

Received August 23, 2021, accepted September 2, 2021, date of publication September 6, 2021, date of current version September 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3110741

A Cooperative Caching and Computing-Offloading Method for 3C Trade-Off in VR Video Services

QI LI¹, DONGYU WANG¹, AND HUA LU²

¹Key Laboratory of Universal Wireless Communication, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China

²Guangdong Communications and Networks Institute, Guangdong 510700, China

Corresponding author: Dongyu Wang (dy_wang@bupt.edu.cn)

ABSTRACT Virtual reality (VR) video services are becoming increasingly popular in Internet of Things (IoT) owing to its immersive experience. The realization of immersive experience relies on extremely large amount of data transmitted, restricting its application. To meet the transmission requirement of VR video services, this paper designs a user-centric network based on the mobile edge computing (MEC) framework, where the 2D/3D field of view (FOV) files can be cached and the projection process from 2D FOV to 3D FOV can be computed at the VR device or MEC servers collaboratively. Based on this, a cooperative caching and computing-offloading strategy is formulated as a decision matrix, and its optimal closed-form expression is obtained by minimizing the transmission requirement under the strict service time constraint to realize the trade-off between communication, caching and computing (3C trade-off). Besides, the effect of the transmission requirement on the network density is also analyzed to provide useful guidelines for network design. Simulation results demonstrate that the proposed method can achieve quasi-optimal performance compared with other contrasting schemes.

INDEX TERMS VR video services, MEC, wireless caching, FOV, 3C trade-off.

I. INTRODUCTION

Due to the immersive experience and full fidelity for user's perception, wireless virtual reality (VR) emerging as an important application of future Internet of Things (IoT) has attracted more and more attentions [1]. A market report estimates that the global multimedia data of VR devices has increased over 650% between 2017 and 2021 [2]. In addition, the immersive experience of VR video mainly depends on the delivery of massive amount of data (in Gigabyte) within an ultra-low latency, in generally, 20 ms. Thus, it requires an ultra-high transmission rate, resulting in a significant burden on the wireless backhaul link [3].

It is noted that the field of views (FOVs) for different users may be not the same when they watch the same 360° VR video, which are decided by the trackers at VR devices. The FOV transmission schemes introduced in [4], including the Pyramid Projection transmission scheme of Facebook and the Tile Wise transmission scheme of Huawei, mainly focus on the transmission of the high-quality image in the current viewing FOV, effectively reducing the

amount of transmission data compared with the Full FOV transmission scheme where the entire panoramic video needs to be transmitted. In order to alleviate the transmission burden over the backhaul link and improve the quality of experience (QoE), we adopt the FOV transmission schemes and just consider the transmission of viewing FOV in the paper. The typical VR framework as shown in Fig. 1 of [1] is introduced. The generation process of VR video includes: 1) Stitching process obtains a spherical video by combining the video sessions captured by a multicamera array. 2) Equirectangular projection process obtains a 2D video by unfolding the spherical video. 3) Extraction process obtains the 2D FOV corresponding to the viewpoint of the tracker at VR device from the 2D video. 4) Projection process maps the 2D FOV into 3D FOV. 5) Rendering process renders the 3D FOV during the display. Due the first three pre-processing processes composite the 360° video from a bulk of sessions or require the entire 360° video as input, these three processes should be completed offline in cloud server (CS) with centralized computing function. Without doubt, the rendering process is completed at the VR device. Thus, where the projection process is completed is the key issue of releasing the wireless backhaul link from heavy transmission pressure of VR video.

The associate editor coordinating the review of this manuscript and approving it for publication was Petros Nicopolitidis¹.

The typical paradigm of mobile edge computing (MEC) is envisioned as one of the key enablers for VR video services, which combines wireless transmission network and computing-caching capabilities at edge network (including MEC server and VR device) [5]–[7]. On the one hand, the caching capability can be leveraged to proactively store some FOV files at edge network during off-peak times for future requests [8]. On the other hand, the ever-increasing computing capability of edge center processing unit (CPU) can be utilized to operate the projection process from 2D FOV to 3D FOV for reducing the service time [9]. Thus, in this paper, we aim to analyze the service mechanism for VR video services in a user-centric MEC network and find out how to make the best use of the caching-computing capabilities at edge network to minimize the transmission requirement while satisfying the stringent service time constraint, i.e., achieving the trade-off between communication, caching and computing (3C trade-off). Our contributions consist of three components:

- We present a user-centric network based on the MEC framework for VR video services. The network allows the FOV files to be cached and the projection process to be computed at the VR device or MEC servers collaboratively, thereby obtaining the collaborative gain and significantly reducing the transmission requirement under the strict service time constraint of VR videos.
- We propose a cooperative caching and computing-offloading strategy, and formulate the strategy as a decision matrix with 27 possible values. Based on the possible values of the decision matrix, the service mechanism of VR video is analyzed, and the required minimum transmission rate (MTR) is derived. We formulate an optimization problem of the decision matrix to minimize the required MTR. By analyzing the decision properties, a closed-form expression for the optimal decision matrix is obtained based on the 3C trade-off.
- Based on the optimal decision matrix, we analyze the effect of the transmission requirement on the network density, and the expression of the ideal network density (IND) in the limited caching capability scenario is derived to provide useful guidelines for network designers.

The remainder of this paper is organized as follows. Related work is presented in Section II. We present the system model in Section III, including the user-centric network architecture, and the cooperative caching and computing-offloading strategy. The service mechanism is analyzed in Section IV. In Section V, we formulate the system problem and analyze the decision property. Simulation results are presented in Section VI and Section VII concludes the paper.

II. RELATED WORK

In the last few years, academia researchers have made great efforts in the development of the VR video services. The realization of immersive VR experience relies on the

transmission and processing of extremely large amount of data. [10] studied the cache-assisted 360° video streaming to increase the overall quality of the delivered 360° videos to users and reduce the service cost. But the entire panoramic video needed to be transmitted in the network of [10]. Since each user only watched a viewpoint of the 360° VR video at any given time, the requested FOV was chosen to be transmitted instead of the entire panoramic video, thereby saving transmission resource significantly [4], [11], [12]. In [13], the authors proposed a VR QoE evaluation framework based on the transmission parameters, where an improved two-step neural network model was developed by using the features of the physiological psychology and cognitive neurology. However, these works mainly focused on the VR video itself, ignoring the potentials brought by the deployment of MEC network.

The potentials in the MEC network for VR video services can be fully tapped via efficiently utilizing caching-computing capabilities of edge network, which has been studied in [1], [14]–[25]. In [14], a Device-to-Device (D2D)-assisted VR video distribution system and a pre-caching algorithm based on QoE gain were proposed. A view synthesis-based 360° VR caching system was designed [15], where a hierarchical collaborative caching problem was formulated to minimize the transmission latency. To further improve the QoE of VR video services, [16]–[25] envisioned the joint computing-caching capabilities of edge network as the key enablers to obtain more potential gains. [16] proposed a collaborative caching allocation and computation offloading policy, where the MEC servers collaborated for executing computation tasks and data caching. An optimization framework for VR video services was formulated in a cache-enabled cooperative network and the fundamental 3C trade-off for VR services was explored [17]. A mobile VR delivery framework was presented [22], where the authors formulated a joint radio communication, caching and computing decision problem to maximize the average tolerant delay within a given transmission rate constraint. [23] proposed to minimize the long-term energy consumption of a terahertz wireless access-based MEC system for high quality immersive VR video services. [25] exploited the potential multicast opportunities by utilizing characteristics of multi-quality 360° VR videos and computation resources only at the users' side.

It is worthy to note that all the works in [1], [14]–[25] only utilized the caching-computing capabilities of MEC servers or the VR device, ignoring the collaboration gain from both two, such as the selection between the route of caching 2D FOV at MEC servers & computing locally and the route of caching & computing 2D FOV at MEC servers in different scenarios. The authors in [26] proposed a MEC-based 360° VR video streaming scheme, where the user's viewpoint prediction was utilized to cache video data proactively and computing tasks were partially offloaded to the MEC server. However, [26] did not consider the choice between computing at MEC servers and computing

TABLE 1. Some parameter notations used in this paper.

Symbol	Definition
M	The number of video viewpoints
\mathcal{P}	The set of viewpoints
q_m	The probability that the VR device requests the viewpoint m
S	The size of 2D FOV
\tilde{S}	The size of 3D FOV
ρ	The ratio of 3D FOV' size to 2D FOV'
η	The number of computation cycles required to process one bit data
K	The number of APs
Φ_M	The set of APs
R_M	The propagation radius of AP
Q_D	The number of 2D FOVs that can be cached in local server
Q_M	The number of 2D FOVs that can be cached in each MEC server
\mathbf{c}_m	The caching decision for the 2D FOV of viewpoint m
$\tilde{\mathbf{c}}_m$	The caching decision for the 3D FOV of viewpoint m
\mathbf{d}_m	The computing-offloading decision for the projection process of viewpoint m at the VR device or MEC servers
\mathbf{D}^m	The decision matrix of viewpoint m
\mathcal{D}	The decision matrix set
f_D	The CPU-cycle frequency of VR device
f_M	The CPU-cycle frequency of MEC server
κ	The CPU power efficiency parameter of VR device and MEC server
P_D	The maximum of power consumption
τ	The service time constraint
ς	The transmission penalty factor

locally. In addition, the works in [16]–[26] did not analyze a specific realization of VR framework based on the MEC network. [1] proposed a typical VR framework but only focused on utilizing the caching-computing capabilities at the VR device to alleviate the communication burden. In the paper, we propose a cooperative caching and computing-offloading strategy in a user-centric network, collaboratively utilizing the caching and computing resources at both MEC servers and the VR device. In this regard, there are 27 kinds of decision, resulting in a sophisticated optimization problem. By analyzing the problem, we find the relationship between the caching-computing decision and 3C trade-off.

