Online Learning-Based Rate Selection for Wireless Interactive Panoramic Scene Delivery

Harsh Gupta* Jiangong Chen† Bin Li† R. Srikant*
*Department of ECE, University of Illinois at Urbana-Champaign, Urbana, IL, USA
†Department of EE, The Pennsylvania State University, University Park, PA, USA

IEEE INFOCOM 2022 - IEEE Conference on Computer Communications | 978-1-6654-5822-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/INFOCOM48880.2022.9796965

***Abstract*—Interactive panoramic scene delivery not only consumes $4 \sim 6\times$ more bandwidth than traditional video streaming of the same resolution but also requires timely displaying the delivered content to ensure smooth interaction. Since users can only see roughly 20% of the entire scene at a time (called the viewport), it is sufficient to deliver the relevant portion of the panoramic scene if we can accurately predict the user's motion. It is customary to deliver a portion larger than the viewport to tolerate inaccurate predictions. Intuitively, the larger the delivered portion, the higher the prediction accuracy and lower the wireless transmission success probability. The goal is to select an appropriate delivery portion to maximize system throughput. We formulate this problem as a multi-armed bandit problem and use the classical Kullback-Leibler Upper Confidence Bound (KL-UCB) algorithm for the portion selection. We further develop a novel variant of the KL-UCB algorithm that effectively leverages two-level feedback (i.e., both prediction and transmission outcomes) after each decision on the selected portion and show its asymptotical optimality, which may be of independent interest by itself. We demonstrate the superior performance of our proposed algorithms over existing heuristic methods using both synthetic simulations and real experimental evaluations.**

I. Introduction

Panoramic videos and virtual reality (VR) provide an interactive and immersive experience in a virtual 3D world and has received great attention from both academia and different industries in recent years. One major challenge in high-resolution panoramic video streaming and virtual reality is that they demand $4 \sim 6\times$ the bandwidth compared to a regular video with the same resolution (see [1]). However, a user may just need to see roughly 20% of the entire panoramic scene without affecting her/his visual perception depending on their perspective. This small and relevant portion of the entire panoramic scene is known as the user's viewport. For instance, in the case of a panoramic roller coaster video, a user can see either the front views or back views at any given time. Therefore, if a user's motion is accurately predicted, it is sufficient to deliver just 20% of the $360°$ scenes surrounding them, thereby significantly reducing network bandwidth consumption.

Unfortunately, it is impossible to achieve zero error in predicting a user's motion. As a result, a portion of the panoramic scene larger than the viewport is usually delivered. As long as the delivered portion covers the user's viewport, the user will successfully view the content. Although a larger delivery portion can tolerate a larger prediction error and thus yield a higher probability of prediction (viewport coverage), it can result in transmission failure since a larger portion of the panoramic scene may exceed the maximum transmission rate at the current wireless channel state. Noting that the interactive panoramic scene delivery requires timely displaying the delivered content, a central question is how to select an appropriate delivery portion at each time to maximize system throughput or some other metric of the user's quality of experience at the same time. Note that increasing or decreasing the delivery portion increases or decreases, respectively, the transmission data rate over the wireless channel. However, this rate selection problem is quite different from traditional rate selection problems in wireless networks where the main goal of rate selection is to adapt to the quality of the wireless channel [2], [3]; in particular, there are no viewport prediction considerations in earlier works on rate selection.

Recent works (e.g., [1], [4], [5], [6], [7], [8], [9]) have explored various efficient user motion prediction algorithms and have incorporated them to reduce the wireless bandwidth requirement of panoramic scene delivery. These papers typically require collecting head motion traces from many users for different video content, and subsequently train a motion prediction model based on the collected data. However, for determining the corresponding delivery portion, they use heuristic methods. Moreover, they do not explore fast-changing wireless environments. To this end, in this paper, we formulate the delivery portion selection problem as a stochastic multi-armed bandit (MAB) problem, where different delivery portions of the panoramic scene correspond to different arms and the goal is to minimize the regret (i.e., the gap between the optimal cumulative throughput and the cumulative throughput under an algorithm) over a finite time horizon. The considered setup has two-level feedback, i.e., after each arm is played, we receive both the prediction and transmission outcomes and the reward is determined by the product of these two independent pieces of information.

The MAB problem is well-studied and has a wide array of practical applications (see [10] for reference). In the traditional stochastic MAB setting, at each time slot, an agent plays an arm (from a set of arms) and receives a random reward drawn (independently across time) from the reward distribution of the arm it played. The goal of the agent is to minimize its regret over a certain time horizon (defined as the loss in expected

This research has been supported in part by NSF grants: CCF 1704970, CNS 2106801, CNS 2152658, CNS 2152610; ONR N00014-19-1-2566, and ARO W911NF-19-1-0379.

The authors have provided public access to their code or data at https://codeocean.com/capsule/9322311/tree/v1.

978-1-6654-5822-1/22/$31.00 ©2022 IEEE

reward incurred by the agent as compared to an oracle which knows the optimal arm) by taking sequential decisions which strike a delicate balance between exploiting the information that the agent already has and exploring different arms in order to gain more information. The fundamental difference between our problem and the standard stochastic MAB problem is that our reward is determined by the product of two independent pieces of random information instead of simply one net reward feedback. Although we can consider the product of the two levels of feedback as the net reward and formulate the problem as the standard stochastic MAB problem, we wish to exploit the potentially higher level of information that the two independent levels of feedback can provide compared to their product.

In their seminal work [11], the authors proved a fundamental lower bound on the regret that can be achieved by any uniformly good algorithm for the traditional stochastic MAB problem. Since then, a number of popular and easy to implement algorithms have been designed which asymptotically achieve this fundamental lower bound (e.g., KL-UCB [12] and Thompson sampling [13]). Some recent works (e.g., [14], [15]) have considered the bandit problem with multiple-level feedback; however, their model is very different and is motivated by advertising applications where a user might click on an ad and then purchase the product advertised in the ad. In the ad model, one gets feedback about whether the user purchased the product or not only if they click the ad. In our problem, we get two pieces of information, one about the wireless transmission and one about the prediction, and both are available in each time slot. To the best of our knowledge, the ad model results cannot be used in our context. Our main contributions in this paper are as follows:

1) We formulate the problem of maximizing the system throughput in panoramic video delivery as a stochastic multi-armed bandit (MAB) problem with two-level feedback. This non-standard formulation of the stochastic MAB problem can be more generally useful in other MAB application domains where two levels of feedback are available (see Section II).
2) We develop a novel variant of KL-UCB algorithm to efficiently solve the stochastic MAB problem with two-level feedback. We show that this algorithm asymptotically minimizes the regret in the sense that the upper bound on the regret of this algorithm asymptotically matches the lower bound (see Section III). This analysis is non-trivial due to the two-level feedback and is presented in Section V.
3) In order to establish the practical utility of our algorithm, we conduct both synthetic simulation and real experimental evaluations for both panoramic video and VR applications. We conclusively establish that our bandit algorithms outperform existing heuristic methods and the KL-UCB algorithm with two-level feedback further improves the system performance (see Section IV).

II. Problem Formulation

We consider a single user with VR headset exploring interactive and panoramic scenes that are delivered from an access point (AP) over a wireless channel, as shown in Fig. 1. The user can rotate its head in three different axes (yaw, pitch, roll) for watching interactive panoramic videos or freely explore panoramic VR scenes with 6-Degrees-of-Freedom (DoF) (3 DoF for user's virtual position and the other 3 DoF for user's head orientation). We assume that there is no playback buffer at user's device to ensure timely and smooth interactions. We also assume that the system operates in slotted time with normalized time slots $t \in \{1, 2, \ldots, n\}$. For example, the time slot duration is set to 33ms and 16ms for panoramic videos and VR applications, respectively. This guarantees the desired quality of experience for panoramic videos and VR applications that require at least 30 and 60 frames-per-second (FPS), respectively.

