Matching Capabilities of Prediction to Communication and Computing for Proactive VR Video Streaming

Xing Wei and Chenyang Yang
School of Electronics and Information Engineering, Beihang University, Beijing 100191, China
Email: {weixing, cyyang}@buaa.edu.cn

Abstract—Proactive tile-based video streaming can avoid motion-to-photon latency of wireless virtual reality (VR) by computing and delivering the predicted tiles in a segment to be requested before playback. However, all existing works either focus on the tile prediction or focus on tile computing and delivering, overlooking the important fact that prediction, computing and communication have to share the same duration. Since the quality of experience (QoE) of proactive tile-based streaming depends on the worst performance of prediction, computing and communication, it is vital to match the prediction capability to the computing and communication capability. In this paper, we jointly optimize the duration of the observation window for tile prediction and the duration used for computing and communication, to maximize the QoE of watching a VR video. We find the global optimal solution with closed-form expression by decomposing the original problem equivalently into subproblems. From the optimal solution we find two regions where tile prediction and computing and communication capabilities respectively play the dominant role, and reveal the tradeoff between the performance of tile prediction and the capability of computing and communication. Simulation results using two existing tile prediction methods with a real dataset demonstrate the gain of the optimized duration over the non-optimized duration of observation window.

Index Terms—Wireless virtual reality, proactive tiled-based video streaming, duration optimization.

I. INTRODUCTION

Wireless virtual reality (VR) can provide immersive experience to wireless users. As the main type of VR services [1], VR video usually has $360^\circ \times 180^\circ$ panoramic view with ultra high resolution (e.g. 16 K [2]). Evidently, delivering such video is cost-prohibitive for wireless networks. Nonetheless, it has been found that the range of angles that human can see at the same time is only a limited area of the full panoramic view (about $110^\circ \times 90^\circ$ [3]), which is referred to as field of view (FoV). This inspires the design to divide the full panoramic view segment into small tiles in spatial domain, and only deliver the tiles that overlap with FoVs, which is the so-called tile-based streaming [4].

To avoid dizziness in watching VR video, the motion-to-photon (MTP) latency should be low (i.e., less than 20 ms [5]). When using reactive tile-based streaming, the tiles overlapped with the FoVs (on magnitude of hundreds of Mbits [5]) should be delivered within the MTP latency after the request of a user, which requires very high data rate. This calls for the proactive tile-based streaming [5] [6] [7], which computes (i.e., projects and renders the tiles to panoramic pictures [8]) and delivers the tiles in a segment that are most likely requested before the segment playback. In this way, the MTP latency can be avoided. Therefore, proactive tile-based streaming is anticipated to be the mainstream manner in the ultimate stage of VR streaming [5].

Proactive tile-based streaming contains two types of operations: prediction, and communication and computing. Existing works have studied these two operations individually. For tile prediction, a linear regression (LR) method was proposed in [7] to predict the central point of FoV, and transform it into corresponding tiles. A deep learning method was proposed in [9] that jointly uses head mounted display (HMD) orientations, image saliency maps, and motion maps to predict the tiles to be requested. A context bandits (CB) learning method was proposed in [6] to predict the tile requests implicitly. For communication and computing resource allocation, the rendering task is offloaded in [10] from HMD to mobile edge computing (MEC) server to reduce the bandwidth usage and computational workload on HMD. The computing and caching resources were leveraged in [11] to reduce communication resource usage. The multicast opportunity was exploited for multiple users watching the same video in [12] to minimize the transmission energy. However, all existing works ignore an important fact: tile prediction and communication and computing have to share the same duration.

Without optimization, longer duration may be allocated to an observation window used for tile prediction or to the communication and computing tasks. Then, one of the following cases may happen for the given resources: (i) The tiles to be requested can be predicted accurately, but there is no enough time remained to deliver and compute the predicted tiles before playback, or (ii) more tiles can be delivered and computed, but many delivered tiles are not requested due to the poor prediction. This suggests the necessity to match the performance of tile prediction and the capability of communication and computing.

In this paper, we aim to address this issue. To this end, we optimize the duration assigned for the two operations in proactive tile-based video streaming to maximize the quality of experience (QoE) of watching a VR video under resource constraint for arbitrarily given method of tile prediction. As far as we know, this is the first work that jointly optimizes the
duration for observation window, communication and computing in proactive tile-based VR streaming. We find the global optimal solution with closed-form expression via a serial of equivalent transformations and Karush-Kuhn-Tucker (KKT) conditions. The optimal solution yields important insights into matching the performance of tile prediction and the capability of communication and computing. We provide simulation results using existing prediction methods and a real dataset, which show the gain from jointly optimizing the duration.