III. SYSTEM MODEL

In this section, we will introduce a cooperative caching and computing-offloading strategy based on the user-centric networks. A notation table for all involved parameters is shown in Table. 1.

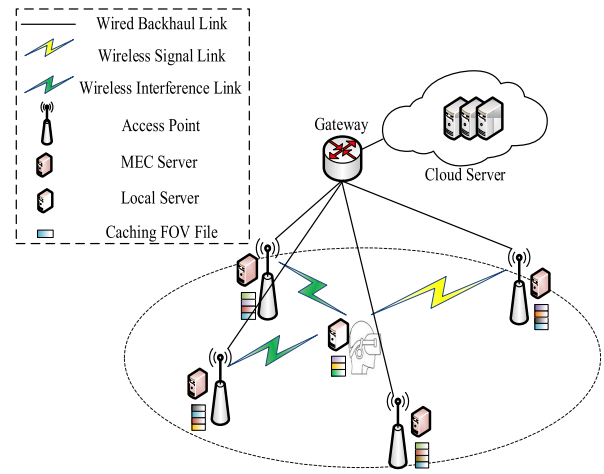


FIGURE 1. The figure of network architecture. There is an AP providing the requested FOV for the VR user while there also exists interference from other APs.

A. USER-CENTRIC NETWORK ARCHITECTURE

Considering the difference between 2D FOV and 3D FOV, an FOV-based VR video library is placed in CS where there are M video viewpoints from the extraction process, and both the 2D FOV and 3D FOV can be provided to users. In the scenario with the FOV-based VR video library, such as VR live streaming of sport events, multiple users in a single-cell theater watch an identical 360° VR video, but there are some differences between their viewpoints [1], [27]. The set of viewpoints is denoted as $\mathcal{P} = \{1, 2, \dots, M\}$. Considering the randomness of user's head rotation, the probability that the VR device requests the viewpoint $m \in \mathcal{P}$ is denoted as q_m , which characterizes how often the viewpoint m is tracked by the VR users as navigating the panoramic scene. According to [15], [28], we assume the viewpoint popularity as the uniform distribution, i.e., $q_m = \frac{1}{M}$. Besides, the sizes of 2D FOV and 3D FOV files of each viewpoint are denoted as S and \tilde{S} (bit), respectively. Since the projection process of each viewpoint has to be computed twice to create a stereoscopic vision, one of which is exposed to the user's left eye and the other to the right eye, the size of 3D FOV file is at least twice larger than that of the corresponding 2D FOV, i.e., $\rho = \frac{\tilde{S}}{S} \geq 2$ [1]. The projection computing task from 2D FOV into 3D FOV of any viewpoint can be modeled as a 3-tuple (S, \tilde{S}, η) , where η is the number of computation cycles required to process one bit data (cycle/bit).

Considering the fact that users cannot move around and their locations are relatively fixed when they watch VR video at the devices, the content delivery network of VR video is a user-centric network, as shown in Fig. 1. There are K single-antenna access points (APs) with MEC servers and an VR device with a local server [29]. More complicated transmission schemes such as multiple-input multiple-output transmission, which require antenna arrays are beyond the scope of this paper. The APs are evenly distributed in the cell with the center of the VR device and radius R_M , where R_M

is the propagation radius of AP. The set of APs is denoted as Φ_M . The MEC servers at APs possess caches of the same sizes $Q_M S$ (bit), while the local server at the VR device has a cache of size $Q_D S$ (bit). In generally, $Q_D < Q_M$ and both are integers. When the VR device is served by a chosen AP in the user-centric network, there may be some interferences from other active APs. Since the interference imposes more negative effect than white gaussian noise, the noise in the delivery process is assumed to be ignored.

B. COOPERATIVE CACHING AND COMPUTING-OFFLOADING STRATEGY

In the caching placement phase, the MEC servers and VR device store the FOV files in the collaborative manner of 0-1 caching, i.e., each FOV file is integrally stored or not. Considering the caching rule “largest diversity content (LDC)” and the fact that each of the K MEC servers can serve the VR device in the user-centric network, the caching resources of all MEC servers can be considered as a whole caching space of the size $KQ_M S$ through network function virtualization (NFV) to store the FOV files as different as possible. We denote the caching decision for the 2D FOV of viewpoint m as vector \mathbf{c}_m .

$$\mathbf{c}_m = \begin{cases} [1 \ 0 \ 0], & \text{No Caching} \\ [0 \ 1 \ 0], & \text{Local Caching} \\ [0 \ 0 \ 1], & \text{MEC Caching} \end{cases} \quad (1)$$

where $\mathbf{c}_m = [0 \ 1 \ 0]$ represents the corresponding 2D FOV can be found at the VR device locally, while $\mathbf{c}_m = [0 \ 0 \ 1]$ represents the FOV can be found in at least one MEC server. Otherwise, none of the VR device and MEC servers cache the FOV. Analogously, the caching decision for the 3D FOV of viewpoint m is denoted as $\tilde{\mathbf{c}}_m$.

$$\tilde{\mathbf{c}}_m = \begin{cases} [1 \ 0 \ 0], & \text{No Caching} \\ [0 \ 1 \ 0], & \text{Local Caching} \\ [0 \ 0 \ 1], & \text{MEC Caching} \end{cases} \quad (2)$$

The computing-offloading decision for the projection process of the viewpoint m at the VR device or MEC servers is denoted as vector \mathbf{d}_m , and we have

$$\mathbf{d}_m = \begin{cases} [1 \ 0 \ 0], & \text{No Computing} \\ [0 \ 1 \ 0], & \text{Local Computing} \\ [0 \ 0 \ 1], & \text{MEC Computing} \end{cases} \quad (3)$$

where $\mathbf{d}_m = [0 \ 1 \ 0]$ indicates the projection process of the corresponding viewpoint from 2D FOV to 3D FOV can be computed at the VR device, while $\mathbf{d}_m = [0 \ 0 \ 1]$ represents the projection process can be completed at one of the K MEC servers. Otherwise, none of VR device and MEC servers compute the process.

Combining the caching and computing-offloading decisions of the viewpoint m , we have a 3×3 decision matrix $\mathbf{D}^m = [\mathbf{c}_m^T \ \tilde{\mathbf{c}}_m^T \ \mathbf{d}_m^T]$. For example, $\mathbf{D}^m = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ includes

$\mathbf{c}_m = [0 \ 1 \ 0]$, $\tilde{\mathbf{c}}_m = [0 \ 0 \ 1]$ and $\mathbf{d}_m = [1 \ 0 \ 0]$, where $\mathbf{D}_{2,1}^m = 1$ denotes the element in the second row and first column is equal to 1. The decision matrices for all viewpoints constitute a decision matrix set \mathcal{D} .

Under the caching capacity constraints of the VR device and MEC servers, we have

$$\sum_{m=1}^M \mathbf{D}_{2,1}^m + \rho \mathbf{D}_{2,2}^m \leq Q_D \quad (4)$$

$$\sum_{m=1}^M \mathbf{D}_{3,1}^m + \rho \mathbf{D}_{3,2}^m \leq KQ_M \quad (5)$$

The VR device and MEC servers run at given CPU-cycle frequencies, denoted as f_D and f_M (cycle/s), respectively. In generally, $f_M > f_D$. The energy consumed for computing one cycle with frequency f_D at VR device is κf_D^2 , where κ is the CPU power efficiency parameter of VR device as well as MEC server [1], [30]. The VR device is assumed to be equipped with a fixed and finite energy capacity, and P_D (W) is denoted as the maximum of power consumption at VR device. In order to make sure that users can experience a full VR video, a power consumption constraint is considered, and we have

$$\frac{\kappa f_D^2 S \eta \sum_{m=1}^M \mathbf{D}_{2,3}^m}{M \tau} \leq P_D \quad (6)$$

where τ (second) is the maximum allowed time of serving a viewpoint request under the service time constraint, which can also be the maximum time of computing a projection process. From (6), $\frac{\kappa f_D^2 S \eta}{\tau}$ is the power consumption for calculating one projection process at VR device and it's noted that $\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \rfloor$ is the maximum number of projection processes that can be computed at the VR device, where $\lfloor \cdot \rfloor$ is the floor operation. In addition, the MEC servers are assumed to be directly connected to the grid so that the power consumption for projection processes at the MEC servers is unlimited and beyond the consideration of this paper.

IV. SERVICE MECHANISM AND ANALYSIS

In order to avoid dizziness and nausea, each viewpoint request from the VR device must be satisfied within the maximum of service time τ . Given the limited transmission capacity of the wired backhaul link from CS to MEC servers, the transmission rate of the link is very small and the transmission time is a much larger value when downloading 2D/3D FOV file from CS. Thus, the transmission time over the wired backhaul link is assumed to be a constant value $\zeta \tau$, where $\zeta \gg 1$ acts as the transmission penalty factor and is mainly related to the request waiting queue. Based on the decision matrix \mathbf{D}^m with 27 possible values, the request for the viewpoint m can be served by the following seven routes, and the values of \mathbf{D}^m corresponding to each route are derived in APPENDIX A. According to the seven routes, we would derive the minimum transmission rate (MTR) from MEC server to VR device while satisfying the service time constraint.