In each time slot t, only a portion (typically 20%) of the whole panoramic scene can be seen by a user, namely the *viewport*. If we can accurately predict a user's head movement, then it is sufficient to deliver just 20% of the whole scenes, which consumes only $1/5$ of the originally required wireless bandwidth. Unfortunately, a user typically randomly moves his/her head depending on his/her interest in the panoramic content. Hence, it is unavoidable to incur errors in head motion prediction. To mitigate this, we usually deliver a portion of the panoramic scene larger than the viewport.

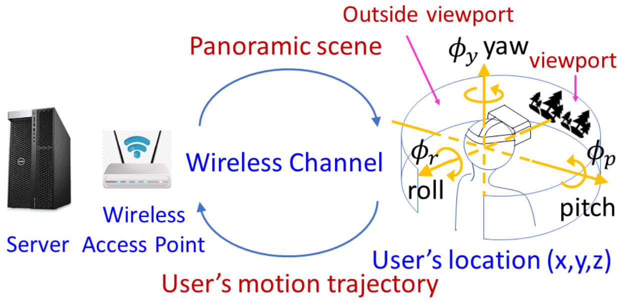


Fig. 1: Wireless panoramic scene delivery.

We note that each panoramic scene can be split into a fixed number of tiles (see Fig. 2) and a tile is the minimum delivery unit required for image encoding and decoding. In each time slot, a subset of tiles can be selected for transmission. Since there are a finite number of subsets of tiles in each panoramic scene, we assume that there are K different rates corresponding to different portions of the panoramic scene: $0 < r_1 < r_2 < \cdots < r_K$, where r_1 refers to the rate when only the predicted viewport is selected for transmission and r_K refers to the rate when the whole panoramic scene is selected. We use $X_i(t) = 1$ to denote that user's viewport is covered by the delivered portion in time slot t when rate r_i is used ($X_i(t) = 0$ otherwise). As shown in Fig. 2, if we choose the rate corresponding to the delivery portion A, i.e., the green area, the viewport prediction will fail. On the other hand, if we deliver the portion B, i.e., the red area plus the green area, the viewport prediction will be successful. Let

$\alpha_i \triangleq \Pr\{X_i(t)=1\}$ be the *prediction probability*. Obviously, the prediction probability is dependent on both the viewport prediction algorithm and the chosen rate. Here, we consider a general case for all of the efficient viewport prediction algorithms and we will use the linear regression (LR) model to predict the user's motion in experimental evaluations (cf. Section IV-B). Note that the AP gets to know the user's exact viewport after each transmission, even if the transmission fails, since the user's device automatically records the user's current motion orientation and sends that information back to the AP. Hence, if rate r_i was used in time slot t for transmission, the AP always knows the outcome of $X_i(t)$.

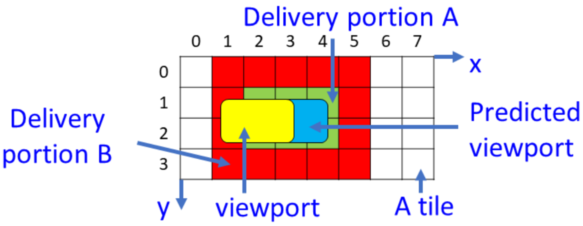


Fig. 2: A panoramic scene with 32 tiles.

We assume that the user's channel rate is independently and identically distributed (i.i.d.) over time. We assume that the channel rate is unknown at the beginning of each time slot[1]. Since we do not buffer panoramic content on the client, we need to deliver the tiles within a time slot to display them on time (e.g., 30 frames per second for panoramic videos or 60 frames per second for VR). As such, if the selected rate is less than or equal to the channel rate, then the wireless transmission will succeed. Otherwise, the transmission will fail. We use $Y_i(t)=1$ to denote a successful transmission when the rate r_i is used in time slot t ($Y_i(t)=0$ if the transmission fails). Let $\beta_i \triangleq \Pr\{Y_i(t)=1\}$ be the *transmission probability*. We use $Z_i(t)=1$ to denote that the user can successfully view the desired content when the rate r_i is used in time slot t ($Z_i(t)=0$ otherwise). Note that $Z_i(t)=1$ happens when both the prediction and the transmission are successful and thus we have $Z_i(t)=X_i(t)Y_i(t)$. Let $i(t)\in\{1,2,...,K\}$ denote the index of the rate used for wireless transmission at time slot t. Then the user's throughput in time slot t is $Z_{i(t)}(t)$[2].

In this paper, the AP needs to make a decision on the selected rate in order to maximize the system throughput. If both the user's prediction and transmission probabilities (i.e., $\{\alpha_i,\beta_i,i=1,2,\dots,K\}$) are known, then the optimal throughput can be achieved by solving the following optimization problem: $i^*\in\arg\max_{i=1,2,\dots,K}\alpha_i\beta_i$. However, both the prediction and transmission probabilities are unknown, since they rely on many factors such as the wireless environment, the panoramic content, and the user's personal behavior. Thus, the algorithm not only needs to learn these statistics (also known as (a.k.a.) exploration) but also to select the best rate so far (a.k.a. exploitation). Our goal is to design a learning algorithm that achieves the maximum system throughput within n time slots, where n is some positive integer. This is equivalent to minimizing the regret, which is the gap between the expected accumulated throughput and the optimal throughput, i.e.,

$$\begin{aligned} R(n) &\triangleq n\alpha_{i^*}\beta_{i^*}-\mathbb{E}\left[\sum_{t=1}^{n}X_{i(t)}(t)Y_{i(t)}(t)\right] \\ &=\sum_{k\neq i^*}\mathbb{E}\left[T_k(n)\Delta_k\right], \end{aligned} \tag{1}$$

where $\Delta_k=\alpha_{i^*}\beta_{i*}-\alpha_k\beta_k$ and $T_k(n)$ denotes the number of times the transmission rate r_k was used until the end of time slot n.

Existing heuristics (e.g., [16]) such as minimum scene delivery correspond to fixing a particular rate in our problem and thus suffer from linear regret. Different from the traditional multi-armed bandit problem (where only the product $X_{i(t)}(t)Y_{i(t)}(t)$ is available as feedback), both prediction and transmission outcomes (i.e., $X_{i(t)}(t)$ and $Y_{i(t)}(t)$) are available to us after each decision on the selected rate. This additional level of feedback information can be leveraged to reduce the regret compared with the single feedback counterpart.

III. Algorithm Design

In this section, we first describe the classic Kullback-Leibler Upper Confidence Bound (KL-UCB) algorithm for the rate selection based on the single feedback information, i.e., whether the user can successfully see the desired content. Then, we develop a variant of KL-UCB algorithm that efficiently leverages two-level feedback information, i.e., both viewport prediction and wireless transmission outcomes, and show that it asymptotically achieves the minimum regret, which is shown to be not greater than that achieved by the single feedback counterpart.

We first present the classic KL-UCB algorithm (see Algorithm 1), which motivates the design for the KL-UCB with two-level feedback information. Let $i^{(I)}(t)$ denotes the index of the rate selected for transmission at time slot t under the classic KL-UCB algorithm and $T_i^{(I)}(t)$ denote the number of times that rate r_i is selected, until time slot t. Let $S_i^{(I)}(t)\triangleq\sum_{\tau=1}^{t}Z_{i^{(I)}(\tau)}(\tau)\mathbb{1}\{i^{(I)}(\tau)=i\}$ denote the number of times that the user successfully sees the desired content when rate r_i is selected until time slot t, where $\mathbb{1}\{\mathcal{A}\}=1$ if event $\mathcal{A}$ is true and 0 otherwise. Let $d(a,b)\triangleq a\log\frac{a}{b}+(1-a)\log\frac{1-a}{1-b}$ denote the KL-divergence between two Bernoulli random variables with mean $a\in(0,1)$ and $b\in(0,1)$, respectively. The classic KL-UCB algorithm uses KL-divergence to indirectly incorporate the uncertainty term in the weight of each rate. In particular, the weight is the largest value such that its KL-divergence away from the sample mean is smaller than some small-term, which is the logarithmic function of the time t. Then, we select the rate with the largest weight.