II. SYSTEM MODEL

Consider a proactive tile-based VR video streaming system, where a MEC server with communication and computing resources serves multiple users. Each VR video is divided into \( L \) segments in temporal domain, and each segment is divided into \( M \) tiles in spatial domain. A segment has the same playback duration as a tile, denoted as \( T_{\text{seg}} \). The MEC server assigns orthogonal communication and computing resources to users, hence we can only consider one user. The user is equipped with a HMD to watch VR videos. The HMD can track the FoV of the user, send the recorded trace of FoV to the MEC server, and pre-buffer one segment\(^1\). While both the MEC server and the HMD can be used for projecting and rendering a video segment before playback, we consider that the MEC server processes the segment before delivering to the HMD. The basic idea can be extended to the case where the HMD accomplishes the computing tasks.

As shown in Fig. 1 by using the recorded FoV trace of a user in an observation window with duration \( t_{\text{obw}} \), the MEC server can predict the tiles in a segment to be most likely watched by the user\(^6\) for deciding which tiles in the segment should be proactively computed and delivered. Then, the MEC projects and renders the segment within duration \( t_{\text{cpt}} \), and transmits the processed tiles within duration \( t_{\text{com}} \) to the user. To ensure a segment able to be computed and delivered before playback, the total time should satisfy

\[
t_{\text{obw}} + t_{\text{com}} + t_{\text{cpt}} \leq T_{\text{ps}}
\]

where \( T_{\text{ps}} \) is the allowed proactive VR streaming time for a segment to avoid MTP latency\(^5\),\(^6\).

A. Performance Metric of Tile Prediction

Degree of overlap (DoO) has been used to measure the overlap between the predicted and requested frames of a panoramic video\(^7\). A larger value of DoO indicates a better prediction. To reflect the prediction performance for proactive tile-based streaming, we consider segment DoO, which measures the overlap of the predicted tiles and the requested tiles in a segment. The segment DoO of the \( l \)-th segment is defined as

\[
\text{DoO}_{l}^{\text{seg}}(t_{\text{obw}}) \triangleq \frac{q_{l}^{T} \cdot p_{l}^{\text{seg}}}{\|q_{l}\|_{1}}
\]

where \( q_{l} = [q_{l,1}, \ldots, q_{l,M}] \) denotes the ground-truth of the tile requests for the segment with \( q_{l,m} \in \{0, 1\} \), and \( p_{l}^{\text{seg}} = [p_{l,1}^{\text{seg}}, \ldots, p_{l,M}^{\text{seg}}] \) denotes the predicted tile requests for the segment with \( p_{l,m}^{\text{seg}} \in \{0, 1\} \), and \((\cdot)^{T}\) denotes transpose of a vector. When the \( m \)-th tile in the \( l \)-th segment is truly requested or predicted to be requested, \( q_{l,m} = 1 \) or \( p_{l,m}^{\text{seg}} = 1 \), otherwise they are zero.

We use average DoO to measure the prediction performance for all segments of a VR video, which is

\[
\text{DoO}^{\text{seg}}(t_{\text{obw}}) \triangleq \frac{1}{L} \sum_{l=1}^{L} \text{DoO}_{l}^{\text{seg}}(t_{\text{obw}}) = \sum_{n=0}^{\infty} a_{n} t_{\text{obw}}^{n}
\]

where the second equality is the power series expansion.

It can be observed from Fig. 1 that the prediction can be more accurate with longer observation window. This is because the predicted segment is closer to the observed FoV trace with a larger value of \( t_{\text{obw}} \). This gives rise to a reasonable assumption as follows.

**Assumption 1**: \( \text{DoO}^{\text{seg}}(t_{\text{obw}}) \) is a monotonous non-decreasing function of \( t_{\text{obw}} \).

B. Capability of Communication and Computing

To reflect the capability of the system for delivering and computing the predicted tiles in a segment, we introduce a notion called communication and computing (CC) capability, which is defined as,

\[
S_{\text{CC}}(t_{\text{com}}, t_{\text{cpt}}) \triangleq \min \left\{ S(t_{\text{com}}, t_{\text{cpt}}), S_{\text{max}} \right\} \in [0, 1]
\]

where \( S_{\text{max}} \) is the number of bits of all the predicted tiles in a segment, and

\[
S(t_{\text{com}}, t_{\text{cpt}}) \triangleq \min \{ C_{\text{com}} t_{\text{com}}, C_{\text{cpt}} t_{\text{cpt}} \}
\]

is the number of bits in the segment that can be delivered within time \( t_{\text{com}} \) and computed within time \( t_{\text{cpt}} \) using the communication resource \( C_{\text{com}} \) and computing resource \( C_{\text{cpt}} \) allocated to the user.