1) Local 3D Caching (L3C)

If $\mathbf{D}_{2,2}^m = 1$, the 3D FOV of the corresponding viewpoint can be obtained locally without the need of extra computing or transmission. In this regard, the service time is negligible and the required MTR is $R^{L3C} = 0$.

2) Local Computing with Local 2D Caching (LC2C)

If $(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m = 1$, the VR device obtains the 2D FOV of the corresponding viewpoint locally, and then projects it into 3D FOV using local CPU. The service time is $\frac{S\eta}{f_D}$ and the required MTR is $R^{LC2C} = 0$. Due to the fact that the computing capability of the VR device has taken huge steps, it's assumed that $\frac{S\eta}{f_D} \leq \tau$, i.e., the projection process computed at the VR device can be completed within the service time constraint.

3) Local Computing without Local 2D Caching (LC)

If $(1 - \mathbf{D}_{2,2}^m) (1 - \mathbf{D}_{2,1}^m) \mathbf{D}_{2,3}^m = 1$, the VR device downloads the 2D FOV of from the MEC servers or CS, and then projects it into 3D FOV using local CPU. When $\mathbf{c}_m = [0\ 0\ 1]$, i.e., $(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{3,1}^m \mathbf{D}_{2,3}^m = 1$, the corresponding 2D FOV can be obtained from the closest MEC server who has stored the file. This route is called as LC-FM. When $\mathbf{c}_m = [1\ 0\ 0]$ and $\tilde{\mathbf{c}}_m = [1\ 0\ 0]$, i.e., $\mathbf{D}_{1,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{2,3}^m = 1$, the nearest MEC server from the device is chosen as the relay, and retrieves the corresponding 2D FOV from CS and then transmits it to the device. The route is called as LC-FS.

The service time under the route of LC can be expressed as $\frac{S}{R^{LC}} + \frac{S\eta}{f_D} + (1 - \mathbf{D}_{3,1}^m) \zeta \tau$, where R^{LC} is the required MTR of LC route. More specifically, the service time under the route of LC-FM is $\frac{S}{R^{LC-FM}} + \frac{S\eta}{f_D}$, and we have $\frac{S}{R^{LC-FM}} + \frac{S\eta}{f_D} \leq \tau$ under the service time constraint. Hence, the required MTR of LC-FM can be derived as $R^{LC-FM} = \frac{S}{\tau - \frac{S\eta}{f_D}}$. The service time under the route of LC-FS is $\zeta \tau + \frac{S}{R^{LC-FS}} + \frac{S\eta}{f_D}$. Since the latter two terms in the above formula are all nonnegative, we have $\zeta \tau + \frac{S}{R^{LC-FS}} + \frac{S\eta}{f_D} \gg \tau$, i.e., the route of LC-FS does not satisfy the service time constraint.

4) MEC 3D Caching (M3C)

If $\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m) = 1$, the VR device downloads the 3D FOV of the corresponding viewpoint from the closest MEC server who has stored the file. The service time is $\frac{S}{R^{M3C}}$. Under the service time constraint $\frac{S}{R^{M3C}} \leq \tau$, the required MTR is $R^{M3C} = \frac{S}{\tau}$.

5) MEC Computing with MEC 2D Caching (MC2C)

If $\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m = 1$, the closest MEC server who has cached the corresponding 2D FOV file projects it into 3D FOV using MEC CPU, and then transmits the 3D FOV file to the VR device. The service time is $\frac{S\eta}{f_M} + \frac{S}{R^{MC2C}}$. Under the constraint $\frac{S\eta}{f_M} + \frac{S}{R^{MC2C}} \leq \tau$, the required MTR is $R^{MC2C} = \frac{S}{\tau - \frac{S\eta}{f_M}}$.

6) MEC Computing without MEC 2D Caching (MC)

If $(1 - \mathbf{D}_{3,1}^m) \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m = 1$, the closest MEC server from the VR device downloads the corresponding 2D FOV file from CS and projects it into 3D FOV using MEC CPU, and then transmits the 3D FOV file to the device. The service time is $\zeta \tau + \frac{S\eta}{f_M} + \frac{S}{R^{MC}}$. Since the latter two terms in the above formula are all nonnegative, $\zeta \tau + \frac{S\eta}{f_M} + \frac{S}{R^{MC}} \gg \tau$, i.e., the route of MC does not satisfy the service time constraint.

7) CS Downloading (SD)

If $\mathbf{D}_{1,2}^m \mathbf{D}_{1,3}^m = 1$, the closest MEC server from the VR device is chosen as the relay, who downloads the corresponding 3D FOV file from CS and then transmits it to the device without the need of extra computing. The service time is $\zeta \tau + \frac{S}{R^{SD}}$, and we also have $\zeta \tau + \frac{S}{R^{SD}} \gg \tau$, i.e., the route of SD does not satisfy the service time constraint.

There is a duplicate value between $(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{3,1}^m \mathbf{D}_{2,3}^m = 1$ (the route of LC-FM) and $\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m) = 1$ (the route of M3C), and the value is $\mathbf{D}^m = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$. In this

case, the routes of LC-FM and M3C can all be adopted. With the regard of the required MTRs, $R^{LC-FM} \leq R^{M3C}$ when $f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$. Hence, when $f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$, the route of LC-FM is adopted. Otherwise, the M3C route is applied.

Remark 1: If $f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$, the route of LC-FM is adopted when $(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{3,1}^m \mathbf{D}_{2,3}^m = 1$, and M3C is applied when $\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,3}^m + \mathbf{D}_{1,1}^m \mathbf{D}_{2,3}^m) = 1$. If $f_D < \frac{\rho S \eta}{(\rho-1)\tau}$, LC-FM is adopted when $\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{2,3}^m = 1$, and M3C is applied when $\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m) = 1$.

Considering that there is a large transmission penalty in the routes of LC-FS, MC and SD, the service times corresponding to the three routes are hugely large. We assume that the transmission rates corresponding to the three routes as $R^P = \zeta \frac{S}{\tau}$ for simplicity. Based on the above analysis, the 3C trade-off under the decision matrix \mathbf{D}^m when $f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$ is shown in Table. 2, and Table. 3 indicates the differences of 3C trade-off between $f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$ and $f_D < \frac{\rho S \eta}{(\rho-1)\tau}$.

V. PROBLEM FORMULATION AND PROPERTY ANALYSIS

In this section, a system problem is formulated based on the MTR, and we analyze the property based on the decision matrix.

A. PROBLEM FORMULATION

For any given decision matrix \mathbf{D}^m , the expression of MTR for random request of viewpoint has the following two cases.

$$1) f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$$

The value of MTR is shown in the formula (7), at the bottom of the next page.

$$2) f_D < \frac{\rho S \eta}{(\rho-1)\tau}$$

The value of MTR is shown in the formula (8), at the bottom of the next page.

TABLE 2. 3C trade-off under the decision matrix \mathbf{D}^m when $f_D \geq \frac{\rho S \eta}{(\rho-1)\tau}$.

\mathbf{D}^m	Route	MTR	Caching Resource	Computing Power Consumption
$\mathbf{D}_{2,2}^m = 1$	L3C	0	\tilde{S} in Local Cache	0
$(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m = 1$	LC2C	0	S in Local Cache	$\frac{\kappa f_D^2 S \eta}{\tau}$ in Local CPU
$(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{3,1}^m \mathbf{D}_{2,3}^m = 1$	LC-FM	R^{LC-FM}	S in MEC Servers	$\frac{\kappa f_D^2 S \eta}{\tau}$ in Local CPU
$\mathbf{D}_{1,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{2,3}^m = 1$	LC-FS	R^P	0	$\frac{\kappa f_D^2 S \eta}{\tau}$ in Local CPU
$\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,3}^m + \mathbf{D}_{1,1}^m \mathbf{D}_{2,3}^m) = 1$	M3C	R^{M3C}	\tilde{S} in MEC Servers	0
$\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m = 1$	MC2C	R^{MC2C}	S in MEC Servers	$\frac{\kappa f_M^2 S \eta}{\tau}$ in MEC CPU
$(1 - \mathbf{D}_{3,1}^m) \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m = 1$	MC	R^P	0	$\frac{\kappa f_M^2 S \eta}{\tau}$ in MEC CPU
$\mathbf{D}_{1,2}^m \mathbf{D}_{1,3}^m = 1$	SD	R^P	0	0

TABLE 3. The differences of 3C trade-off when $f_D < \frac{\rho S \eta}{(\rho-1)\tau}$.

\mathbf{D}^m	Route	MTR	Caching Resource	Computing Power Consumption
$\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{2,3}^m = 1$	LC-FM	R^{LC-FM}	S in MEC Servers	$\frac{\kappa f_D^2 S \eta}{\tau}$ in Local CPU
$\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m) = 1$	M3C	R^{M3C}	\tilde{S} in MEC Servers	0

Therefore, the system problem is formulated as follows.