[1] While the channel rate can be estimated, it is typically inaccurate, especially when the user frequently rotates his/her headset.

[2] Our model can also be extended to other panoramic image coding schemes, where the portion containing the viewport is encoded with a high resolution and the rest of the scenes are encoded with a lower resolution. In such a regime, the rate r_i can be viewed as a measure of the user's image quality, and transmission is considered to fail if the intended quality is not delivered to the user. As such, the user's throughput in time slot t is $r_{i(t)}X_{i(t)}(t)Y_{i(t)}(t)$.

```
Algorithm 1: KL-UCB with single-level feedback
  • Choose each rate once.
  • Subsequently, in each time slot $t$, select the rate index $i^{(I)}(t)$ satisfying
      $i^{(I)}(t) \in \underset{i=1,2,\ldots,K}{\arg\max} \max\Bigg\{p \in [0,1] : d\left(\frac{S_i^{(I)}(t)}{T_i^{(I)}(t)}, p\right) \le \frac{\log(1+t\log^2 t)}{T_i^{(I)}(t)}\Bigg\}.$
```

In our considered setup, both viewport prediction and wireless transmission outcomes are available after each decision on the selected rate. As such, we develop a variant of KL-UCB (see Algorithm 2) that efficiently leverages these two pieces of information. To describe our algorithm, we will define the following quantities. Let $i^{(II)}(t)$ denote the index of the rate selected for transmission at time slot t under the KL-UCB with two-level feedback and $T_i^{(II)}(t)$ denote the number of times that rate r_i is selected, until time slot t. Let $S_{i,1}^{(II)}(t) \triangleq \sum_{\tau=1}^{t} X_{i^{(II)}(\tau)}(\tau)\mathbb{1}\{i^{(II)}(\tau) = i\}$ and $S_{i,2}^{(II)}(t) \triangleq \sum_{\tau=1}^{t} Y_{i^{(II)}(\tau)}(\tau)\mathbb{1}\{i^{(II)}(\tau) = i\}$ respectively denote the number of times that the prediction is successful and the number of times that the wireless transmission is successful when rate r_i is selected until time slot t.

```
Algorithm 2: KL-UCB with two-level feedback
  • Choose each rate once.
  • Subsequently, in each time slot $t$, select the rate index $i^{(II)}(t)$ satisfying
      $i^{(II)}(t) \in \underset{i=1,2,\ldots,K}{\arg\max} \max\Bigg\{pq :$
      $d\left(\frac{S_{i,1}^{(II)}(t)}{T_i^{(II)}(t)}, p\right) + d\left(\frac{S_{i,2}^{(II)}(t)}{T_i^{(II)}(t)}, q\right) \le \frac{\log(1+t\log^2 t)}{T_i^{(II)}(t)};$
      $\frac{S_{i,1}^{(II)}(t)}{T_i^{(II)}(t)} \le p \le 1; \quad \frac{S_{i,2}^{(II)}(t)}{T_i^{(II)}(t)} \le q \le 1\Bigg\}.$
```

Intuitively, in Algorithm 2, we maintain a pair of counters (i.e., $S_{i,1}^{(II)}(t)$ and $S_{i,2}^{(II)}(t)$) to track prediction and transmission outcomes when rate r_i is used, and use these counters to obtain the corresponding estimated probabilities for successful prediction and transmission. Different from the classic KL-UCB with single feedback (cf. Algorithm 1), we jointly consider the uncertainties in the estimated probabilities for successful prediction and transmission. It turns out that such a design asymptotically achieves the minimum regret, which is shown to be not greater than that achieved by the single feedback counterpart.

Theorem 1. *The KL-UCB with two-level feedback (cf. Algorithm 2) asymptotically minimizes the regret in the sense that the achieved regret asymptotically matches that achieved by any uniformly good algorithm (i.e., as $n \to \infty$, its achieved regret (for any valid problem instance) belongs to the set $o(n^{\delta})$[3], for any $\delta \in (0,1)$.), and its regret $R^{(II)}(n)$ satisfies:*

$$\lim_{n\to\infty} \frac{R^{(II)}(n)}{\log n} = \sum_{k \neq i^*} \frac{\Delta_k}{\min_{\substack{0\le x,y\le 1\\ xy\ge\alpha_{i^*}\beta_{i^*}}} d(\alpha_k, x) + d(\beta_k, y)},$$

where we recall that $i^* \in \arg\max_{i=1,2,\ldots,K} \alpha_i\beta_i$.

Proof: We first characterize the fundamental lower bound on the regret (as defined in (1)) achieved by any uniformly good algorithm if two-level feedback is available, i.e., for the stochastic multi-armed bandit problem with two-level feedback, the following result holds for the regret achieved by any uniformly good algorithm ψ:

$$\liminf_{n\to\infty} \frac{R(n)}{\log n} \ge \sum_{k \neq i^*} \frac{\Delta_k}{\min\limits_{\substack{0\le x,y\le 1\\ xy\ge\alpha_{i^*}\beta_{i^*}}} d(\alpha_k, x) + d(\beta_k, y)}. \tag{2}$$

This derivation is along line of the analysis in [11], and is omitted due to the space limit. Then, we analyze the regret performance of the KL-UCB with two-level feedback and show that it is asymptotically optimal. The intuition behind the regret analysis is along similar lines as the intuition behind the analysis of the classic KL-UCB algorithm (see [10] for more details). From the definition of regret in (1), we note that in order to prove an upper bound on the regret achieved by Algorithm 2, we simply need to upper bound the number of times the algorithm transmits at a sub-optimal rate. To this end, we split the analysis into the following steps:

1) Let τ be defined as the time after which, for a small $\epsilon > 0$, the optimal rate's index (as computed by Algorithm 2) will always be strictly greater than $(\alpha_{i^*} - \epsilon)(\beta_{i^*} - \epsilon)$. Intuitively, after time τ, the optimal rate's index will be close to its true value. We will upper bound $\mathbb{E}[\tau]$ in Lemma 2 in Section V.
2) We bound the expected number of times the index of a sub-optimal rate will be greater than $\alpha_i\beta_i + \epsilon$, for a small $\epsilon > 0$ (see Lemma 3). After time τ, a sub-optimal rate will be transmitted only if its index exceeds its true index substantially, since the optimal rate's index will be close to its true index. Therefore, Lemma 3 in Section V allows us to upper bound the number of times a sub-optimal rate will be transmitted after the time τ.
3) We combine the above two results to bound the expected number of times a sub-optimal rate is transmitted and subsequently get the bound on regret.

The detailed proof is available in Section V. ■

It has been shown that the classic KL-UCB algorithm asymptotically minimizes the regret in the presence of single-

[3] $f(n) \in o(g(n))$ if for all $c > 0$, there exists some $k > 0$ such that $0 \le f(n) < cg(n), \forall n \ge k$

feedback information and its regret $R^{(I)}(n)$ (see [10]) satisfies:

$$\lim_{n\to\infty} \frac{R^{(I)}(n)}{\log n} = \sum_{k\neq i^*} \frac{\Delta_k}{\min\limits_{\substack{0\leq x,y\leq 1\\ xy\geq \alpha_{i^*}\beta_{i^*}}} d(\alpha_k\beta_k, xy)}.$$

Next, we show that the achieved regret under the KL-UCB with two-level feedback is less than or equal to that with single feedback information (treating the product as the feedback) asymptotically.