C. Quality of Experience (QoE)

For proactive tile-based streaming, there will be no MTP latency if (1) can be satisfied. Hence, the QoE will be improved if the prediction performance can be improved
the predicted tiles can be computed and delivered. Therefore, we can use the following metric to reflect the QoE,

$$\text{DoO}^{seg}(t_{\text{obw}}) \cdot S_{CC}^{*}(t_{\text{com}}, t_{\text{cpt}}) \in [0, 100\%]$$

When its value is 100%, all the requested tiles in a VR video are proactively computed and delivered before playback.

III. DURATION OPTIMIZATION FOR COMMUNICATION, COMPUTING, AND OBSERVATION WINDOW

In this section, we formulate a problem to jointly optimize the duration for communication, computing, and the observation window that maximizes the QoE of watching a VR video. We first decouple the problem into two subproblems, and obtain the global optimal solution with closed-form. Then, we show the impact of the allocated resources on the optimal duration, and analyze the relation between the prediction performance and the CC capability.

A. Problem Formulation and Decomposition

To maximize the QoE for proactive VR video streaming under communication and computing resource constraints, the problem that optimizes the duration of observation window $t_{\text{obw}}$, the duration for transmission $t_{\text{com}}$ and the duration for computing $t_{\text{cpt}}$ can be formulated as

$$\textbf{P0} : \max_{t_{\text{obw}}, t_{\text{com}}, t_{\text{cpt}}} \text{DoO}^{seg}(t_{\text{obw}}) \cdot S_{CC}(t_{\text{com}}, t_{\text{cpt}})$$

$$\text{s.t.} \quad S_{CC}(t_{\text{com}}, t_{\text{cpt}}) = \min\{S(t_{\text{com}}, t_{\text{cpt}}), S_{\text{max}}\}$$

$$\begin{align*}
S(t_{\text{com}}, t_{\text{cpt}}) &= \min\{S_{\text{com}}, t_{\text{com}}, C_{\text{cpt}} t_{\text{cpt}}\} \\
t_{\text{obw}} + t_{\text{com}} + t_{\text{cpt}} &\leq T_{pS} \\
t_{\text{obw}} &\geq \tau, \quad t_{\text{com}}, t_{\text{cpt}} \geq 0
\end{align*}$$

where $\tau$ is the minimal duration of the observation window.

Considering the separability of the objective function and the constraints in $\textbf{P0}$, we can decompose the problem into two subproblems without loss of optimality. To this end, we introduce a variable $t_{CC}$ as a constraint imposed on the communication and computing duration, i.e.,

$$t_{\text{com}} + t_{\text{cpt}} \leq t_{CC}$$

such that the constraint in $[4]$ can be decomposed into $[4]$ and a constraint $t_{\text{obw}} + t_{CC} \leq T_{pS}$. Then, from $[3a]$-$[3d]$, the variables $t_{\text{com}}$ and $t_{\text{cpt}}$ can be regarded as the functions of $t_{CC}$. To find the optimal solution of $\textbf{P0}$, we can first optimize these two variables with the following problem

$$\textbf{P1} : \max_{t_{\text{com}}, t_{\text{cpt}}(t_{CC}), t_{\text{obw}}} S_{CC}(t_{CC})$$

$$\text{s.t.} \quad [3a] [3c] [4]$$

and then optimize $t_{\text{obw}}$ from the following problem

$$\textbf{P2} : \max_{t_{\text{obw}}, t_{CC}} \text{DoO}^{seg}(t_{\text{obw}}) \cdot S_{CC}^{*}(t_{CC})$$

$$\text{s.t.} \quad t_{\text{obw}} + t_{CC} \leq T_{pS}$$

$$t_{\text{obw}} \geq \tau$$

where $S_{CC}(t_{CC}) \triangleq S_{CC}(t_{\text{com}}, t_{CC}, t_{\text{cpt}}(t_{CC}))$, and $S_{CC}^{*}(t_{CC})$ is the maximized value of the objective function of $\textbf{P1}$ after optimizing the problem.

B. Solution of Problem $\textbf{P1}$

By introducing an auxiliary variable $p(t_{CC})$, problem $\textbf{P1}$ can be equivalently transformed into the following problem

$$\begin{align*}
\max_{t_{\text{com}}(t_{CC}), t_{\text{cpt}}(t_{CC}), p(t_{CC})} & \quad S_{CC}(t_{CC}) \\
\text{s.t.} & \quad S_{CC}(t_{CC}) \leq p(t_{CC}) \leq S_{\text{max}} \\
& \quad p(t_{CC}) \leq S(t_{CC}) \leq S_{\text{max}} \\
& \quad S(t_{CC}) \leq C_{\text{com}} t_{\text{com}} \\
& \quad S(t_{CC}) \leq C_{\text{cpt}} t_{\text{cpt}}
\end{align*}$$

$$\begin{align*}
\big(7a\big) & \quad \big(7b\big) & \quad \big(7c\big) & \quad \big(7d\big) & \quad \big(7e\big) & \quad \big(7f\big) & \quad \big(7g\big)
\end{align*}$$

where $p(t_{CC}) \triangleq \min\{S(t_{CC}), S_{\text{max}}\}$, and $\bar{S}(t_{CC}) \triangleq S(t_{\text{com}}(t_{CC}), t_{\text{cpt}}(t_{CC}))$ (noting that it differs from $S_{CC}(t_{CC})$ defined in $\textbf{P2}$).