Problem 1: $\min_{\mathcal{D}} MTR$

$$s.t. (4), (5) \quad \sum_{m=1}^M \mathbf{D}_{2,3}^m \leq \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor \quad (9)$$

$$\begin{aligned} \mathbf{D}_{i,j}^m &\in \{0, 1\}, \forall i, j \in \{1, 2, 3\}; \\ \sum_{i=1}^3 \mathbf{D}_{i,j}^m &= 1, \forall j \in \{1, 2, 3\} \\ m &\in \mathcal{P} \end{aligned} \quad (10)$$

where the constraint (9) corresponds to the constraint of the maximum number of projection processes that can be computed at the VR device. The constraint (10) limits the values of the three column vectors in the decision matrix.

B. DECISION PROPERTY ANALYSIS AND OPTIMAL SOLUTION

We denote the number of locally cached 2D FOV files, that of the locally cached 3D FOV files, that of the 2D FOV files stored in the MEC servers, that of the 3D FOV files stored in the MEC servers and that of the locally computed projection processes as $C_L = \sum_{m=1}^M \mathbf{D}_{2,1}^m$, $\tilde{C}_L = \sum_{m=1}^M \mathbf{D}_{2,2}^m$,

$C_M = \sum_{m=1}^M \mathbf{D}_{3,1}^m$, $\tilde{C}_M = \sum_{m=1}^M \mathbf{D}_{3,2}^m$, $D_L = \sum_{m=1}^M \mathbf{D}_{2,3}^m$, respectively.

In generally, $C_L + \tilde{C}_L + C_M + \tilde{C}_M \leq M$, i.e., the total number of the cached FOV files at the VR device and MEC servers is not more than the amount of viewpoints. From (4), (5) and (9), it's noted that $\tilde{C}_L \in \{0, 1, \dots, \lfloor \frac{Q_D}{\rho} \rfloor\}$, $C_L \in \{0, 1, \dots, Q_D - \rho \tilde{C}_L\}$, $\tilde{C}_M \in \{0, 1, \dots, \lfloor \frac{\kappa Q_M}{\rho} \rfloor\}$, $C_M \in$

$$MTR = \sum_{m=1}^M q_m \left\{ \begin{aligned} &(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{3,1}^m \mathbf{D}_{2,3}^m R^{LC-FM} + \mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,3}^m + \mathbf{D}_{1,1}^m \mathbf{D}_{2,3}^m) R^{M3C} \\ &+ \mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m R^{MC2C} + \mathbf{D}_{1,2}^m (\mathbf{D}_{1,3}^m + \mathbf{D}_{3,3}^m + \mathbf{D}_{1,1}^m \mathbf{D}_{2,3}^m - \mathbf{D}_{3,3}^m \mathbf{D}_{3,1}^m) R^P \end{aligned} \right\} \quad (7)$$

$$MTR = \sum_{m=1}^M q_m \left\{ \begin{aligned} &\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{2,3}^m R^{LC-FM} + \mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m) R^{M3C} + \\ &\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m R^{MC2C} + \mathbf{D}_{1,2}^m (\mathbf{D}_{1,3}^m + \mathbf{D}_{3,3}^m + \mathbf{D}_{1,1}^m \mathbf{D}_{2,3}^m - \mathbf{D}_{3,3}^m \mathbf{D}_{3,1}^m) R^P \end{aligned} \right\} \quad (8)$$

$\{0, 1, \dots, KQ_M - \rho\tilde{C}_M\}$ and $D_L \in \left\{0, 1, \dots, \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor\right\}$.

Since the projection computing task model (S, \tilde{S}, η) and the viewpoint popularity are the same for each viewpoint $m \in \mathcal{P}$. Without loss of generality, we have

$$\mathbf{D}_{2,2}^m = \begin{cases} 1, & m = 1, 2, \dots, \tilde{C}_L \\ 0, & \text{Otherwise} \end{cases} \quad (11)$$

It's noted that if the 3D FOV file of viewpoint m is already cached at the VR device, there is no need to store the corresponding 2D FOV at the device or to store the corresponding 2D/3D FOV file at the MEC servers, as well as to process the projection.

Property 1: For any viewpoint $m \in \mathcal{P}$, if $\mathbf{D}_{2,2}^m = 1$, we have $\mathbf{D}_{2,1}^m = \mathbf{D}_{3,1}^m = \mathbf{D}_{3,2}^m = 0$ and $\mathbf{D}_{1,3}^m = 1$.

Considering the caching rule "LDC", the 2D FOV and 3D FOV files stored at the device should be different viewpoints, and so are the 2D FOV and 3D FOV files stored at the MEC servers. Taking the caching capacity constraints (4) and (5) into consideration, we have

Property 2: If the values of \tilde{C}_L and \tilde{C}_M are given, C_L and C_M are equal to $Q_D - \rho\tilde{C}_L$ and $KQ_M - \rho\tilde{C}_M$, respectively.

$$\mathbf{D}_{2,1}^m = \begin{cases} 1, & m = \tilde{C}_L + 1, \tilde{C}_L + 2, \dots, \tilde{C}_L + C_L \\ 0, & \text{Otherwise} \end{cases} \quad (12)$$

On the other hand, the FOV files cached at the device and MEC servers should also belong to different viewpoints.

Property 3: For any viewpoint $m \in \mathcal{P}$, $\mathbf{D}_{2,2}^m + \mathbf{D}_{2,1}^m + \mathbf{D}_{3,2}^m + \mathbf{D}_{3,1}^m \leq 1$.

Therefore, the caching decision at the MEC servers is

$$\mathbf{D}_{3,1}^m = \begin{cases} 1, & m = \tilde{C}_L + C_L + 1, \dots, \tilde{C}_L + C_L + C_M \\ 0, & \text{Otherwise} \end{cases} \quad (13)$$

$$\mathbf{D}_{3,2}^m = \begin{cases} 1, & m = \left(\begin{matrix} \tilde{C}_L + C_L \\ + C_M + 1 \end{matrix} \right), \dots, \left(\begin{matrix} \tilde{C}_L + C_L \\ + C_M + \tilde{C}_M \end{matrix} \right) \\ 0, & \text{Otherwise} \end{cases} \quad (14)$$

When the 2D FOV file of viewpoint m is stored in the local cache, the projection process should be completed using local CPU, and there is no need to store the corresponding 3D FOV file at the local cache or MEC servers and to compute at MEC servers. In addition, when the local CPU does not have enough ability to compute the projection process of viewpoint m , there is no need of caching the corresponding 2D FOV locally.

Property 4: For any viewpoint $m \in \mathcal{P}$, $\mathbf{D}_{2,2}^m + \mathbf{D}_{3,2}^m + \mathbf{D}_{2,3}^m + \mathbf{D}_{3,3}^m \leq 1$ & $\mathbf{D}_{2,1}^m \leq \mathbf{D}_{2,3}^m$.

When the 2D FOV file of viewpoint m is stored at the MEC servers, the projection process can be computed using local CPU or MEC CPU.

Property 5: For any viewpoint $m \in \mathcal{P}$, if $\mathbf{D}_{3,1}^m = 1$, there are two cases to compute the corresponding projection process.

$$1) \frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau$$

The time of computing a projection process at the VR device falls between $\frac{(\rho-1)\tau}{\rho}$ and τ . If the MEC CPU-cycle frequency $f_M \leq \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D}$, the route of LC-FM is adopted. Otherwise, MC2C is applied.

$$2) \frac{S\eta}{f_D} < \frac{(\rho-1)\tau}{\rho}$$

The time of computing a projection process at the VR device is less than $\frac{(\rho-1)\tau}{\rho}$. In this regard, we always choose the route of LC-FM.

Proof: Seen APPENDIX B.

Based on the **Property 4** and **Property 5**, the computing decision at the VR device is

$$\mathbf{D}_{2,3}^m = \begin{cases} 1, & m = \tilde{C}_L + 1, \tilde{C}_L + 2, \dots, \tilde{C}_L + D_L \\ 0, & \text{Otherwise} \end{cases} \quad (15)$$

When $\left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M \leq \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$ or $\frac{S\eta}{f_D} < \frac{(\rho-1)\tau}{\rho}$, the route of LC-FM is adopted. We should set the value of D_L as large as possible within the range of $C_L \leq D_L \leq C_L + C_M$ and $D_L \in \left\{0, 1, \dots, \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor\right\}$ through allocating proper local caching spaces for 2D and 3D FOV files, ensuring that each 2D FOV file cached at the VR device can be projected into 3D FOV using local CPU (the LC2C route). In this regard, the route of LC-FM is also adopted while some 2D FOV files from MEC servers can be projected into 3D FOV also using local CPU. When $\left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M > \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$, the route of MC2C is adopted. We should set $D_L = C_L$ to ensure that each 2D FOV file only cached at the device can be projected into 3D FOV using local CPU while the 2D FOV files stored at the MEC servers are projected using MEC CPU.

Due to the unlimited computing energy supply at MEC servers, we have

$$\mathbf{D}_{3,3}^m = \begin{cases} 1, & m = \tilde{C}_L + D_L + 1, \dots, \tilde{C}_L + C_L + C_M \\ 0, & \text{Otherwise} \end{cases} \quad (16)$$

Based on the above analysis, **Problem 1** is equivalent to **Problem 2**.