Theorem 2. *The achieved regret under the KL-UCB with two-level feedback is not greater than that under the single feedback counterpart in the asymptotic regime, i.e.,*

$$\begin{aligned}\sum_{k\neq i^*} &\frac{\Delta_k}{\min\limits_{\substack{0\leq x,y\leq 1\\ xy\geq \alpha_{i^*}\beta_{i^*}}} d(\alpha_k, x) + d(\beta_k, y)} \\ &\leq \sum_{k\neq i^*} \frac{\Delta_k}{\min\limits_{\substack{0\leq x,y\leq 1\\ xy\geq \alpha_{i^*}\beta_{i^*}}} d(\alpha_k\beta_k, xy)}. \end{aligned} \tag{3}$$

Proof: It suffices to show that

$$d(\alpha_k, x) + d(\beta_k, y) \geq d(\alpha_k\beta_k, xy), \quad \forall x, y \in (0,1). \tag{4}$$

We consider four independent Bernoulli random variables $X_1 \sim Ber(\alpha_k)$, $Y_1 \sim Ber(\beta_k)$, $X_2 \sim Ber(x)$, and $Y_2 \sim Ber(y)$. By considering the two random vectors $Z_1 = (X_1, Y_1)$ and $Z_2 = (X_2, Y_2)$, from the additive property of the KL-divergence (see [17] for details) for independent random variables, we get

$$\begin{aligned} KL(Z_1||Z_2) &= KL(X_1||X_2) + KL(Y_1||Y_2) \\ &= d(\alpha_k, x) + d(\beta_k, y), \end{aligned} \tag{5}$$

where we have used the notation KL to denote the KL-divergence between the distributions of random vectors or random variables and $d(a, b)$ is the KL-divergence between two Bernoulli random variables with means a and b. Next, consider a channel which takes two Bernoulli random variables X, Y as input and produces the output XY. Suppose we give $Z_1 = (X_1, Y_1)$ and $Z_2 = (X_2, Y_2)$ as inputs to this channel, the KL-divergence between the outputs will be less than or equal to the KL-divergence between the corresponding inputs, a property known as the data processing inequality in information theory [17]. Thus, we have

$$KL(Z_1||Z_2) \geq KL(X_1Y_1||X_2Y_2) = d(\alpha_k\beta_k, xy). \tag{6}$$

By combining (5) and (6), we have the desired result. ■

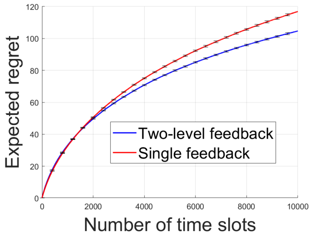
Remark 1. *We would like to point out that depending on the values of $\{\alpha_i, \beta_i; i = 1, 2, ..., K\}$, the difference between the regret in the two-level feedback case and that in the traditional single feedback case can be significant.*

IV. Performance Evaluation

In this section, we first perform synthetic simulations to compare the regret performance between our proposed KL-UCB algorithm with two-level feedback and single feedback counterpart. Then, we implement KL-UCB algorithms with both single and two-level feedback in a real system, and demonstrate their superior performance over existing heuristic methods.

	α_n	β_n
r_1	0.1	0.99
r_2	0.3	0.6
r_3	0.5	0.4
r_4	0.65	0.2
r_5	0.9	0.05

(a) Simulation parameters



(b) Regret performance

Fig. 3: Synthetic simulation

A. Synthetic Simulation

In this subsection, we consider a synthetic experiment with both the prediction and transmission outcomes being directly generated by Bernoulli random variables with means α_i and β_i, respectively, when rate r_i is used. In such a case, we consider the simulation setup with five different selected rates, as listed in Fig. 3a. In the simulation setup, we run 5000 experiments to get the average results and each experiment's time horizon is set to 10^4 time slots. We plot the mean and 1.96 standard deviation (95% confidence interval) of the regret in Fig. 3b. We can observe from Fig. 3b that KL-UCB algorithm with two-level feedback outperforms its counterpart with single feedback information, which coincides with our theoretical analysis in Section III.

B. Real-World Experiments

In this subsection, we design a system for the interactive panoramic scene delivery, where a client requests the panoramic scene from a server via WiFi and displays it in real-time. We use a commercial off-the-shelf smartphone (Google Pixel 4XL) as the client and a Lambda workstation with Intel Core i9-9920X CPU @ 3.50GHz × 24, NVIDIA GeForce RTX 2080 Ti Graphics Card × 4, 128 GB memory, 2 TB disk, and Ubuntu 18.04 as the edge server. A Netgear R6700 router is responsible for the wireless communication between the client and the server. We compare KL-UCB algorithms with both single and two-level feedback with the existing heuristic methods (e.g. [16], [6], [18]) such as minimum scene delivery (i.e. delivering tiles that are overlapped with the user's predicted viewport) and whole scene delivery. We record a 3 DoF (orientation only) motion trace for a free educational panoramic video (see [19]) and a 6 DoF (3 DoF for position and 3 DoF for orientation) motion trace for a commercial VR scene (see [20]), respectively. Notice that the VR scene is displayed in 60 FPS, while the panoramic video plays in 30 FPS which matches the frame rate of the original video.

Client Design. The client is built on Android Studio using Android SDK and Java. The motion thread on the client sends its current trace and the transmission result of the last time slot to the server in each time slot (the slot duration is determined by the target frame rate, 33 ms for 30 FPS, 16 ms for 60 FPS). The other thread is responsible for receiving the tiles from the server and passing it to the decoders. If the packet loss is detected or the tiles cannot be delivered within a time slot, the transmission will be regarded as a failure. Note that the client will not buffer the panoramic scene for the interactive applications. Android Media Codec accelerates the decoding of the received tiles with multiple hardware decoders working in parallel. The number of the decoders is set to 15 in our experiments. We use Open GL ES to do the reprojection from the equirectangular map to the panoramic view to display the received tiles. The viewport of the client is set to a typical value of $100^\circ \times 90^\circ$ (see [6], [16]). We display tiles that have an overlap with the current viewport.

Server Design. The server is developed on Eclipse by Java. Once the server receives the motion trace from the client, it predicts the pose in the next time slot using the linear regression model. To be specific, we use motion trajectories in previous three time slots to train a LR model and drop it after the prediction. Based on the prediction and transmission results of the last time slot, the server updates the KL-UCB algorithms. Then, the server delivers the tiles overlapped with the predicted viewport plus a margin area whose size is determined by the selected rate. In our experiments, we have set five arms corresponding to different portions: $100^\circ \times 90^\circ$ (minimum viewport), $102^\circ \times 91^\circ$, $108^\circ \times 94^\circ$, $120^\circ \times 100^\circ$, and $360^\circ \times 180^\circ$ (whole 360° scene).

Offline Preparation. To focus on the communication part of the system, all the tiles are generated offline based on the original video and the VR scene by FFmpeg [21] in 4K resolution with default Constant Rate Factor (CRF) of 23. We split each equirectangular map of the panoramic frame into 4×6 tiles to save more bandwidth. We assume that all tiles are stored on the memory before the transmission such that the processing time of tiles is negligible during the runtime.

Communication Methodology. To avoid the influence of the TCP rate control algorithm on the performance of the communication, we use real-time transport protocol (RTP), which is built over UDP and accepted by state-of-the-art real-time video streaming systems (see [22], [23]). Since RTP is unreliable, we may lose some of the contents during the transmission, which matches the assumption on the unreliable wireless transmission in our problem formulation (cf. Section II). We limit the maximal bandwidth of the communication to 100 Mbps by Linux TC [24] to avoid trivial portion selection.

Performance Comparison. We repeat fifty times of experiments for each algorithm and get the average result to reduce the randomness. The evaluation results for the panoramic video are shown in Fig. 4. Fig. 4a shows the required network bandwidth of different algorithms compared with the whole panoramic scene delivery. We can observe that both KL-UCB algorithms with single and two-level feedback save up to 50% of the network bandwidth. Fig. 4b shows the throughput improvement of different algorithms compared with the heuristic minimum scene delivery. We can see that both KL-UCB algorithms reaches a 10% performance improvement. We also compare the cumulative regret between the KL-UCB with two-level feedback and single feedback counterpart compared with the best fixed-arm policy, as shown in Fig. 4c. Notice that the tile size varies in each time slot due to the different panoramic scenes, which means the transmission probability varies even under the same network condition. Therefore, the optimal arm keeps changing. As such, we run experiments for each fixed arm and choose the arm with the best average throughput over the whole time horizon. We can observe from Fig. 4c that KL-UCB with two-level feedback outperforms single feedback counterpart. We can observe similar phenomena in the VR application, as shown in Fig. 5. However, the performance improvement between the KL-UCB with two-level feedback and single feedback counterpart is smaller than that for the panoramic video application. This is because that the motion prediction errors in VR applications are not only contributed by head orientation prediction (as in panoramic video applications) but also the virtual location prediction.

