Thompson-Sampling-Based Wireless Transmission for Panoramic Video Streaming

Jiangong Chen, Bin Li, and R. Srikant

***Abstract*—Panoramic video streaming has received great attention recently due to its immersive experience. Different from traditional video streaming, it typically consumes $4 \sim 6\times$ larger bandwidth with the same resolution. Fortunately, users can only see a portion (roughly 20%) of 360° scenes at each time and thus it is sufficient to deliver such a portion, namely Field of View (FoV), if we can accurately predict user's motion. In practice, we usually deliver a portion larger than FoV to tolerate inaccurate prediction. Intuitively, the larger the delivered portion, the higher the prediction accuracy. This however leads to a lower transmission success probability. The goal is to select an appropriate delivered portion to maximize system throughput, which can be formulated as a multi-armed bandit problem, where each arm represents the delivered portion. Different from traditional bandit problems with single feedback information, we have two-level feedback information (i.e., both prediction and transmission outcomes) after each decision on the selected portion. As such, we propose a Thompson Sampling algorithm based on two-level feedback information, and demonstrate its superior performance than its traditional counterpart via simulations.**

I. Introduction

Panoramic or $360°$ video streaming has received great attention in recent years, since it provides immersive experience for users as if they are in a virtual 3D world. One key challenge in high resolution panoramic video streaming is that $360°$ video delivery typically consumes $4 \sim 6\times$ bandwidth of a regular video with the same resolution (see [1]). Fortunately, a user may only need to see as low as 20% of $360°$ scenes, known as Field of View (FoV), depending on her/his perspective without affecting her/his visual perception. For instance, in the case of a panoramic roller coaster video, a user can see either the front views or the back views in a time slot. Thus, if a user's motion is accurately predicted, it is sufficient to deliver just 20% of $360°$ video scenes surrounding him/her, which dramatically reduces the network bandwidth consumption.

In practice, we usually deliver a portion larger than FoV to tolerate the motion prediction error. In order for the user to successfully view his/her desired content, the portion should be successfully delivered and covers the user's FoV. Intuitively, the larger the delivery portion, the more tolerance for motion prediction error and the lower chance for successful wireless transmission. Therefore, the natural question is how to select an appropriate delivery portion at each time with the goal of maximizing system throughput.

Recent works (e.g., [1]–[7]) have developed efficient user's motion prediction algorithms and incorporated them into the panoramic video delivery. However, these results require a large training dataset to achieve desired performance and thus are not directly applicable in fast changing wireless environments. As such, we aim to develop an algorithm that can quickly determine the optimal delivery portion over a finite time horizon. This can be formulated as a multi-armed bandit problem, where each arm corresponds to the delivery portion of the panoramic scene and the goal is to minimize the cumulative regret (i.e., the difference between the optimal cumulative throughput and the cumulative throughput under the proposed algorithm) over a finite time horizon. The main difference lies in that the considered setup has two-level feedback information, i.e., both prediction and transmission outcomes are available after an arm is played. As such, we develop an algorithm based on the state-of-the-art bandit algorithm, known as Thompson Sampling (e.g., [8]–[11]), which samples each arm according to its posterior distribution and selects the optimal arm. In particular, we propose a Thompson Sampling method based on two-level feedback information and show that it outperforms its counterpart with single feedback information via simulations.

II. System Model

We consider a single user downloading a panoramic video from an access point over a wireless channel. We assume that the system operates in a time-slotted manner. At each time slot, the user can only see a portion (typically 20%) of the whole panoramic scene, namely a Field of View (FoV). If a user's head movement can be accurately predicted, then it is sufficient to deliver just 20% of the panoramic images, which significantly reduces the wireless bandwidth consumption. Unfortunately, it is hard to accurately predict a user's motion. As such, we usually deliver a portion larger than the FoV to overcome the inaccurate user's motion prediction.

Let $R(t)$ denote the size of the portion of the panoramic images that are going to be transmitted over the wireless channel in time slot t and thus we call $R(t)$ *the selected rate* in time slot t. We assume that $R(t)$ can only be chosen from the set $\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$ [1], where $0 < r_1 < r_2 < \cdots < r_N$

Jiangong Chen (jiangong_chen@uri.edu) and Bin Li (binli@uri.edu) are with the Department of Electrical, Computer and Biomedical Engineering at the University of Rhode Island, Kingston, RI 02881, USA. R. Srikant (rsrikant@illinois.edu) is with the Department of Electrical and Computer Engineering and Coordinated Science Laboratory at the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA.

This work is supported by the NSF grants: CNS-CAREER-1942383, CNS-NeTS-1815563, CNS-NeTS-1717108, ECCS-1609370, CCF-1934986, ONR grant N00014-19-1-2566 and ARO grant W911NF-19-1-0379.