Problem 2 : $\min_{\mathcal{D}} MTR$

$$s.t. \ \tilde{C}_L \in \left\{0, 1, \dots, \left\lfloor \frac{Q_D}{\rho} \right\rfloor\right\} \quad (18)$$

$$C_L = Q_D - \rho\tilde{C}_L \quad (19)$$

$$\tilde{C}_M \in \left\{0, 1, \dots, \left\lfloor \frac{KQ_M}{\rho} \right\rfloor\right\} \quad (20)$$

$$C_M = KQ_M - \rho\tilde{C}_M \quad (21)$$

$$D_L \in \left\{0, 1, \dots, \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor\right\} \quad (22)$$

where MTR is shown in the formula (17), at the bottom of the next page, whether $f_D \geq \frac{\rho S\eta}{(\rho-1)\tau}$ or $f_D < \frac{\rho S\eta}{(\rho-1)\tau}$. The first term is the transmission rate in the route of LC-FS, MC and SD. The second term corresponds to the gain of L3C, which increases with the number of 3D FOV files locally cached,

i.e., \widetilde{C}_L . In generally, we have $C_L \leq D_L$, which means that all locally stored 2D FOV files can be projected into 3D FOV files using local CPU through adjusting the proportion $\frac{\widetilde{C}_L}{C_L}$ appropriately. The third term corresponds to the gain of LC2C, which increases with the number of 2D FOV files locally cached, i.e., C_L . The fourth term corresponds to the gain of LC-FM, which increases with the local remaining computing capabilities (excluding the capabilities of projecting the locally stored 2D FOV files), i.e., $D_L - C_L$. The fifth term corresponds to the gain of MC2C, which increases with the number of 2D FOV files cached in the MEC servers and excluding the portion downloaded to the VR device, i.e., $C_L + C_M - D_L$. The last term corresponds to the gain of M3C, which increases with the number of 3D FOV files cached at the MEC servers, i.e., \widetilde{C}_M .

Based on **Problem 2**, the optimal decision matrix set \mathcal{D} under the service time constraint can be obtained, yielding the 3C trade-off.

$$1) \left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M \leq \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right] \quad \text{or} \quad \frac{S\eta}{f_D} < \frac{(\rho-1)\tau}{\rho}$$

Strategy 1: The optimal decision matrix set \mathcal{D}^* is given as

$$C_L^* = \min \{Q_D, D_L^*\} \quad (23)$$

$$\widetilde{C}_L^* = \min \left\{ \frac{Q_D - C_L^*}{\rho}, M - C_L^* \right\} \quad (24)$$

$$C_M^* = \min \left\{ K Q_M, M - C_L^* - \widetilde{C}_L^* \right\} \quad (25)$$

$$\widetilde{C}_M^* = \min \left\{ \frac{K Q_M - C_M^*}{\rho}, M - C_L^* - \widetilde{C}_L^* - C_M^* \right\} \quad (26)$$

$$D_L^* = \min \left\{ \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor, M \right\} \quad (27)$$

$$2) \left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M > \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$$

Strategy 2: The optimal decision matrix set \mathcal{D}^* is given as

$$C_L^* = \min \left\{ Q_D, \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor, M \right\} \quad (28)$$

$$\widetilde{C}_L^* = \min \left\{ \frac{Q_D - C_L^*}{\rho}, M - C_L^* \right\} \quad (29)$$

$$C_M^* = \min \left\{ K Q_M, M - C_L^* - \widetilde{C}_L^* \right\} \quad (30)$$

$$\widetilde{C}_M^* = \min \left\{ \frac{K Q_M - C_M^*}{\rho}, M - C_L^* - \widetilde{C}_L^* - C_M^* \right\} \quad (31)$$

$$D_L^* = C_L^* \quad (32)$$

The *Strategy 1* is mainly limited by the local computing capability, while the *Strategy 2* is mainly limited by the local caching capability.

Proof: Seen APPENDIX C.

C. EFFECT OF TRANSMISSION REQUIREMENT ON NETWORK ARCHITECTURE

In the user-centric networks, the VR device receives the corresponding FOV file from the chosen AP as well as the interferences from other active APs through the wireless link. The actual Signal-to-Interference Ratio (SIR) at the device in the interference-limited network is

$$SIR = \frac{S_0}{\beta \sum_{k \in \Phi_M \setminus \{0\}} I_k} \quad (33)$$

where S_0 is the received signal from the chosen AP, I_k is the interference from one of the other APs who are serving other users. Which MEC server the requested FOV file is stored in is random in the NFV process of caching resources and APs are evenly distributed in the cell, so the probability distribution functions (PDFs) of S_0 and I_k are the same. In addition, considering that it is impossible that all APs are always idle and waiting to serve one user, we define β ($0 \leq \beta \leq 1$) as the proportion of active APs.

Considering the MTR, we have

$$W_M \log_2(SIR + 1) \geq MTR \quad (34)$$

where W_M is the transmission bandwidth of AP. From (34), we have $K \leq \frac{1}{\beta \left(2^{\frac{MTR}{W_M}} - 1 \right)} + 1$ from the perspective of mean

value. We define the density of the user-centric network as the density of APs, i.e., $\frac{K}{\pi R_M^2}$.

Remark 2: Based on the formulas (5), (7) and (8), there are more FOV files that can be cached at the MEC servers with the increasing of the MEC number K , resulting in the decreasing of MTR. But there is a constraint in the value of K , i.e., $K \leq \frac{1}{\beta \left(2^{\frac{MTR}{W_M}} - 1 \right)} + 1$, to satisfy the transmission require-

ment. Hence, the density of user-centric network should be set as large as possible in the limited caching capability scenario while satisfying the constraint of K , i.e., the ideal network

$$\text{density is } IND \triangleq \frac{\left\lfloor \frac{1}{\beta \left(2^{\frac{MTR}{W_M}} - 1 \right)} + 1 \right\rfloor}{\pi R_M^2}.$$

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed network by MATLAB. The simulation setup and performance analysis are presented as follows.

A. SIMULATION SETUP

According to the references [1], [6], [8], the default parameter setting for the user-centric network is shown in Table. 4. Considering user's head motion and the fact that the 2D FOV

$$MTR = R^P - \frac{\widetilde{C}_L}{M} R^P - \frac{C_L}{M} R^P - \frac{(D_L - C_L)}{M} \left(R^P - R^{LC-FM} \right) - \frac{(C_L + C_M - D_L)}{M} \left(R^P - R^{MC2C} \right) - \frac{\widetilde{C}_M}{M} \left(R^P - R^{M3C} \right) \quad (17)$$

TABLE 4. Parameter setting of the network.

Parameters	Values	Parameters	Values
M	7.2×10^5	ρ	2
S	3 Mbit	η	10 cycle/bit
K	3	R_M	100 m
Q_M	1.44×10^5	Q_D	0.36×10^5
f_M	5 GHz	f_D	2 GHz
κ	10^{-27}	P_D	5 W
τ	20 ms	ς	10
β	0.8	W_M	5 GHz

files are extracted from the 360° 2D videos, each user may request different set of viewpoints during each VR experience. We consider each 360° 2D video of 5 minutes for simplicity, and the size of 2D FOV file with the duration of 1 second is assumed to be 3 Mbit. Based on [1], each 360° 2D image is divided into 24 viewpoints and the division of each 2D image at any time is conducted 100 times, each with a unique combination of 24 viewpoints. Hence, there are $M = 24 \times 100 \times 5 \times 60 = 7.2 \times 10^5$ viewpoints. Since the size of each 3D FOV file is at least twice larger than that of 2D FOV, and we choose $\rho = 2$ and $\tilde{S} = 6$ Mbit. The number of MEC servers K is set as 3, and the total caching capacities of the VR device and MEC servers are assumed less than the total data size of FOV files in the limited caching capability scenario.

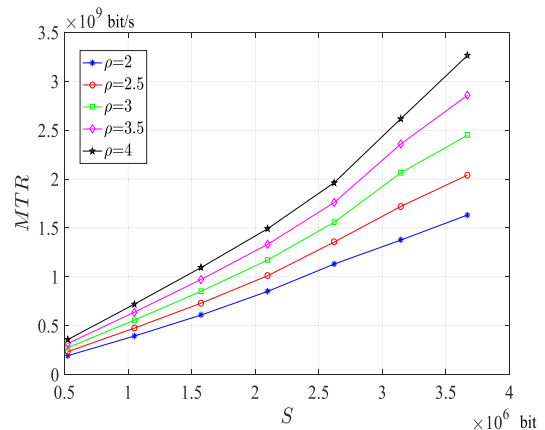
In order to analyze the network performance, the ratio of 3D FOV size and 2D FOV size ρ is set in $\{2, 2.5, 3\}$, the caching capacity of the VR device Q_D is set in the range from 0.072×10^5 to 1.368×10^5 , the MEC server number K is set in the range from 1 to 7, the CPU-cycle frequency of the VR device f_D is set in the range from 2 GHz to 5 GHz, and the averaged power consumption P_D is set in $\{2.5 W, 5 W, 7.5 W\}$.

To analyze the performance of the cooperative caching and computing-offloading method, three schemes are presented to be compared with the proposed strategy.

1) CS Downloading: There is no caching and computing capabilities at the VR device and APs. Any requested 3D FOV file is obtained directly from CS. Hence, we have $\mathbf{D}_{1,1}^m = \mathbf{D}_{1,2}^m = \mathbf{D}_{1,3}^m = 1, \forall m \in \mathcal{P}$. This scheme corresponds to the route of SD.