V. Performance Analysis of KL-UCB with Two-Level Feedback

In this section, we analyze the regret achieved by the KL-UCB with two-level feedback (cf. Algorithm 2). First, we present and prove an important lemma that quantifies the effect that small perturbations in the constraints of the optimization problem in the denominator of the lower bound (cf. (2)) have on its solution. This lemma is extremely critical in our analysis for establishing an upper bound on the regret achieved by Algorithm 2, and is also useful in the proof of the lower bound.

Lemma 1. *For $0 \leq x, y < 1$ and a constant c such that $xy < c \leq 1$, consider the following optimization problems:*

$$\big(p^*(x,y), q^*(x,y)\big) = \arg \min_{0 \leq p,q \leq 1;\ pq \geq c} d(x,p) + d(y,q)$$

$$\big(p^*_{-\epsilon}(x,y), q^*_{-\epsilon}(x,y)\big) = \arg \min_{0 \leq p,q \leq 1;\ pq \geq c-\epsilon} d(x,p) + d(y,q)$$

For $0 \leq x, y < 1$ and a constant c such that $xy < c < 1$, consider the following optimization problem:

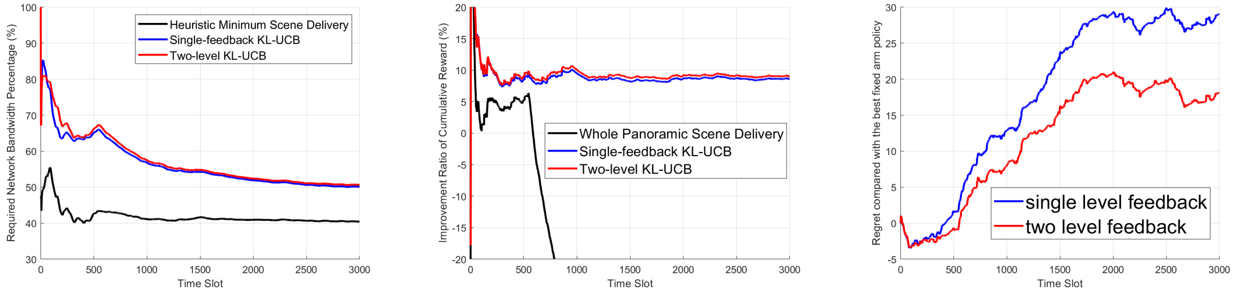
$$\big(p^*_{+\epsilon}(x,y), q^*_{+\epsilon}(x,y)\big) = \arg \min_{0 \leq p,q \leq 1;\ pq \geq c+\epsilon} d(x,p) + d(y,q)$$

The following results hold for the solutions to the above three optimization problems:

1) *$p^*(x,y) > x$ and $q^*(x,y) > y$.*
2) *Let $h_c(x,y) \triangleq \sqrt{(x-y)^2 + 4c(1-x)(1-y)}$. The following results hold:*
   (a) *For $\epsilon < \min\{\frac{h_c^2(x,y)}{8}, \frac{c-xy}{1+x+y}\}$:*

$$\min_{\substack{\alpha,\beta \in [x-\epsilon, x+\epsilon] \\ \times [y-\epsilon, y+\epsilon]}} p^*(\alpha,\beta) \geq p^*(x,y) - \frac{1 + \frac{4}{h_c(x,y)}}{1-y}\epsilon$$

$$\min_{\substack{\alpha,\beta \in [x-\epsilon, x+\epsilon] \\ \times [y-\epsilon, y+\epsilon]}} q^*(\alpha,\beta) \geq q^*(x,y) - \frac{1 + \frac{4}{h_c(x,y)}}{1-x}\epsilon$$

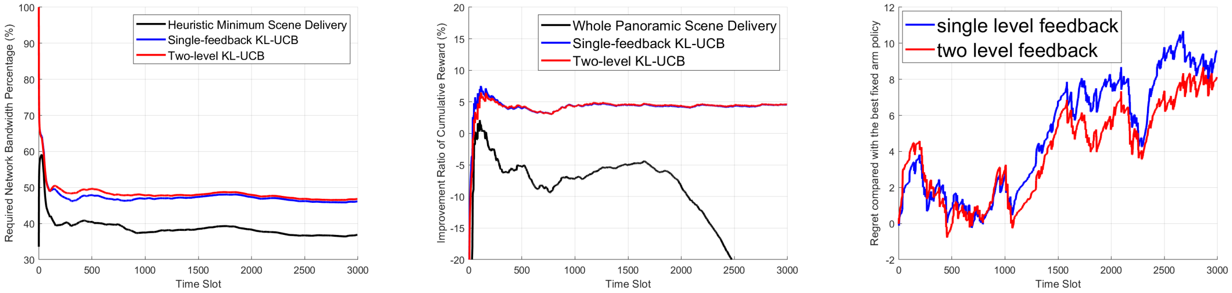


(a) Required network bandwidth w.r.t whole panoramic scene delivery

(b) Throughput improvement over heuristic minimum scene delivery

(c) Comparison between two-level and single-level KL-UCB

Fig. 4: Real-world evaluation results for the panoramic video with 30 FPS



(a) Required network bandwidth w.r.t whole panoramic scene delivery

(b) Throughput improvement over heuristic minimum scene delivery

(c) Comparison between two-level and single-level KL-UCB

Fig. 5: Real-world evaluation results for the VR scene with 60 FPS

(b) For $\epsilon < \min\{\frac{h_c^2(x,y)}{8}, c - xy\}$:

$$p^*_{-\epsilon}(x,y) \geq p^*(x,y) - \frac{2(1-x)}{h_c(x,y)}\epsilon$$

$$\text{and } q^*_{-\epsilon}(x,y) \geq q^*(x,y) - \frac{2(1-y)}{h_c(x,y)}\epsilon,$$

(c) For $\epsilon < \min\{\frac{h_c^2(x,y)}{8}, 1 - c\}$:

$$p^*_{+\epsilon}(x,y) \geq p^*(x,y) + \frac{1-x}{2h_c(x,y)}\epsilon$$

$$\text{and } q^*_{+\epsilon}(x,y) \geq q^*(x,y) + \frac{1-y}{2h_c(x,y)}\epsilon,$$

Additionally, the following bounds quantify the impact of perturbations on the KL-divergence between two Bernoulli random variables.

3) Let $\beta > \alpha$.

a) For $\epsilon_1 \in [0, \frac{1-\alpha}{2}]$ and $\epsilon_2 \in [0, 1-\beta]$ such that $\epsilon_1+\epsilon_2 < \beta - \alpha$, the following result holds:

$$d(\alpha+\epsilon_1, \beta-\epsilon_2) \geq d(\alpha,\beta) - c_1\epsilon_1 - c_2\epsilon_2,$$

where $c_1 = \log\frac{\beta(1-\alpha)}{\alpha(1-\beta)} + 2$ and $c_2 = \frac{1-\alpha}{1-\beta}$.

b) For $\epsilon_1 \in [0, \min\{\frac{\alpha}{2}, \frac{1-\alpha}{2}\}]$ and $\epsilon_2 \in [0, \min\{\frac{\beta}{2}, \frac{1-\beta}{2}\}]$, the following result holds:

$$d(\alpha-\epsilon_1, \beta+\epsilon_2) \leq d(\alpha,\beta) + c_1'\epsilon_1 + c_2'\epsilon_2,$$

where $c_1' = \log\frac{\beta(1-\alpha)}{\alpha(1-\beta)} + 4$ and $c_2' = \frac{2(1-\alpha)}{(1-\beta)}$.