[1] In practice, panoramic images are usually projected into a rectangular and divided into a finite number of tiles. For example, if you split the whole scene into 4 rows and 8 columns, then you will get 32 tiles in total.

and r_1 and r_N are the size of FoV and the whole panoramic images, respectively. We use $X_n(t) = 1$ to denote that $R(t) = r_n$ is large enough such that the delivered portion covers the user's FoV in time slot t and $X_n(t) = 0$ otherwise. Let $\alpha_n \triangleq \Pr\{X_n(t) = 1\}$ be the *prediction probability* and thus we have $\alpha_1 < \alpha_2 < \cdots < \alpha_N$ (since the larger the delivered portion, the higher the prediction probability). Here, the AP knows the FoV after each transmission even it fails, since the user's device automatically records user's current position (yaw, pitch, roll) and sends back to the AP. Hence, the AP knows the outcome of $X(t)$ no matter whether the transmission succeeds or fails in time slot t.

We use $C(t)$ to capture user's channel fading in time slot t, which is assumed to be independently and identically distributed (i.i.d.) over time. We do not know the channel rate at the beginning of each time slot. If the selected rate is not greater than the channel rate in time slot t (i.e., $R(t) \leq C(t)$), then the wireless transmission will be successful in time slot t. Otherwise, the transmission will fail. We use $Y_n(t) = 1$ to denote a successful transmission by selecting r_n in time slot t and $Y_n(t) = 0$ otherwise. Let $\beta_n \triangleq \Pr\{Y_n(t) = 1\}$ be the *transmission probability*. Note that the higher the selected rate $R(t)$, the lower the probability that the wireless transmission succeeds. Hence, we have $\beta_1 > \beta_2 > \cdots > \beta_N$.

In this paper, the AP needs to make a decision on the selected rate in order to maximize the system throughput. If both user's prediction and transmission probabilities (i.e., $\{\alpha_n, \beta_n, n = 1, 2, \ldots, N\}$) are known, then this can be achieved by solving the following optimization problem:

$$n^* \in \underset{n=1,2,\ldots,N}{\arg\max}\ r_n \alpha_n \beta_n. \tag{1}$$

Unfortunately, both prediction and transmission probabilities are unknown, since they depend on many factors such as user's behavior, panoramic video content, and wireless environment. This requires the algorithm not only to learn these statistics (also known as (a.k.a.) exploration) but also to select the best rate so far (a.k.a. exploitation). Let $I(t) \in \{1, 2, \ldots, N\}$ denote the index of the selected rate in time slot t. Our goal is to design a learning algorithm that achieves the maximum system throughput within T time slots, where T is some positive integer. This is equivalent to minimizing the regret, which is the gap between the accumulated throughput and the optimal throughput, i.e.,

$$\mathrm{Reg}(T) \triangleq T r_{n^*} \alpha_{n^*} \beta_{n^*} - \mathbb{E}\left[\sum_{t=1}^{T} r_{I(t)} X_{I(t)}(t) Y_{I(t)}(t)\right]. \tag{2}$$

In the next section, we will develop an algorithm that effectively minimizes the regret.

III. Algorithm Design

In this section, we will design an algorithm to achieve a low regret over a finite time-horizon T. This is similar to the classical multi-armed bandit problem that can be efficiently solved by the well-known Thompson Sampling (TS) algorithm (see [12]). Indeed, we can combine the prediction and transmission results by letting $Z_n(t) = X_n(t)Y_n(t), \forall n = 1, 2, \ldots, N$, and then the TS algorithm (see Algorithm 1) can asymptotically minimize the regret if we only know the outcome of $Z_{I(t)}(t), t = 1, 2, \ldots, T$. In Algorithm 1, We maintain a pair

Algorithm 1 Thompson Sampling with Single Feedback

for each rate r_n, $n = 1, 2, \ldots, N$, set $S_n = 0$ and $F_n = 0$.
for each $t = 1, 2, \ldots, T$:

1) For each rate r_n, draw $\gamma_n(t) \sim$ Beta $(S_n + 1, F_n + 1)$[2].
2) Choose the selected rate $r_{I(t)}$ satisfying
$$I(t) = \underset{n=1,2,\ldots,N}{\arg\max}\ r_n \gamma_n(t).$$
3) Observe the random outcome $Z_{I(t)}(t)$.
4) (Posterior Update) If $Z_{I(t)}(t) = 1$, set $S_{I(t)} = S_{I(t)} + 1$; otherwise, set $F_{I(t)} = F_{I(t)} + 1$.

end for

of counters which count the number of successes or failures for each arm until time slot t. Then, in each time slot t, we draw posterior probability for each arm from its updated Beta distribution. Finally, we choose the action with the maximal product of the rate and posterior probability and update the counters accordingly.

Different from the traditional multi-armed bandit problem, both prediction and transmission outcomes are available after each decision on the selected rate. As such, we obtain two-level feedback information from the environment. This motivates us to develop a two-level feedback TS algorithm, as shown in Algorithm 2.