2) Only Caching: There are only caching capabilities at the VR device and APs without computing capabilities. Due to the caching rule ‘‘LDC’’, we have $\mathbf{D}_{2,2}^m = 1 (m = 1, 2, \dots, \lfloor \frac{Q_D}{\rho} \rfloor)$, $\mathbf{D}_{3,2}^m = 1 (m = \lfloor \frac{Q_D}{\rho} \rfloor + 1, \lfloor \frac{Q_D}{\rho} \rfloor + 2, \dots, \lfloor \frac{Q_D}{\rho} \rfloor + \lfloor \frac{KQ_M}{\rho} \rfloor)$ and $\mathbf{D}_{1,3}^m = 1 (\forall m \in \mathcal{P})$. This scheme includes the routes of L3C, M3C and SD.

3) Independent Caching and Computing (IC&C): There is no cooperation between the VR device and MEC servers. The caching capacity at the VR device is utilized for storing 3D FOV files, and the device obtains the other 3D FOV files from MEC servers or CS. The caching and computing capabilities at the MEC servers are totally utilized for storing

FIGURE 2. S versus MTR under different ρ .

and projecting 2D FOV files. Hence, we have $\mathbf{D}_{2,2}^m = 1 (m = 1, 2, \dots, \lfloor \frac{Q_D}{\rho} \rfloor)$, $\mathbf{D}_{3,1}^m = 1 (m = \lfloor \frac{Q_D}{\rho} \rfloor + 1, \lfloor \frac{Q_D}{\rho} \rfloor + 2, \dots, \lfloor \frac{Q_D}{\rho} \rfloor + KQ_M)$ and $\mathbf{D}_{3,3}^m = 1 (m = \lfloor \frac{Q_D}{\rho} \rfloor + 1, \lfloor \frac{Q_D}{\rho} \rfloor + 2, \dots, \lfloor \frac{Q_D}{\rho} \rfloor + KQ_M)$. This scheme includes the routes of L3C, MC2C and SD.

B. SIMULATION RESULTS AND DISCUSSIONS

Fig. 2 shows the impact of video file size on MTR under different values of ρ . Obviously, the increasing of S results that there are less 2D FOV files cached in the limited caching spaces of VR device or MEC servers. Thus, the VR device needs to obtain the un-cached files from CS, leading to a transmission penalty and the increasing of MTR. On the other hand, when the value of S is fixed, the size of 3D FOV file increases with ρ , limiting the number of the stored 3D FOV files and the number of the transmitted 3D FOV files under the service time constraint, which also leading to the increasing of MTR.

The effect of Q_D on MTR under different values of ρ and the comparison with other schemes are shown in Fig. 3. In the proposed strategy, the local caching hit probability increases with Q_D , resulting in the decreasing of the probability to fetch the required FOV files from CS as well as the minishing of MTR. On the other hand, the size of 3D FOV file increases with ρ , limiting the number of the stored 3D FOV files and the number of the transmitted 3D FOV files under the service time constraint, which finally leading to the increasing of MTR. Besides, due to no caching capability in the CS downloading scheme, there is no variation of MTR with the changing Q_D . The reason of the variation trends in the only caching scheme and IC&C scheme is the same as the proposed strategy. In addition, the value of MTR in the proposed strategy is obviously superior to those in other schemes. This indicates that the cooperation of the 2C resource (including caching and computing) between the device and MEC servers can put a huge gain on the remaining 1C, i.e., communication. When the value of Q_D is small, MTR in the proposed strategy closes to that in the IC&C scheme. This is because there are

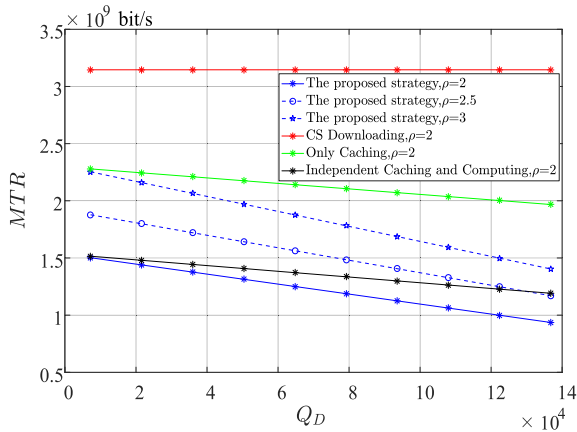


FIGURE 3. Q_D versus MTR under different ρ compared with other schemes.

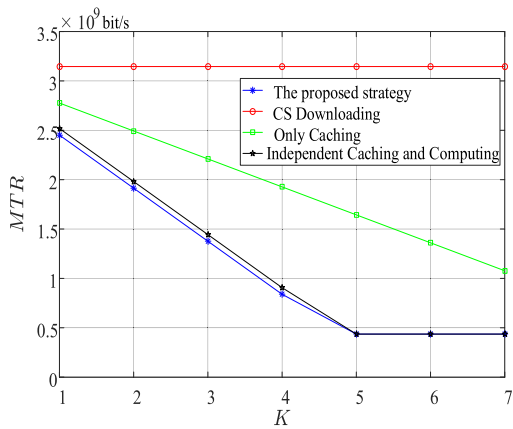


FIGURE 4. K versus MTR and the comparison with other schemes.

few FOV files cached locally and little difference in the local caching gain. But with the increasing of Q_D , the proposed strategy can obtain more gains in the route of LC2C at the cost of fewer occupied local caching resource rather than in the route of L3C, resulting in the significant reduction of MTR .

Fig. 4 shows the comparison of the effect of K on MTR in the proposed strategy with other schemes. Due to the NFV process, the effect of K on MTR can be considered as the effect of the caching capability of the whole MEC servers on MTR . In the proposed strategy, the caching hit probability at the MEC servers increases with K in the range from 1 to 5, resulting in the minishing of MTR . When $K > 5$, all the FOV files are cached at the device or MEC servers, and so there is no variation of MTR with the changing K . However, the total data size of the 3D FOV files that can be stored in the only caching scheme is larger than those of the proposed strategy where the cached files include 2D FOV and 3D FOV, so the value of MTR of the only caching scheme decreases with K in the range of $1 \leq K \leq 7$. Besides, the gains obtained from the routes of LC-FM and M3C close to that of MC2C, so the two curves corresponding to the proposed strategy and the IC&C scheme approach within the range of $1 \leq K \leq 5$.

Fig. 5 shows the effect of f_D on MTR under different values of P_D . In the proposed strategy, the local

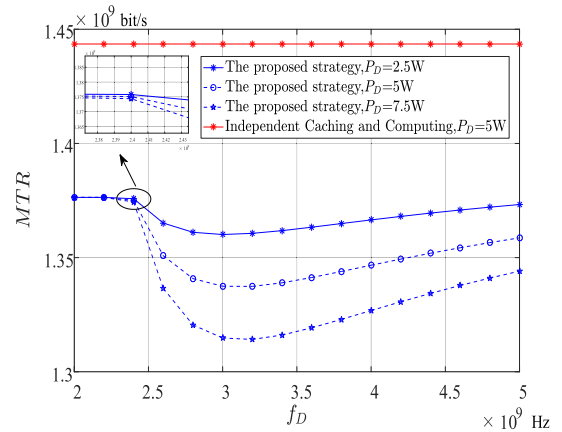
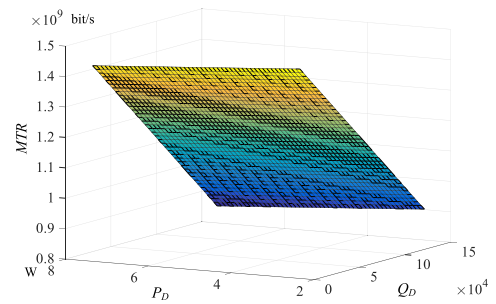
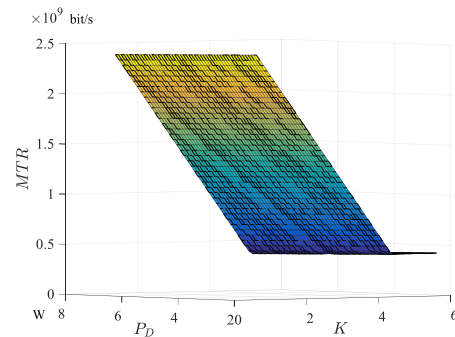


FIGURE 5. f_D versus MTR under different P_D .



(a) Q_D and P_D versus MTR .



(b) K and P_D versus MTR .

FIGURE 6. The analysis of 3C trade off.

computing capability increases with P_D , which leads to the increasing of the probability of LC2C and LC-FM and the decreasing of MTR . When $f_D \leq 2.4 \times 10^9$, $\left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M > \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$ and the **Strategy 2** is adopted. When $f_D > 2.4 \times 10^9$, $\left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M \leq \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$ and the **Strategy 1** is applied. This is the reason why the curves of the proposed strategy begin going down suddenly in $f_D = 2.4 \times 10^9$. The value of MTR is mainly limited by the caching capability at the VR device when the **Strategy 2** is applied in the range of $2.0 \times 10^9 < f_D \leq 2.4 \times 10^9$, so MTR is little affected by

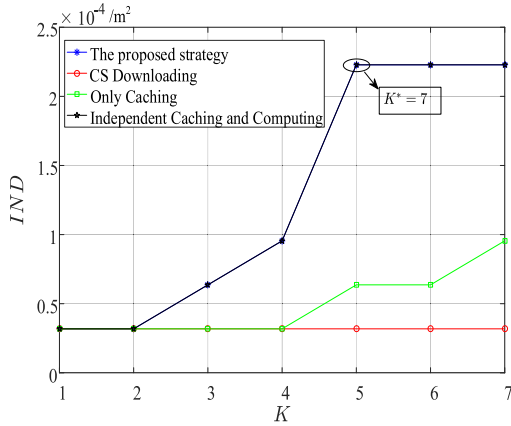


FIGURE 7. K versus IND and the comparison with other schemes.