The proof follows tedious but straightforward calculus/algebra and thus is omitted here due to the space limit.

Theorem 3. *The regret achieved by Algorithm 2 for the stochastic multi-armed bandit problem with two-level feedback satisfies the following:*

$$\limsup_{n\to\infty} \frac{R(n)}{\log n} \leq \sum_{k\neq i^*} \frac{\Delta_k}{\min_{\substack{0\leq x,y\leq 1\\ xy\geq\alpha_{i^*}\beta_{i^*}}} d(\alpha_k, x) + d(\beta_k, y)}.$$

The proof of Theorem 3 needs to upper bound the number of times the algorithm transmits at a sub-optimal rate. Hence, we need to first establish the following two lemmas.

Lemma 2. *(Underestimating the optimal arm) Let $X_1, X_2, ..., X_n$ and $Y_1, Y_2, ..., Y_n$ be independent and identically distributed Bernoulli random variables with mean α and β, respectively. Let $\hat{\alpha}_s = \frac{1}{s}\sum_{j=1}^{s} X_j$ and $\hat{\beta}_s = \frac{1}{s}\sum_{j=1}^{s} Y_j$. Let $\epsilon > 0$, $d^+(x,y) = d(x,y)\mathbb{I}(x \leq y)$, $f(t) = 1 + t\log^2 t$ and*

$$\tau = \min\Big\{t : \max_{1\leq s\leq n} d^+(\hat{\alpha}_s, \alpha-\epsilon) - \frac{\log f(t)}{s} \leq 0 \text{ and}$$

$$\max_{1\leq s\leq n} d^+(\hat{\beta}_s, \beta-\epsilon) - \frac{\log f(t)}{s} \leq 0\Big\}$$

Then, $\mathbb{E}[\tau] \leq \frac{4}{\epsilon^2}$.

Proof.

$$\begin{aligned}\mathbb{P}(\tau > t) \leq &\mathbb{P}(\{\exists 1 \leq s \leq n : d^+(\hat{\alpha}_s, \alpha - \epsilon) - \frac{\log f(t)}{s} > 0\} \text{ or } \\ &\{\exists 1 \leq s \leq n : d^+(\hat{\beta}_s, \beta - \epsilon) - \frac{\log f(t)}{s} > 0\}) \\ \leq &\mathbb{P}(\{\exists 1 \leq s \leq n : d^+(\hat{\alpha}_s, \alpha - \epsilon) - \frac{\log f(t)}{s} > 0\}) \\ &+\mathbb{P}(\{\exists 1 \leq s \leq n : d^+(\hat{\beta}_s, \beta - \epsilon) - \frac{\log f(t)}{s} > 0\}),\end{aligned} \tag{7}$$

where the last inequality follows from the union bound. Let us consider the first term on the right-hand side of the above inequality (the second term can be analysed similarly). Note that the proof is similar to the proof of in [10, Lemma 10.7].

$$\begin{aligned}&\mathbb{P}(\{\exists 1 \leq s \leq n : d^+(\hat{\alpha}_s, \alpha - \epsilon) - \frac{\log f(t)}{s} > 0\}) \\ &\leq \sum_{s=1}^{n} \mathbb{P}(d^+(\hat{\alpha}_s, \alpha - \epsilon) - \frac{\log f(t)}{s} > 0) \\ &\overset{(a)}{=} \sum_{s=1}^{n} \mathbb{P}(d(\hat{\alpha}_s, \alpha - \epsilon) - \frac{\log f(t)}{s} > 0, \hat{\alpha}_s < \alpha - \epsilon) \\ &\overset{(b)}{\leq} \sum_{s=1}^{n} \mathbb{P}(d(\hat{\alpha}_s, \alpha) > \frac{\log f(t)}{s} + 2\epsilon^2, \hat{\alpha}_s < \alpha) \\ &\overset{(c)}{\leq} \sum_{s=1}^{n} \exp(-s(\frac{\log f(t)}{s} + 2\epsilon^2)) \\ &= \frac{1}{f(t)} \sum_{s=1}^{n} \exp(-2\epsilon^2 s) \leq \frac{1}{2f(t)\epsilon^2},\end{aligned} \tag{8}$$

where (a) follows from the definition of $d^+(x, y)$, (b) follows from [10, Lemma 10.2], (c) follows from [10, Corollary 10.4]. A similar analysis can be done for the second term on the right-hand side of Equation (7) to obtain:

$$\mathbb{P}(\{\exists 1 \leq s \leq n : d^+(\hat{\beta}_s, \beta - \epsilon) - \frac{\log f(t)}{s} > 0\}) \leq \frac{1}{2f(t)\epsilon^2}.$$

Combining the above inequality with Equations (7)-(8):

$$\mathbb{E}[\tau] = \int_0^\infty \mathbb{P}(\tau \geq t)dt \leq \int_0^\infty \frac{1}{f(t)\epsilon^2}dt \leq \frac{4}{\epsilon^2}, \tag{9}$$

where the last inequality follows from the fact that $\int_0^\infty \frac{dt}{\log(1+t\log^2 t)} \leq 4$. □

Lemma 3. *(Overestimating a sub-optimal arm) Let $X_1, X_2, ..., X_n$ and $Y_1, Y_2, ..., Y_n$ be independent and identically distributed Bernoulli random variables with mean α and β, respectively. Assume that $\alpha\beta < 1$. Let $\hat{\alpha}_s = \frac{1}{s}\sum_{j=1}^{s} X_j$ and $\hat{\beta}_s = \frac{1}{s}\sum_{j=1}^{s} Y_j$. Let $h_c(x, y)$ be as defined in Lemma 1. Let $\Delta > 0$, such that $\alpha\beta + \Delta < 1$. Let $a > 0$ and*

$$p^*, q^* = \arg \min_{\substack{0 \leq p,q \leq 1 \\ pq \geq \alpha\beta + \Delta}} d(\alpha, p) + d(\beta, q)$$

$$\kappa = \sum_{s=1}^{n} \mathbb{I}\Big\{ \min_{\substack{0 \leq x,y \leq 1 \\ xy \geq \alpha\beta + \Delta}} d(\hat{\alpha}_s, x) + d(\hat{\beta}_s, y) \leq \frac{a}{s} \Big\}$$

1) If $0 \leq \alpha, \beta < 1$, then

$$\mathbb{E}[\kappa] \leq \inf_{\epsilon \in (0, \Gamma_{\alpha,\beta,\Delta})} \big(\frac{a}{M_{\alpha,\beta,\Delta,\epsilon}} + \frac{2}{\epsilon^2}\big),$$

where

$$\begin{aligned}M_{\alpha,\beta,\Delta,\epsilon} \triangleq &d(\alpha + \epsilon, p^* - \frac{1 + \frac{4}{h_{\alpha\beta+\Delta}(\alpha,\beta)}}{1 - \beta}\epsilon) \\ &+ d(\beta + \epsilon, q^* - \frac{1 + \frac{4}{h_{\alpha\beta+\Delta}(\alpha,\beta)}}{1 - \alpha}\epsilon),\end{aligned}$$

and

$$\begin{aligned}\Gamma_{\alpha,\beta,\Delta} \triangleq \min\{&\frac{h^2_{\alpha\beta+\Delta}(\alpha, \beta)}{8}, \frac{\Delta}{1 + \alpha + \beta}, \\ &\frac{(p^* - \alpha)(1 - \beta)}{2 - \beta + \frac{4}{h_{\alpha\beta+\Delta}(\alpha,\beta)}}, \frac{(q^* - \beta)(1 - \alpha)}{2 - \alpha + \frac{4}{h_{\alpha\beta+\Delta}(\alpha,\beta)}}\}.\end{aligned}$$

2) If $\alpha = 1$ or $\beta = 1$, then

$$\mathbb{E}[\kappa] \leq \inf_{\epsilon \in (0, \Delta)} \big(\frac{a}{d(\gamma + \epsilon, \gamma + \Delta)} + \frac{3}{2\epsilon^2}\big),$$

where $\gamma \triangleq \min\{\alpha, \beta\}$.