This is a convex problem. After some regular derivations, its optimal solution can be found from its KKT conditions as

$$\begin{align*}
t^{*}_{\text{com}}(t_{CC}) &= \begin{cases} C_{\text{com}} + C_{\text{cpt}} t_{\text{com}}, & t_{CC} \leq T_{\text{max}}^{CC} \\
\alpha, & t_{CC} > T_{\text{max}}^{CC} \end{cases} \\
t^{*}_{\text{cpt}}(t_{CC}) &= \begin{cases} C_{\text{com}}, & t_{CC} \leq T_{\text{max}}^{CC} \\
\beta, & t_{CC} > T_{\text{max}}^{CC} \end{cases} \\
p^{*}(t_{CC}) &= \begin{cases} C_{\text{com}} + C_{\text{cpt}} t_{\text{com}}, & t_{CC} \leq T_{\text{max}}^{CC} \\
S_{\text{max}}, & t_{CC} > T_{\text{max}}^{CC} \end{cases}
\end{align*}$$

$$\begin{align*}
\big(8a\big) & \quad \big(8b\big) & \quad \big(8c\big)
\end{align*}$$

where $\alpha + \beta \equiv T_{\text{max}}^{CC}$, and

$$T_{\text{max}}^{CC} \triangleq \frac{S_{\text{max}}}{C_{\text{com}}} + \frac{S_{\text{max}}}{C_{\text{cpt}}}$$

is the required duration to deliver and compute all the predicted tiles. Upon substituting the optimal solution, the maximized value of the objective function becomes

$$S_{CC}(t_{CC}) \triangleq \begin{cases} \frac{C_{\text{com}} C_{\text{cpt}}}{(C_{\text{com}} + C_{\text{cpt}}) S_{\text{max}}}, & t_{CC} \leq T_{\text{max}}^{CC} \\
1, & t_{CC} > T_{\text{max}}^{CC} \end{cases}$$

$$\big(9\big)$$

When $t_{CC} = T_{\text{max}}^{CC}$, we can find from the case of $t_{CC} \leq T_{\text{max}}^{CC}$ that $S_{CC}(t_{CC}) = 1$, which is exactly the same as the case of $t_{CC} > T_{\text{max}}^{CC}$. Therefore, the case of $t_{CC} > T_{\text{max}}^{CC}$ in $[9]$ can be omitted.

C. Solution of Problem $\textbf{P2}$

Upon substituting the solution under $t_{CC} \leq T_{\text{max}}^{CC}$ in $[9]$, problem $\textbf{P2}$ can be re-written as

$$\begin{align*}
\max_{t_{\text{obw}}, t_{CC}} & \quad \text{DoO}^{seg}(t_{\text{obw}}) \cdot \frac{C_{\text{com}} C_{\text{cpt}} t_{\text{cc}}}{(C_{\text{com}} + C_{\text{cpt}}) S_{\text{max}}} \\
\text{s.t.} & \quad t_{\text{obw}} + t_{CC} \leq T_{pS} \\
& \quad t_{CC} \leq T_{\text{max}}^{CC} \\
& \quad t_{\text{obw}} \geq \tau, t_{CC} \geq 0
\end{align*}$$

$$\begin{align*}
\big(11a\big) & \quad \big(11b\big) & \quad \big(11c\big) & \quad \big(11d\big)
\end{align*}$$
Since \( \frac{C_{\text{com}} C_{\text{cpt}}}{(C_{\text{com}}+C_{\text{cpt}}) S_{\text{max}}} > 0 \), the objective function in (11a) is a monotonous non-decreasing function of \( t_{\text{obs}} \) and \( t_{\text{CC}} \) under Assumption 1. Therefore, the optimal solution is achieved when \( t_{\text{obs}} \) and \( t_{\text{CC}} = T_{\text{ps}} \). Upon substituting \( t_{\text{obs}} = T_{\text{ps}} - t_{\text{CC}} \), problem (11) is degenerated as

\[
\max_{t_{\text{cc}}} \text{DoO}^{\text{seg}}(T_{\text{ps}} - t_{\text{CC}}) \quad \frac{C_{\text{com}} C_{\text{cpt}} t_{\text{CC}}}{(C_{\text{com}} + C_{\text{cpt}}) S_{\text{max}}} \tag{12a}
\]

subject to

\[
t_{\text{CC}} \leq T_{\text{CC}} \tag{12b}
\]

\[
T_{\text{ps}} - t_{\text{CC}} \geq \tau \tag{12c}
\]