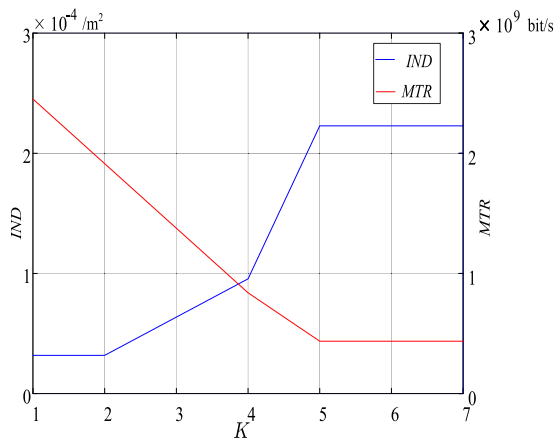


FIGURE 8. The relationship between MTR and IND under the changing K.

f_D . The local computing number decreases with f_D when the **Strategy 1** is applied in the range of $f_D > 2.4 \times 10^9$, resulting in the increasing of MTR. In addition, the three contrast schemes have nothing to do with the computing capability of the VR device, so the IC&C scheme is just chosen as a benchmark. Obviously, the local computing gain from the VR device is remarkable.

In Fig. 6, the trade of between caching, computing and communication is analyzed when $f_D = 3 \times 10^9$. From Fig. 6 (a), we can see that the value of MTR decreases at the cost of occupying more caching or computing resources in the VR device. Obviously, when there are more caching spaces in the MEC servers or more abundant computing resources in the VR device, MTR also decreases in Fig. 6 (b).

The comparison of the effect of K on the ideal network density IND in the proposed strategy with other schemes is shown in Fig. 7. The value of MTR decreases with K as shown in Fig. 4, which means a smaller actual transmission rate is sufficient to meet the transmission requirement. Thus, IND increases with K within the range of $1 \leq K \leq 5$. In the proposed strategy, there is no variation in the curve when $K > 5$. This is because that all the FOV files are cached at the device or MEC servers and there is no caching gain with the increasing of K, this scenario is also called

the abundant caching capability scenario. Considering the **Remark 2**, the optimal value of the AP number is 7 and the value of IND is 0.2228×10^{-3} in the limited caching capability scenario. The relationship between MTR and IND under the changing K is shown in Fig. 8. Obviously, when $K = 5$ under the default parameter setting, the value of MTR is optimal while the actual network density of APs is less than IND in the abundant caching capability scenario.

VII. CONCLUSION

In this paper, a user-centric network based on the MEC framework was proposed to meet the transmission requirement of VR video services. The 2D and 3D FOV files could be collaboratively cached while the projection process from 2D FOV to 3D FOV could also be computed at the VR device or MEC servers. Firstly, a cooperative caching and computing-offloading strategy was formulated as a decision matrix, and the service mechanism of VR video was analyzed. Secondly, an optimization problem based on the 3C trade-off was formulated to minimize the required MTR, and we obtained the closed-form expression for the optimal decision matrix by analyzing the decision properties. Finally, we analyzed the effect of the MTR on the network density and the expression of IND in the limited caching capability scenario was derived to provide useful guidelines for network designers. The effectiveness of the proposed strategy was verified by numerical examples.

APPENDIX A

Based on the decision matrix \mathbf{D}^m with 27 possible values, which route is chosen to response to the request for the viewpoint m is analyzed as follows.

1) When $\mathbf{D}^m = \begin{bmatrix} * & 0 & * \\ * & 1 & * \\ * & 0 & * \end{bmatrix}$ where $[* \ * \ *]^T$ represents the

value set of $[1 \ 0 \ 0]^T$, $[0 \ 1 \ 0]^T$ and $[0 \ 0 \ 1]^T$, the VR device can directly obtain the 3D FOV file of the corresponding viewpoint locally. The characteristic of \mathbf{D}^m is $\mathbf{D}_{2,2}^m = 1$ when the route of L3C is adopted.

2) When $\mathbf{D}^m = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ or $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$, the VR

device can obtain the corresponding 3D FOV file in the route of LC2C and the characteristic of the matrix is $(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m = 1$.

3) When $\mathbf{D}^m = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ or $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, the VR device

can obtain the corresponding 3D FOV in the route of LC-FM. The characteristic of \mathbf{D}^m is $(1 - \mathbf{D}_{2,2}^m) \mathbf{D}_{3,1}^m \mathbf{D}_{2,3}^m = 1$.

When $\mathbf{D}^m = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$, the VR device can obtain the cor-

responding 3D FOV in the route of LC-FS. The characteristic of \mathbf{D}^m is $\mathbf{D}_{1,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{2,3}^m = 1$.

4) When $\mathbf{D}^m = \begin{bmatrix} * & 0 & 1 \\ * & 0 & 0 \\ * & 1 & 0 \end{bmatrix}$, $\begin{bmatrix} * & 0 & 0 \\ * & 0 & 0 \\ * & 1 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ or $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, the VR device can obtain the corresponding

3D FOV in the route of M3C. The characteristic of \mathbf{D}^m is $\mathbf{D}_{3,2}^m (1 - \mathbf{D}_{2,1}^m \mathbf{D}_{2,3}^m) = 1$.

5) When $\mathbf{D}^m = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$, the VR device can obtain the

corresponding 3D FOV in the route of MC2C. The characteristic of \mathbf{D}^m is $\mathbf{D}_{3,1}^m \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m = 1$.

6) When $\mathbf{D}^m = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ or $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, the VR device

can obtain the corresponding 3D FOV in the route of MC. The characteristic of \mathbf{D}^m is $(1 - \mathbf{D}_{3,1}^m) \mathbf{D}_{1,2}^m \mathbf{D}_{3,3}^m = 1$.

7) When $\mathbf{D}^m = \begin{bmatrix} * & 1 & 1 \\ * & 0 & 0 \\ * & 0 & 0 \end{bmatrix}$, the VR device can obtain the

corresponding 3D FOV in the route of SD. The characteristic of \mathbf{D}^m is $\mathbf{D}_{1,2}^m \mathbf{D}_{1,3}^m = 1$.

In addition, there is a duplicate value of \mathbf{D}^m between the route of LC-FM and M3C, and this value is $\mathbf{D}^m = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$.

APPENDIX B

When the 2D FOV file of viewpoint m is stored at the MEC servers, whether the projection process is computed using local CPU or MEC CPU is decided by the MTRs of the routes LC-FM and MC2C.

$$R^{MC2C} - R^{LC-FM} = \frac{\tau S (\rho - 1) + S^2 \eta \left(\frac{1}{f_M} - \frac{\rho}{f_D} \right)}{\left(\tau - \frac{S\eta}{f_D} \right) \left(\tau - \frac{S\eta}{f_M} \right)}$$

When $R^{MC2C} - R^{LC-FM} \geq 0$, we have

$$\frac{S\eta}{\rho f_M} \geq \frac{S\eta}{f_D} - \frac{(\rho - 1)\tau}{\rho}$$

Due $\frac{S\eta}{f_D} \leq \tau$, there are two cases to be analyzed.

1) If $\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau$, we have $\frac{S\eta}{f_D} - \frac{(\rho-1)\tau}{\rho} \geq 0$. While $f_M \leq \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D}$, $R^{MC2C} - R^{LC-FM} \geq 0$ and the route of LC-FM is applied. When $f_M > \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D}$, $R^{MC2C} - R^{LC-FM} < 0$ and the route of MC2C is adopted.

2) If $\frac{S\eta}{f_D} < \frac{(\rho-1)\tau}{\rho}$, we have $\frac{S\eta}{f_D} - \frac{(\rho-1)\tau}{\rho} < 0$ and $R^{MC2C} - R^{LC-FM} \geq 0$ is always true. Thus, the route of LC-FM is applied.

APPENDIX C

When $\left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M \leq \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$ or $\frac{S\eta}{f_D} < \frac{(\rho-1)\tau}{\rho}$, we have $R^{LC-FM} \leq R^{MC2C}$ and $C_L \leq D_L \leq C_L + C_M$.