Proof. Let $\epsilon \in (0, \Delta)$ and $\xi = \frac{a}{M_{\alpha,\beta,\Delta,\epsilon}}$. Let

$$\mathcal{L}_{s,\epsilon} \triangleq \big\{\{X_i, Y_i\}_{i=1}^{s} : |\hat{\alpha}_s - \alpha| > \epsilon \text{ or } |\hat{\beta}_s - \beta| > \epsilon\big\}.$$

We have

$$\begin{aligned}\mathbb{E}[\kappa] = &\sum_{s=1}^{n} \mathbb{P}\big(\min_{\substack{0 \leq x,y \leq 1; \\ xy \geq \alpha\beta + \Delta}} d(\hat{\alpha}_s, x) + d(\hat{\beta}_s, y) \leq \frac{a}{s}\big) \\ &\leq \sum_{s=1}^{n} \mathbb{P}(\mathcal{L}_{s,\epsilon}) \\ &\quad + \sum_{s=1}^{n} \mathbb{P}\big(\min_{\substack{0 \leq x,y \leq 1; \\ xy \geq \alpha\beta + \Delta}} d(\hat{\alpha}_s, x) + d(\hat{\beta}_s, y) \leq \frac{a}{s}\Big| \mathcal{L}^C_{s,\epsilon}\big) \\ &\overset{(a)}{\leq} \sum_{s=1}^{n} \mathbb{P}(\mathcal{L}_{s,\epsilon}) + \sum_{s=1}^{n} \mathbb{P}\Big(M_{\alpha,\beta,\Delta,\epsilon} \leq \frac{a}{s} \Big| \mathcal{L}^C_{s,\epsilon}\Big) \\ &\overset{(b)}{\leq} \sum_{s=1}^{n} \mathbb{P}(\mathcal{L}_{s,\epsilon}) + \xi \\ &\leq \sum_{s=1}^{n} \mathbb{P}(|\hat{\alpha}_s - \alpha| > \epsilon) + \sum_{s=1}^{n} \mathbb{P}(|\hat{\beta}_s - \beta| > \epsilon) + \xi \\ &\overset{(c)}{\leq} \sum_{s=1}^{\infty} \Big(\exp\big(- sd(\alpha + \epsilon, \alpha)\big) + \exp\big(- sd(\alpha - \epsilon, \alpha)\big)\Big) \\ &+ \sum_{s=1}^{\infty} \Big(\exp\big(- sd(\beta + \epsilon, \beta)\big) + \exp\big(- sd(\beta - \epsilon, \beta)\big)\Big) + \xi\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{1}{d(\alpha+\epsilon,\alpha)} + \frac{1}{d(\alpha-\epsilon,\alpha)} + \frac{1}{d(\beta+\epsilon,\beta)} + \\
&\quad + \frac{1}{d(\beta-\epsilon,\beta)} + \xi \overset{(d)}{\leq} \frac{2}{\epsilon^2} + \frac{a}{M_{\alpha,\beta,\Delta,\epsilon}},
\end{aligned}
$$

where (a) follows from the result 2(a) in Lemma 1, (b) follows from the definition of ξ, (c) follows from Chernoff's bound, and (d) follows from Pinsker's inequality. The result follows from taking the infimum over ϵ to obtain the tightest bound.

Let us consider the case when either $\alpha = 1$ or $\beta = 1$ (both can not be equal to one due to the assumption that $\alpha\beta < 1$). Without loss of generality, let us assume that $\alpha = 1$ and $\beta < 1$. It can be readily seen that $p^* = 1$ and $q^* = \alpha\beta + \Delta = \beta + \Delta$ (similar to proof of Lemma 1). With the above observation, the result can be proved similar to the proof for the previous case (when $0 \leq \alpha, \beta < 1$). □

Proof of Theorem 3:
Equipped with the three lemmas we have, we now prove the main result. Consider a sub-optimal rate r_i, i.e., $i \neq i^*$. Recall that $T_i(n)$ denotes the number of times that the rate r_i is used for transmission until the end of time slot n. We will bound $\mathbb{E}[T_i(n)]$ and eventually bound the overall regret using (1). Let $(p^*_{i^*}, q^*_{i^*}) = \arg \min_{\substack{0\leq p,q\leq 1\\ pq\geq \alpha_{i^*}\beta_{i^*}}} d(\alpha_{i^*}, p) + d(\beta_{i^*}, q)$ and $f(t) = 1 + t\log^2 t$. We will split the analysis into two cases: (i) $0 \leq \alpha_i, \beta_i < 1$, (ii) either $\alpha_i = 1$ or $\beta_i = 1$.

Case 1: $0 \leq \alpha_i, \beta_i < 1$.

Choose $\epsilon_1 > 0$ such that $\epsilon_1(\alpha_{i^*} + \beta_{i^*}) < \alpha_{i^*}\beta_{i^*} - \alpha_i\beta_i$. Also, let:

$$
\begin{aligned}
\tau = \min\Big\{ t : &\max_{1\leq s\leq n} d^+(\hat{\alpha}_{i^*,s}, \alpha_{i^*} - \epsilon_1) - \frac{\log f(t)}{s} \leq 0 \\
&\text{and } \max_{1\leq s\leq n} d^+(\hat{\beta}_{i^*,s}, \beta_{i^*} - \epsilon_1) - \frac{\log f(t)}{s} \leq 0 \Big\}
\end{aligned}
$$

$$
\kappa = \sum_{s=1}^{n} \mathbb{I}\Big\{ \min_{\substack{0\leq x,y\leq 1;\\ xy\geq \Upsilon_{i,\epsilon_1}}} d(\hat{\alpha}_{i,s}, x) + d(\hat{\beta}_{i,s}, y) \leq \frac{\log f(n)}{s} \Big\},
$$

where $\Upsilon_{i,\epsilon_1} \triangleq \alpha_{i^*}\beta_{i^*} - \epsilon_1(\alpha_{i^*} + \beta_{i^*})$.

We have:

$$
\begin{aligned}
\mathbb{E}[T_i(n)] &= \mathbb{E}\big[\sum_{t=1}^{n} \mathbb{I}\{A_t = i\}\big] \\
&\leq \mathbb{E}[\tau] + \mathbb{E}\big[\sum_{t=\tau+1}^{n} \mathbb{I}\{A_t = i\}\big] \\
&\overset{(a)}{\leq} \mathbb{E}[\tau] + \mathbb{E}\big[\sum_{t=1}^{n} \mathbb{I}\{A_t = i \text{ and} \\
&\qquad \min_{\substack{0\leq x,y\leq 1;\\ xy\geq \Upsilon_{i,\epsilon_1}}} d(\hat{\alpha}_{i,s}, x) + d(\hat{\beta}_{i,s}, y) \leq \frac{\log f(t)}{T_i(t-1)}\}\big] \\
&\overset{(b)}{\leq} \mathbb{E}[\tau] + \mathbb{E}[\kappa],
\end{aligned}
$$

where (a) follows from Algorithm 2 and the definition of τ, and (b) follows from the definition of κ. Combining the above inequality with different lemmas proved previously:

$$
\begin{aligned}
\mathbb{E}[T_i(n)] &\leq \frac{4}{\epsilon_1^2} + \inf_{\epsilon_2 \in (0, \Gamma_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1}})} \left(\frac{\log(1 + t\log^2 t)}{M_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1},\epsilon_2}} + \frac{2}{\epsilon_2^2} \right) \\
&\leq \frac{4}{\epsilon_1^2} + \inf_{\epsilon_2 \in (0, \Gamma_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1}})} \left(\frac{\log(1 + t\log^2 t)}{M'_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1},\epsilon_2}} + \frac{2}{\epsilon_2^2} \right),
\end{aligned} \tag{10}
$$

where the first inequality follows from Lemmas 2 and 3 with

$$
\Delta_{i,\epsilon_1} = \alpha_{i^*}\beta_{i^*} - \alpha_i\beta_i - \epsilon_1(\alpha_{i^*} + \beta_{i^*})
$$

and $M_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1},\epsilon_2}$ and $\Gamma_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1}}$ as defined in Lemma 3.