Further considering (2), the optimal solution of the problem in (12) can be derived from its KKT conditions as

\[
t_{\text{obs}}^* = \begin{cases} 
H, & \phi(t_{\text{obs}}) > 0 \quad \text{no feasible solution, otherwise} \\
\phi^{-1}(0), & \phi(t_{\text{obs}}) = 0, \text{ and } t_{\text{obs}} \geq H 
\end{cases} \tag{13a}
\]

\[
t_{\text{CC}}^* = \begin{cases} 
G, & \phi(t_{\text{obs}}) > 0 \quad \text{no feasible solution, otherwise} \\
W, & \phi(t_{\text{obs}}) = 0, \text{ and } t_{\text{CC}} \leq W 
\end{cases} \tag{13b}
\]

where \( H \triangleq \max[T_{\text{ps}} - T_{\text{CC}}^\text{max}, \tau] \), \( W \triangleq \min(T_{\text{CC}}^\text{max}, T_{\text{ps}} - \tau) \), \( \phi(t_{\text{obs}}) = \sum_{n=1}^{\infty} a_n t_{\text{obs}}^n - (\sum_{n=1}^{\infty} a_n t_{\text{obs}}^{n-1}) (T_{\text{ps}} - t_{\text{obs}}) \), \( \phi^{-1}(\cdot) \) is the inverse function of \( \phi(\cdot) \), and \( G \triangleq T_{\text{ps}} - \phi^{-1}(0) \).

By substituting (13b) into (8a) and (8b), we obtain the optimal duration for computing and communications as

\[
t_{\text{com}}^* = \begin{cases} 
\frac{C_{\text{com}}}{C_{\text{com}} + C_{\text{cpt}}} W, & \phi(t_{\text{obs}}) > 0 \quad \text{no feasible solution, otherwise} \\
\frac{C_{\text{com}}}{C_{\text{com}} + C_{\text{cpt}}} G, & \phi(t_{\text{obs}}) = 0, \text{ and } t_{\text{CC}} \leq W 
\end{cases} \tag{14a}
\]

\[
t_{\text{cpt}}^* = \begin{cases} 
\frac{C_{\text{com}}}{C_{\text{com}} + C_{\text{cpt}}} W, & \phi(t_{\text{obs}}) > 0 \quad \text{no feasible solution, otherwise} \\
\frac{C_{\text{com}}}{C_{\text{com}} + C_{\text{cpt}}} G, & \phi(t_{\text{obs}}) = 0, \text{ and } t_{\text{CC}} \leq W 
\end{cases} \tag{14b}
\]

### D. Prediction Performance versus and CC Capability

In Fig. 2, we show the tradeoff between the prediction performance and the CC capability. Considering that increasing the value of \( \frac{1}{T_{\text{CC}}} \) in (9) can reflect the increase of either communication or computing resource, we provide the values of \( t_{\text{obs}}^* \) and \( t_{\text{CC}}^* \) versus \( \frac{1}{T_{\text{CC}}} \), where \( \tau = 0.1 \) s, and \( T_{\text{ps}} = 1 \) s. The optimal duration are obtained from (13a) and (13b) under the case of \( \phi(t_{\text{obs}}) > 0 \). As verified with two prediction methods and a real dataset to be introduced in section IV, this case always occurs.

We can observe that both the relations \( t_{\text{obs}}^* - \frac{1}{T_{\text{CC}}} \) and \( t_{\text{CC}}^* - \frac{1}{T_{\text{CC}}} \) can be divided into two regions, and both boundary lines in Fig. (a) and Fig. (b) between the two regions are \( \frac{1}{T_{\text{CC}}} = t_{\text{obs}}^* \). When the value of \( \frac{1}{T_{\text{CC}}} \) is small, i.e., at least one type of resources is limited, with which not all the predicted tiles can be delivered or computed, the bottleneck of proactive VR streaming is the resources. To increase the number of bits able to be computed or delivered, all remaining time should be used for computing or delivering after satisfying the minimal duration required by the observation window, i.e., \( t_{\text{obs}} = \tau, t_{\text{CC}}^* = T_{\text{ps}} - \tau \). We call this region as “Resource-limited region”. When \( \frac{1}{T_{\text{CC}}} > \frac{1}{T_{\text{ps}} - \tau} \), the time allocated to communication or computing tasks is equal to the required time for delivering and computing all the predicted tiles. As the value of \( \frac{1}{T_{\text{CC}}} \) further increases, all the predicted tiles can be delivered and computed, then the prediction performance becomes the bottleneck of proactive streaming. According to Assumption 1, DoO\( ^{\text{seg}}(t_{\text{obs}}) \) can be improved by increasing \( t_{\text{obs}} \). Hence, the optimal solution should allocate longer duration for the observation window, i.e., \( t_{\text{CC}}^* = T_{\text{ps}} - T_{\text{CC}}^\text{max} \) and \( t_{\text{obs}}^* = T_{\text{ps}} - \frac{T_{\text{CC}}^\text{max}}{2} \). The value of \( t_{\text{obs}}^* \) decreases and the value of \( t_{\text{obs}}^* \) increases with \( \frac{1}{T_{\text{CC}}} \). We call this region with \( \frac{1}{T_{\text{CC}}} > \frac{1}{T_{\text{ps}} - \tau} \) as “Prediction-limited region”.