The value of MTR in the formula (17) can be simplified as

$$R = R^P - \frac{(Q_D + KQ_M)(R^P - R^{MC2C})}{M} - \frac{Q_D R^{LC-FM}}{M} + \frac{(\rho - 1)R^P}{M} \widetilde{C}_L + \frac{\rho(R^{LC-FM} - R^{MC2C})}{M} \widetilde{C}_L + \frac{(\rho - 1)R^P}{M} \widetilde{C}_M + \frac{R^{M3C} - \rho R^{MC2C}}{M} \widetilde{C}_M + \frac{R^{LC-FM} - R^{MC2C}}{M} D_L$$

In the above formula, the first three terms are constant. In the fourth and fifth terms, $R^P > 0$ and $R^{LC-FM} - R^{MC2C} \leq 0$, but the order of magnitude of R^P is much larger than that of $R^{LC-FM} - R^{MC2C}$. Hence, the value of \widetilde{C}_L should decrease to minimize MTR . Due to the fact that $R^{M3C} < R^{MC2C}$, $R^P > 0$ and $R^{M3C} - \rho R^{MC2C} < 0$ in the sixth and seventh terms, but the order of magnitude of R^P is also much larger than that of $R^{M3C} - \rho R^{MC2C}$. Hence, the value of \widetilde{C}_M should decrease to minimize MTR . In the last term, due to $R^{LC-FM} - R^{MC2C} \leq 0$, we should set D_L as large as possible to minimize MTR , i.e., $D_L^* = \min \left\{ \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor, M \right\}$. Due to the constraints (19) and (21), a smaller C_L means a larger C_L , so the relationship between \widetilde{C}_M and C_M is. Hence, we have $C_L^* = \min \{Q_D, D_L^*\}$, $\widetilde{C}_L^* = \min \left\{ \frac{Q_D - C_L^*}{\rho}, M - C_L^* \right\}$, $C_M^* = \min \{KQ_M, M - C_L^* - \widetilde{C}_L^*\}$ and $\widetilde{C}_M^* = \min \left\{ \frac{KQ_M - C_M^*}{\rho}, M - C_L^* - \widetilde{C}_L^* - C_M^* \right\}$.

When $\left[\frac{(\rho-1)\tau}{\rho} \leq \frac{S\eta}{f_D} \leq \tau \ \& \ f_M > \frac{S\eta f_D}{\rho S\eta - (\rho-1)\tau f_D} \right]$, we have $R^{LC-FM} > R^{MC2C}$ and $C_L = D_L$. The value of MTR can be simplified as

$$R = R^P - \frac{Q_D R^P}{M} - \frac{KQ_M(R^P - R^{MC2C})}{M} + \frac{(\rho - 1)R^P}{M} \widetilde{C}_L + \frac{(\rho - 1)R^P}{M} \widetilde{C}_M + \frac{R^{M3C} - \rho R^{MC2C}}{M} \widetilde{C}_M$$

In the above formula, the first three terms are constant. In order to minimize the value of MTR , \widetilde{C}_L , \widetilde{C}_M should decrease as much as possible. Hence, we have $C_L^* = \min \left\{ Q_D, \left\lfloor \frac{P_D M \tau}{\kappa f_D^2 S \eta} \right\rfloor, M \right\}$, $D_L^* = C_L^*$, $\widetilde{C}_L^* = \min \left\{ \frac{Q_D - C_L^*}{\rho}, M - C_L^* \right\}$, $C_M^* = \min \left\{ KQ_M, M - C_L^* - \widetilde{C}_L^* \right\}$ and $\widetilde{C}_M^* = \min \left\{ \frac{KQ_M - C_M^*}{\rho}, M - C_L^* - \widetilde{C}_L^* - C_M^* \right\}$.

REFERENCES

- [1] 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.
- [2] Juniper. (2017). *Virtual Reality Markets: Hardware, Content & Accessories 2017-2022*. [Online]. Available: <https://www.juniperresearch.com/researchstore/innovationdisruption/virtual-reality/hardware-content-accessories>
- [3] Q ABI Research. (Feb. 2017). *Augmented and Virtual Reality: The First Wave of 5G Killer APPS*. [Online]. Available: <https://www.qualcomm.com/documents/augmented-and-virtual-reality-first-wave-5g-killer-apps>

- [4] Huawei. (2019). *Cloud VR Solution White Paper*. [Online]. Available: <https://www.huawei.com/en/press-events/news/2018/9/cloud-vr-solutionwhite-paper>
- [5] D. Wang, X. Tian, H. Cui, and Z. Liu, "Reinforcement learning-based joint task offloading and migration schemes optimization in mobility-aware MEC network," *China Commun.*, vol. 17, no. 8, pp. 31–44, Aug. 2020.
- [6] Q. Li, X. Wang, D. Wang, Y. Zhang, Y. Lan, Q. Liu, and L. Song, "Analysis of an SDN-based cooperative caching network with heterogeneous contents," *Electronics*, vol. 8, no. 12, p. 1491, Dec. 2019.
- [7] E. Bastug, M. Bennis, M. Médard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 110–117, Jun. 2017.
- [8] Q. Li, X. Wang, and D. Wang, "Optimal D2D cooperative caching system in SDN based wireless network," in *Proc. IEEE 30th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2019, pp. 1–7.
- [9] Y. Cui, W. He, C. Ni, C. Guo, and Z. Liu, "Energy-efficient resource allocation for cache-assisted mobile edge computing," in *Proc. IEEE 42nd Conf. Local Comput. Netw. (LCN)*, Oct. 2017, pp. 640–648.
- [10] P. Maniotis and N. Thomos, "Tile-based edge caching for 360° live video streaming," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Feb. 1, 2021, doi: [10.1109/TCSVT.2021.3055985](https://doi.org/10.1109/TCSVT.2021.3055985).
- [11] V. R. Gaddam, M. Riegler, R. Eg, C. Griwodz, and P. Halvorsen, "Tiling in interactive panoramic video: Approaches and evaluation," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1819–1831, Sep. 2016.
- [12] T. El-Ganainy and M. Hefeeda, "Streaming virtual reality content," 2016, *arXiv:1612.08350*. [Online]. Available: <http://arxiv.org/abs/1612.08350>
- [13] Z. Fei, F. Wang, J. Wang, and X. Xie, "QoE evaluation methods for 360-degree VR video transmission," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 1, pp. 78–88, Jan. 2020.
- [14] H. Huang, B. Liu, L. Chen, W. Xiang, M. Hu, and Y. Tao, "D2D-assisted VR video pre-caching strategy," *IEEE Access*, vol. 6, pp. 61886–61895, 2018.
- [15] J. Dai, Z. Zhang, S. Mao, and D. Liu, "A view synthesis-based 360° VR caching system over MEC-enabled C-RAN," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3843–3855, Oct. 2020.
- [16] A. Ndikumana, S. Ullah, T. LeAnh, N. H. Tran, and C. S. Hong, "Collaborative cache allocation and computation offloading in mobile edge computing," in *Proc. 19th Asia-Pacific Netw. Oper. Manage. Symp. (APNOMS)*, Sep. 2017, pp. 366–369.
- [17] J. Chakareski, "VR/AR immersive communication: Caching, edge computing, and transmission trade-offs," in *Proc. Workshop Virtual Reality Augmented Reality Netw.*, Aug. 2017, pp. 36–41.
- [18] M. Erol-Kantarci and S. Sukhmani, "Caching and computing at the edge for mobile augmented reality and virtual reality (AR/VR) in 5G," *Ad Hoc Netw.*, vol. 223, pp. 169–177, Jan. 2018.
- [19] 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.
- [20] 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.
- [21] X. Yang, Z. Chen, K. Li, Y. Sun, N. Liu, W. Xie, and Y. Zhao, "Communication-constrained mobile edge computing systems for wireless virtual reality: Scheduling and tradeoff," *IEEE Access*, vol. 6, pp. 16665–16677, 2018.
- [22] T. Dang and M. Peng, "Joint radio communication, caching, and computing design for mobile virtual reality delivery in fog radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1594–1607, Jul. 2019.
- [23] J. Du, F. R. Yu, G. Lu, J. Wang, J. Jiang, and X. Chu, "MEC-assisted immersive VR video streaming over terahertz wireless networks: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9517–9529, Oct. 2020.
- [24] Y. Liu, J. Liu, A. Argyriou, and S. Ci, "MEC-assisted panoramic VR video streaming over millimeter wave mobile networks," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1302–1316, May 2019.
- [25] K. Long, Y. Cui, C. Ye, and Z. Liu, "Optimal wireless streaming of multi-quality 360 VR video by exploiting natural, relative smoothness-enabled and transcoding-enabled multicast opportunities," *IEEE Trans. Multimedia*, early access, Oct. 12, 2020, doi: [10.1109/TMM.2020.3029880](https://doi.org/10.1109/TMM.2020.3029880).
- [26] Q. Cheng, H. Shan, W. Zhuang, L. Yu, Z. Zhang, and T. Q. S. Quek, "Design and analysis of MEC- and proactive caching-based 360 mobile VR video streaming," *IEEE Trans. Multimedia*, early access, Mar. 19, 2021, doi: [10.1109/TMM.2021.3067205](https://doi.org/10.1109/TMM.2021.3067205).
- [27] L. Teng, G. Zhai, Y. Wu, X. Min, W. Zhang, Z. Ding, and C. Xiao, "QoE driven VR 360° video massive MIMO transmission," *IEEE Trans. Wireless Commun.*, to be published.
- [28] J.-T. Lee, D.-N. Yang, and W. Liao, "Efficient caching for multi-view 3D videos," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–7.
- [29] Y. Sun, Z. Chen, M. Tao, and H. Liu, "Communication, computing and caching for mobile VR delivery: Modeling and trade-off," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [30] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Power-delay tradeoff in multi-user mobile-edge computing systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

• • •