The second inequality follows from Lemma 1 with

$$
\begin{aligned}
M'_{\alpha_i,\beta_i,\Delta_{i,\epsilon_1},\epsilon_2} &= d(\alpha_i + \epsilon_2, p_i^* - k_1\epsilon_1 - k_2\epsilon_2) \\
&\quad + d(\beta_i + \epsilon_2, q_i^* - k_3\epsilon_1 - k_4\epsilon_2),
\end{aligned}
$$

where

$$
p_i^*, q_i^* = \arg \min_{\substack{0\leq p,q\leq 1;\\ pq\geq \alpha_{i^*}\beta_{i^*}}} d(\alpha_i, p) + d(\beta_i, q),
$$

$$
k_1 = \frac{2(1-\alpha_i)(\alpha_{i^*} + \beta_{i^*})}{h_{\alpha_{i^*}\beta_{i^*}}(\alpha_i, \beta_i)}, k_2 = \frac{1 + \frac{4}{h_{\alpha_i\beta_i + \Delta_{i,\epsilon_1}}(\alpha_i,\beta_i)}}{1 - \beta_i},
$$

$$
k_3 = \frac{2(1-\beta_i)(\alpha_{i^*} + \beta_{i^*})}{h_{\alpha_{i^*}\beta_{i^*}}(\alpha_i, \beta_i)}, k_4 = \frac{1 + \frac{4}{h_{\alpha_i\beta_i + \Delta_{i,\epsilon_1}}(\alpha_i,\beta_i)}}{1 - \alpha_i}.
$$

Case 2: either $\alpha_i = 1$ or $\beta_i = 1$. In this case, we can use Lemma 3 and proceed as we did in the previous case to get:

$$
\mathbb{E}[T_i(n)] \leq \frac{4}{\epsilon_1^2} + \inf_{\epsilon_2 \in (0, \Delta_{i,\epsilon_1})} \Big(\frac{\log(1 + t\log^2 t)}{d(\gamma + \epsilon_2, \gamma + \Delta_{i,\epsilon_1})} + \frac{3}{2\epsilon_2^2} \Big), \tag{11}
$$

where $\epsilon_1 \in (0, \frac{\alpha_{i^*}\beta_{i^*} - \alpha_i\beta_i}{\alpha_{i^*} + \beta_{i^*}})$, $\gamma = \min\{\alpha, \beta\}$ and Δ_{i,ϵ_1} as defined in the previous case.

The final result can be obtained by combining (10)–(11), using results 2 and 3 from Lemma 1 and taking limit superior (see [10, Chapter 8] for more details).

VI. Conclusion

In this paper, we considered a multi-armed bandit problem with an application to interactive panoramic scene delivery over wireless, where each arm corresponds to a delivery portion of the panoramic scene (or a rate). The larger the delivery portion, the higher the viewport prediction probability and lower the wireless transmission success probability. We proposed KL-UCB algorithms with both single and two-level feedback for the rate selection, and showed that the KL-UCB algorithm with two-level feedback asymptotically minimizes the regret and its achieved regret is not greater than that with single-feedback counterpart. We perform both synthetic simulations and real experimental evaluations to demonstrate the superior performance of bandit algorithms over existing heuristic methods.

References

[1] Y. Bao, H. Wu, T. Zhang, A. A. Ramli, and X. Liu, “Shooting a moving target: Motion-prediction-based transmission for 360-degree videos,” in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 1161–1170.

[2] R. Combes, S. Magureanu, and A. Proutiere, “Minimal exploration in structured stochastic bandits,” in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 1763–1771. [Online]. Available: http://papers.nips.cc/paper/6773-minimal-exploration-in-structured-stochastic-bandits.pdf

[3] H. Gupta, A. Eryilmaz, and R. Srikant, “Link rate selection using constrained thompson sampling,” in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 739–747.

[4] M. Hosseini and V. Swaminathan, “Adaptive 360 vr video streaming: Divide and conquer,” in *2016 IEEE International Symposium on Multimedia (ISM)*. IEEE, 2016, pp. 107–110.

[5] 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 transactions on pattern analysis and machine intelligence*, vol. 41, no. 11, pp. 2693–2708, 2018.

[6] F. Qian, B. Han, Q. Xiao, and V. Gopalakrishnan, “Flare: Practical viewport-adaptive 360-degree video streaming for mobile devices,” in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018, pp. 99–114.

[7] N. Kan, J. Zou, K. Tang, C. Li, N. Liu, and H. Xiong, “Deep reinforcement learning-based rate adaptation for adaptive 360-degree video streaming,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 4030–4034.

[8] Y. Zhang, P. Zhao, K. Bian, Y. Liu, L. Song, and X. Li, “Drl360: 360-degree video streaming with deep reinforcement learning,” in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 1252–1260.

[9] Y. Guan, C. Zheng, X. Zhang, Z. Guo, and J. Jiang, “Pano: Optimizing 360 video streaming with a better understanding of quality perception,” in *Proceedings of the ACM Special Interest Group on Data Communication*, 2019, pp. 394–407.

[10] T. Lattimore and C. Szepesvári, “Bandit algorithms,” *Available online*, p. 28, 2018.

[11] T. L. Lai and H. Robbins, “Asymptotically efficient adaptive allocation rules,” *Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[12] A. Garivier and O. Cappé, “The kl-ucb algorithm for bounded stochastic bandits and beyond,” in *Proceedings of the 24th annual conference on learning theory*, 2011, pp. 359–376.

[13] S. Agrawal and N. Goyal, “Analysis of thompson sampling for the multi-armed bandit problem,” in *Conference on learning theory*, 2012, pp. 39–1.

[14] K. Cai, K. Chen, L. Huang, and J. C. Lui, “Multi-level feedback web links selection problem: Learning and optimization,” in *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2017, pp. 763–768.

[15] K. Chen, K. Cai, L. Huang, and J. Lui, “Beyond the click-through rate: Web link selection with multi-level feedback,” *arXiv preprint arXiv:1805.01702*, 2018.

[16] X. Liu, C. Vlachou, M. Yang, F. Qian, L. Zhou, C. Wang, L. Zhu, K.-H. Kim, G. Parmer, Q. Chen *et al.*, “Firefly: Untethered multi-user {VR} for commodity mobile devices,” in *2020 {USENIX} Annual Technical Conference ({USENIX}{ATC} 20)*, 2020, pp. 943–957.

[17] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 1999.

[18] Z. Lai, Y. C. Hu, Y. Cui, L. Sun, N. Dai, and H.-S. Lee, “Furion: Engineering high-quality immersive virtual reality on today’s mobile devices,” *IEEE Transactions on Mobile Computing*, vol. 19, no. 7, pp. 1586–1602, 2019.

[19] N. Geographic, “First-ever 3d vr filmed in space.” [Online]. Available: https://vuze.camera/vr-gallery/3d-360-video/first-ever-3d-vr-filmed-space

[20] “QA Office and Security Room.” [Online]. Available: https://assetstore.unity.com/packages/3d/environments/urban/qa-office-and-security-room-114109

[21] “FFmpeg.” [Online]. Available: http://ffmpeg.org/

[22] S. Wang, X. Zhang, M. Xiao, K. Chiu, and Y. Liu, “Sphericrtc: A system for content-adaptive real-time 360-degree video communication,” in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 3595–3603.

[23] H. Zhang, A. Zhou, J. Lu, R. Ma, Y. Hu, C. Li, X. Zhang, H. Ma, and X. Chen, “Onrl: improving mobile video telephony via online reinforcement learning,” in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[24] “Linux TC.” [Online]. Available: http://man7.org/linux/man-pages/man8/tc.8.html