In Fig. 3, we show the tradeoff between the prediction performance and the CC capability. We set \( T_{\text{ps}}^\text{max} \) as \( \{0.1, 0.2, \ldots, 9.9, 10\} \) s. With each value of \( T_{\text{ps}}^\text{max} \), we compute the values of \( t_{\text{obs}}^*, t_{\text{CC}}^*, \) and \( S_{\text{CC}}^* \) with (13a), (13b) and (10) under the case of \( \phi(t_{\text{obs}}) > 0 \), respectively. Then, for each value of \( S_{\text{CC}}^* \), we obtain the required prediction performance to achieve the given QoE from (3a) by dividing \( S_{\text{CC}}^* \). To help understand the tradeoff between the prediction performance and the CC capability, we take the average QoE of 90% as an example, where the corresponding values of \( t_{\text{obs}}^* \) and \( t_{\text{CC}}^* \) are illustrated.
and $t_{CC}$ are provided in the figure. To achieve the same QoE, increasing the CC capability from 90% to 100% can reduce the required prediction performance from 100% to 90%.

IV. SIMULATION RESULTS

In this section, we use simulations to verify Assumption I with two prediction methods, linear regression (LR) and context bandits (CB), and evaluate the performance of the optimal solution.

Again, we set $\tau = 0.1$ s, and $T_{ps} = 1$ s.

A. $\text{DoO}^\text{seg}$ Functions of LR and CB Methods

The method proposed in [7] employs a sequence of traces for the central points of FoVs to train a LR model for predicting the central point of the FoV. Then, the predicted tiles can be obtained from the predicted central point of FoV via projecting [6]. This LR method performs similarly to the update-to-date deep learning methods [13] [15] for tile prediction. The CB method proposed in [6] predicts the tiles to be requested implicitly, by using the tile requests in an observation window as contexts and setting the tile requests in the next segment as arms.

We consider a real dataset [15], which contains 500 traces of tile requests from $K = 50$ users watching 10 VR videos. For each video, $K \approx 50$ traces can be used to compute the ground-truth of $\text{DoO}^\text{seg}(t_{obw})$. By running each prediction method on the $k$-th trace, $\text{DoO}^\text{seg,k}(t_{obw})$ can be obtained under different duration of observation window. Then, for each video, the ground-truth of $\text{DoO}^\text{seg}(t_{obw})$ can be obtained as $\text{DoO}^\text{seg}(t_{obw}) = \frac{1}{K} \sum_{k=1}^{K} \text{DoO}^\text{seg,k}(t_{obw})$ for fitting. For LR method, the fitted function is

$$\text{DoO}^\text{seg,LR}(t_{obw}) = \sum_{n=0}^{4} a_n t_{obw}^n$$

(15)

For CB method, the fitted function is

$$\text{DoO}^\text{seg,CB}(t_{obw}) = a_1 t_{obw} + a_0$$

(16)

To evaluate the fitting performance, we use mean square error (MSE) as the metric that is widely used in regression analysis, which is

$$\text{MSE} = \frac{1}{D} \sum_{d=1}^{D} \left( \text{DoO}^\text{seg}(t_{obw}^d) - \text{DoO}^\text{seg}(\hat{t}_{obw}^d) \right)^2$$

(17)

where $D$ is the number of values of $t_{obw}$, $t_{obw}^d$ is the $d$-th value of $t_{obw}$, and $\text{DoO}^\text{seg}(t_{obw}^d)$ is the fitted value of $\text{DoO}^\text{seg}(t_{obw}^d)$.

The corresponding fitting parameters for each video and the fitting performance are provided in Tables I and II, where $D = 28$.

In Fig. 4 we compare the fitting function and the ground-truth of $\text{DoO}^\text{seg}(t_{obw})$. For each method, we choose two curves, where one has the best and the other has the worst fitting performance, which are highlighted in Tables I and II respectively. We can see that both the best and worst fitting functions fit well.

B. Performance of the Proposed Solution

We use the average QoE taken over the 10 videos as the metric to evaluate the performance of the optimal solution, which is obtained from (13a) and (13b) and with legend “Optimal time allocation”.