Algorithm 2 Thompson Sampling with two-level Feedback

for each rate r_n, $n = 1, 2, \ldots, N$, set $S_n^{(1)} = 0$ and $F_n^{(1)} = 0$ for the first-level feedback, $S_n^{(2)} = 0$ and $F_n^{(2)} = 0$ for the second-level feedback.
for each t=$1, 2, \ldots, T$:

1) For each rate r_n, draw $\alpha_n(t) \sim$ Beta $(S_n^{(1)} + 1, F_n^{(1)} + 1)$ and $\beta_n(t) \sim$ Beta $(S_n^{(2)} + 1, F_n^{(2)} + 1)$.
2) Choose the selected rate $r_{I(t)}$ satisfying
$$I(t) = \underset{n=1,2,\ldots,N}{\arg\max}\ r_n \alpha_n(t) \cdot \beta_n(t).$$
3) Observe the random outcomes for both prediction $X_{I(t)}(t)$ and transmission $Y_{I(t)}(t)$.
4) (Posterior Update) If $X_{I(t)}(t) = 1$, set $S_{I(t)}^{(1)} = S_{I(t)}^{(1)} + 1$; otherwise, set $F_{I(t)}^{(1)} = F_{I(t)}^{(1)} + 1$. If $Y_{I(t)}(t) = 1$, set $S_{I(t)}^{(2)} = S_{I(t)}^{(2)} + 1$; otherwise, set $F_{I(t)}^{(2)} = F_{I(t)}^{(2)} + 1$.

end for

In Algorithm 2, we maintain one pair of counters for each outcome in each arm. In particular, we maintain counters for successful and unsuccessful predictions or transmissions. In

[2]Beta(a, b) refers to the beta distribution whose probability density function is given by $p_{a,b}(x) \triangleq x^{a-1}(1-x)^{b-1}/B(a,b)$, $x \in [0, 1]$, where $B(a, b) \triangleq \Gamma(a)\Gamma(b)/\Gamma(a + b)$ and $\Gamma(\cdot)$ is the Gamma function.

each time slot, we generate posterior probabilities of motion prediction and wireless transmission for each arm independently. Then, we select the arm with the maximum product of prediction and transmission probabilities as well as its rate. Since Algorithm 2 has two-level feedback information, it yields better performance compared with Algorithm 1, as demonstrated via simulations in the next section.

IV. Numerical Results

In this section, we compare the regret performance between both Algorithm 1 with single feedback information and Algorithm 2 with two-level feedback information. We consider two different simulation setups, both with five available selected rates, as listed in TABLE I. In the simulations, we set the time horizon T to 10^4 time slots, and run 5000 experiments to make sure that the average regret is sufficiently accurate. The simulation results are shown in Fig. 1. From Fig. 1, we can observe that Algorithm 2 with two-level feedback information outperforms the traditional Thompson Sampling algorithm with single feedback information. The reason lies in that two-level information provides a much richer feedback on both prediction and transmission decisions than the single feedback on the successful/unsuccessful decisions.

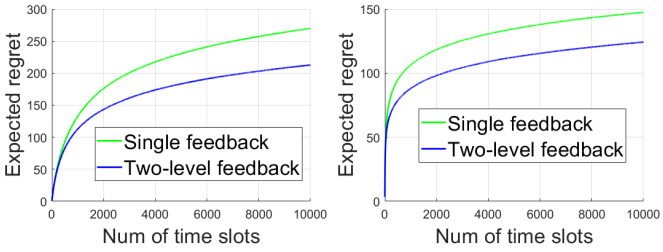
	arm1	arm2	arm3	arm4	arm5
Rate r_n	2	3	5	6	9
Prediction prob. α_n	0.1	0.3	0.5	0.65	0.9
Transmission prob. β_n	0.99	0.6	0.4	0.2	0.05
Average throughput	0.198	0.54	1	0.78	0.405

(a) First simulation setup

	arm1	arm2	arm3	arm4	arm5
Rate r_n	2	3	8	10	11
Prediction prob. α_n	0.01	0.08	0.8	0.88	0.95
Transmission prob. β_n	0.99	0.9	0.85	0.15	0.05
Average throughput	0.198	0.216	5.44	1.32	0.5225

(b) Second simulation setup

TABLE I: Simulation Parameters



(a) First simulation setup (b) Second simulation setup

Fig. 1: Regret Performance

V. Conclusion

In this paper, we considered the problem of adaptive rate selection for panoramic video streaming and formulated it as a multi-armed bandit problem with two-level feedback information, where both prediction and transmission outcomes of the selected arm are available after each play. Intuitively, a larger selected rate increases the probability of successful prediction but is at the cost of increasing the chance of unsuccessful transmission. Our goal was to appropriately determine the selected rate in each time slot with the goal of maximizing system throughput over a finite horizon. To this end, we proposed a modified Thompson Sampling algorithm efficiently leveraging the two-level feedback information and demonstrated that its performance is much better than its counterpart with single-feedback information.

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] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

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

[9] E. Kaufmann, N. Korda, and R. Munos, "Thompson sampling: An asymptotically optimal finite-time analysis," in *International conference on algorithmic learning theory*. Springer, 2012, pp. 199–213.

[10] S. Agrawal and N. Goyal, "Further optimal regret bounds for thompson sampling," in *Artificial intelligence and statistics*, 2013, pp. 99–107.

[11] A. Gopalan, S. Mannor, and Y. Mansour, "Thompson sampling for complex online problems," in *International Conference on Machine Learning*, 2014, pp. 100–108.

[12] T. Lattimore and C. Szepesvári, "Bandit algorithms," *preprint*, p. 28, 2018.