio access networks and the integration of sensing, communication and computing.

Shiwen Mao [S'99, M'04, SM'09] (smao@ieee.org) received his Ph.D. in electrical and computer engineering from Polytechnic University, Brooklyn, NY. He is a Professor and Earle C. Williams Eminent Scholar, and Director of the Wireless Engineering Research and Education Center at Auburn University. His research interests include wireless networks and multimedia communications. He is a Distinguished Lecturer of the IEEE Communications Society and IEEE Council of RFID. He is the Editor-in-Chief of IEEE Transactions on Cognitive Communications and Networking. He is a co-recipient of the 2021 Best Paper Award of Elsevier/KeAi Digital Communications and Networks, the 2021 IEEE Internet of Things Journal Best Paper Award, the 2021 IEEE Communications Society Outstanding Paper Award, the IEEE Vehicular Technology Society 2020 Jack Neubauer Memorial Award, the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems, and several best conference paper/demo awards.