Since there are no related works of tile-based streaming on jointly optimizing the duration of observation window and for computing and delivering, we employ two schemes without optimizing $T_{obw}$ for comparison. In the first scheme, $T_{CC} = T_{obw} = \frac{T}{3}$, which is with legend “Equal time allocation”. In the second scheme, $T_{obw} = T_{ps} = \frac{T}{3}$, $T_{CC} = 2 \times \frac{T}{3}$, i.e., more time is allocated for computing and communication, which is with legend “$1 \frac{1}{3}$: $\frac{2}{3}$ time allocation”. Both schemes employ $t^*_{obw}$ and $t^*_{CC}$ for the given value of $t_{CC}$, which are obtained from (3a) and (3b).

The results are provided in Fig. 5. As expected, the optimal solution yields the best performance. In addition, we can observe a resource-limited region, a prediction-limited

| Video name | Fitting parameters | MSE    |
|------------|-------------------|--------|
| coaster    | $a_0$             | $a_1$  | $a_2$  | $a_3$  | $\text{MSE}$ |
|            | 0.7101            | 0.1394 | -0.1609| 0.0717 | $9.6782 \times 10^{-7}$ |
| coaster2   | 0.7274            | 0.1686 | -0.2354| 0.1215 | $7.3758 \times 10^{-7}$ |
| diving     | 0.5773            | 0.1462 | -0.1955| 0.1015 | $7.2133 \times 10^{-7}$ |
| drive      | 0.7305            | 0.0759 | -0.0879| 0.0392 | $2.9301 \times 10^{-7}$ |
| game       | 0.7276            | 0.2081 | -0.2722| 0.1192 | $9.6005 \times 10^{-7}$ |
| landscape  | 0.7155            | 0.0588 | -0.0990| -0.0100| $7.0822 \times 10^{-7}$ |
| pacman     | 0.7813            | 0.1030 | -0.1231| 0.0627 | $1.1475 \times 10^{-6}$ |
| panel      | 0.6883            | 0.0423 | -0.0484| 0.0178 | $9.9769 \times 10^{-7}$ |
| ride       | 0.7404            | 0.1499 | -0.1802| 0.0837 | $7.3371 \times 10^{-7}$ |
| sport      | 0.7309            | 0.1060 | -0.0816| 0.0210 | $1.4887 \times 10^{-6}$ |

| Video name | Fitting parameters | MSE    |
|------------|-------------------|--------|
| coaster    | $a_0$             | $a_1$  | $a_2$  | $a_3$  | $\text{MSE}$ |
|            | 0.7702            | 0.1242 | 4.7013 | $10^{-6}$ |
| coaster2   | 0.7665            | 0.1277 | 4.5542 | $10^{-6}$ |
| diving     | 0.6993            | 0.1669 | 6.6648 | $10^{-6}$ |
| drive      | 0.7211            | 0.1544 | 8.1762 | $10^{-6}$ |
| game       | 0.6718            | 0.1833 | 7.5323 | $10^{-6}$ |
| landscape  | 0.6680            | 0.1793 | 9.1398 | $10^{-6}$ |
| pacman     | 0.7638            | 0.1343 | 5.2914 | $10^{-6}$ |
| panel      | 0.6565            | 0.1863 | 9.0814 | $10^{-6}$ |
| ride       | 0.7183            | 0.1562 | 7.1564 | $10^{-6}$ |
| sport      | 0.7008            | 0.1648 | 8.4279 | $10^{-6}$ |

![Fig. 4: Fitting performance.](image-url)
region and a boundary line similar to Fig. 2 where the curves in the two regions have different slopes. The curve in the resource-limited region has a larger slope, since the computing and communication resources are the bottleneck and hence increasing resources can improve the QoE. The curve in the prediction-limited region has a smaller slope, where the increase of resources only plays a role of increasing $t_{obw} = T_{ps} - T_{CC}$ by reducing the duration for computing and communication $T_{CC}$. When the observation window becomes longer, the prediction performs better, i.e., the value of $D_{obw}$ increases, and hence the QoE is improved but slowly with the increased resources. The gain of the proposed solution over the two baselines for comparison is the largest on the boundary line, since in this case the prediction capability perfectly matches the CC capability, i.e., the QoE achieved by the prediction performance is the same as the QoE that can be achieved by the computation and communication resources. The results are similar if we choose LR for tile prediction, which are not shown for conciseness.

To show how much communication and computing resources are required for matching the prediction capability, we consider the following example. Each segment of the videos in the dataset is divided into $M = 200$ tiles in 10 rows and 20 columns [16], each tile has the resolution of $640 \times 540$ pixels and is with bitrate $r = 3$ Mbps and duration $T_{seg} = 1$ s [17]. Hence, the number of bits in each tile is $s = r \times T_{seg} = 3$ Mbits. The maximal number of predicted tiles in a segment is $N = 80$, thus the number of bits of all predicted tiles in a segment $S_{max} = N \times s = 240$ Mbits. The value of $T_{CC}$ in the boundary line is $T_{ps} - \tau = 0.9$ s. Assume that $C_{com} = C_{cpt}$. By substituting the values of $T_{max}$, $T_{CC}$ and $S_{max}$ into (9), we obtain that $C_{com} = C_{cpt} = 533.3$ Mbps required in the boundary. If $T_{CC} = 5$, we can obtain that $C_{com} = C_{cpt} = 2.4$ Gbps. Further considering Fig. 5 we can see that significantly more resources are required in order to provide the marginal gain in terms of improving the QoE in the prediction-limited region.

V. CONCLUSION

In this paper, we optimized the duration of the observation window, computing and delivering in proactive tile-based VR streaming and obtained the global optimal solution. By using the optimal solution, we found the prediction-limited and resource-limited region, and revealed the tradeoff between the performance of tile prediction and the capability for computing and delivering the predicted tiles. We employed simulations with two existing tile prediction methods and a real dataset to validate our analysis, and evaluate the performance of the proposed optimal solution. Our results demonstrated the benefit of matching the capability of tile prediction to the capability of computing and communication. In the resource-limited region, the achieved QoE is low. In the prediction-limited region, the QoE can be further improved but at the cost of increasing the resources considerably.

REFERENCES

[1] Virtual Reality (VR) Media Services Over 3GPP, 3GPP Std. document TR 26.918, Mar. 2018.
[2] E. Bastug, M. Bennis, M. Medard, and M. Debbah, “Toward interconnected virtual reality: Opportunities, challenges, and enablers,” IEEE Commun. Mag., vol. 55, no. 6, pp. 110–117, June 2017.
[3] VR Lens Lab, “Field of view for virtual reality headsets explained,” https://vr-lens-lab.com/field-of-view-for-virtual-reality-headsets/.
[4] M. Hosseini and V. Swammathan, “Adaptive 360 VR video streaming: Divide and conquer,” IEEE ISM, 2016.
[5] Huawei, “Whitepaper on the VR-oriented bearer network requirement,” Huawei Technologies CO., LTD, Tech. Rep., 2016. [Online]. Available: https://www.huawei.com/en/industry-insights/technology/white-papers/whitepaper-on-the-vr-oriented-bearer-network-requirement.
[6] W. Xing and C. Yang, “Tile-based proactive virtual reality streaming via online hierarchical learning,” APCC, to appear, 2019.
[7] F. Qian, L. Ji, B. Han, and V. Gopalakrishnan, “Optimizing 360 video delivery over cellular networks,” ACM SIGCOMM Workshop, 2015.
[8] S. Mangiante, G. Klas, A. Navon, Z. Guanhua, J. Ran, and M. D. Silva, “VR is on the edge: How to deliver 360° videos in mobile networks,” ACM SIGCOMM, 2017.
[9] C.-L. Fan, J. Lee, W.-C. Lo, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu, “Fixation prediction for 360 video streaming in head-mounted virtual reality,” ACM NOSSDAV, 2017.
[10] W. Lo, C. Huang, and C. Hsu, “Edge-assisted rendering of 360 videos streamed to head-mounted virtual reality,” IEEE ISM, 2018.
[11] X. Yang, Z. Chen, K. Li, Y. Sun, and H. Zheng, “Optimal task scheduling in communication-constrained mobile edge computing systems for wireless virtual reality,” APCC, 2017.
[12] C. Guo, Y. Cui, and Z. Liu, “Optimal multicast of tiled 360 VR video in OFDMA systems,” IEEE Commun. Lett., vol. 22, no. 12, pp. 2563–2566, Dec. 2018.
[13] T. C. Nguyen and J. Yun, “Predictive tile selection for 360-degree VR video streaming in bandwidth-limited networks,” IEEE Commun. Lett., vol. 22, no. 9, pp. 1858–1861, Sep. 2018.
[14] C. Perfecto, M. S. ElBamby, J. D. Ser, and M. Bennis. (2018) Taming the latency in multi-user VR 360°: A QoE-aware deep learning-aided multicast framework. arXiv. [Online]. Available: [http://arxiv.org/abs/1811.07388].
[15] M. Xu, Y. Song, J. Wang, M. Qiao, L. Huo, and Z. Wang, “Predicting head movement in panoramic video: A deep reinforcement learning approach,” IEEE Trans. Pattern Anal. Machine Intell., vol. 41, no. 11, pp. 2693–2708, Nov 2019.
[16] W.-C. Lo, C.-L. Fan, J. Lee, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu, “360° video viewing dataset in head-mounted virtual reality,” ACM MM’Sy, 2017.
[17] A. Mahzari, A. T. Nasrabadi, A. Samie, and R. Prakash, “Few-vu-aware edge caching for adaptive 360 video streaming,” ACM MM, 2018.