DOI: 10.3969/j.issn.1673-5188.2018.02.007
http://kns.cnki.net/kcms/detail/34.1294.TN.20180427.1001.004.html, published online April 27, 2018

Markov Based Rate Adaption Approach for Live Streaming over HTTP/2

XIE Lan[1], ZHANG Xinggong[1], HUANG Cheng[2], and DONG Zhenjiang[2]

(1. Institute of Computer Science and Technology, Peking University, Beijing 100080, China;
2. ZTE Corporation, Nanjing 210012, China)

Abstract

Dynamic adaptive streaming over HTTP (DASH) has been widely deployed. However, large latency in HTTP/1.1 cannot meet the requirements of live streaming. Data-pushing in HTTP/2 is emerging as a promising technology. For video live over HTTP/2, new challenges arise due to both low-delay and small buffer constraints. In this paper, we study the rate adaption problem over HTTP/2 with the aim to improve the quality of experience (QoE) of live streaming. To track the dynamic characteristics of the streaming system, a Markov-theoretical approach is employed. System variables are taken into account to describe the system state, by which the system transition probability is derived. Moreover, we design a dynamic reward function considering both the quality of user experience and dynamic system variables. Therefore, the rate adaption problem is formulated into a Markov decision based optimization problem and the best streaming policy is obtained. At last, the effectiveness of our proposed rate adaption scheme is demonstrated by numerous experiment results.

Keywords

DASH; live; rate adaption; Markov decision

1 Introduction

In recent years, dynamic adaptive streaming over HTTP (DASH) has been widely adopted for providing uninterrupted video streaming service to users with dynamic network conditions and heterogeneous devices [1], [2]. Contrary to the past Real-Time Transport Protocol/User Datagram Protocol (RTP/UDP), the use of HTTP over Transmission Control Protocol (TCP) greatly simplifies the traversal of firewalls and network address translators (NAT) which can be easily deployed within content delivery networks (CDN). Moreover, the rate adaption scheme is one of the most essential components to improve the streaming quality. By far, many rate adaptation schemes have been designed for DASH, including bandwidth-based rate adaption schemes and buffer-based rate adaption schemes [3], [4]. Akhshabi et al. [1] compared rate adaption for three popular DASH clients: Netflix client, Microsoft Smooth Streaming [5], and Adobe Open Source Meda Framework (OSMF). The conclusion in [1] indicates that none of the rate adaptation is good enough.

On the other hand, to track the dynamic characteristics of streaming system, Markov theory has been shown to be effective [6]–[8]. Regarding the work by García's et al. [6], they use Stochastic Dynamic Programming (SDP) optimization to solve the rate adaption problem for DASH. In which, the cost function is designed to stable the buffer occupancy at a certain level, leading to frequent video bitrate fluctuations. In [9], rate selection is performed offline by a Markov Decision Process (MDP) assuming that the available bandwidth can be estimated using a transition matrix. Applying the model online, however, may result in inaccurate estimation due to unpredictable characteristics of network conditions. This work is further extended in [7] and [10].

However, most of existing works focus on Video on Demand (VoD) service, and they are not suitable for live streaming where low latency is required. Moreover, by adopting HTTP/2, a low end-to-end latency can be ensured because multiple video fragments can be pushed to clients by a single request [11], [12]. This has been demonstrated by Wei et al. [13] based on their multiple experiments on video streaming over HTTP/2. As for the rate adaptation in live streaming, some new challenges arise; for example, the low start-up delay is required, and the bandwidth variation cannot be smoothed by setting a large buffer.

In this paper, we study the rate adaptation problem for live streaming over HTTP/2. Similar to [6], the Markov theory is applied to analyze the dynamic characteristics of the system and the rate adaptation problem is formulated into an optimization problem. To track the dynamic characteristics of the streaming system, several system variables are used to describe the system state, including video rate, buffer occupancy, available bandwidth, playback deadline and download time for each segment, and then the system transition probability is derived. Moreover, a dynamic reward function is designed under three scenarios of buffer occupancy to meet the requirement of user experience. The experiments done by bandwidth trace have shown that our proposed algorithm can provide a smooth and high video rate while guaranteeing a continuous video playback.

The rest of the paper is organized as follows. Section 2 presents the overview of Markov Decision based rate adaptation. The system state is introduced and the state transition probabil-

This work was supported in part by China "973" Program under Grant No. 2014CB340303" and ZTE Industry-Academia-Research Cooperation Funds.

ity is derived in Section 3. In Section 4, the dynamic reward function is described. At last, we show experiment results in Section 5, and conclude the paper in Section 6.

2 Markov Based Rate Adaptation

In this paper, we propose a Markov based rate adaption approach. First, we define system state at stage k as u_k, which has taken into account several system variables. An action a_k is defined at stage k, denoting a specific bitrate that is assigned for segment $k+1$. After taking action a_k, the system state transfers from u_k to u_{k+1}, i.e.,

$$u_k = f(u_k, a_k). \tag{1}$$

Due to the stochastic nature of the system, it can be characterized in terms of conditional probability distribution among states, that is, $P(u_{k+1}|u_k, a_k)$ which is the transition probability from u_k to u_{k+1} under action a_k.

On the other hand, in order to evaluate how good an action is, a reward function is also designed and the reward of action a_k or state u_k is defined as $R(u_k, a_k)$. Then, the long-term reward can be written as:

$$V(u_k, a_k) = \sum_{\{u_{k+1}\}} P(u_{k+1}|u_k, a_k) \cdot (R(u_k, a_k) + \gamma V(u_{k+1}, a_{k+1})), \tag{2}$$

where $V(u_k, a_k)$ calculates the sum of the rewards of all the possible next state u_{k+1} and $0 < \lambda < 1$ is a future discount rate that controls how much effect future rewards have on the decision at the current stage.

The streaming policy π is a mapping between system state u_k and action a_k.Obviously, finding the optimal strategy policy $\pi^*(u_k)$ which can maximize (2) is the goal of MDP. Therefore, our rate adaptation task can be finally formulated as an optimization problem:

$$\pi^*(u_k) = \arg\max_{a_k} V(u_k, a_k). \tag{3}$$

The detailed definition and analysis of the MDP is formulated mathematically in the following section.

3 State Transition Probability

In this section, we introduce the system state in detail and its transition probability. At stage k, the k-th segment is pushed to the client. Different from VoD service, in a live streaming scenario, media segments are available only after they have been generated. To reveal this feature, we have considered the playback deadline and the arrival time of each segment, and the system state is defined as:

$$u_k = \{q_k, v_k, \hat{t}_k, t_k, bw_k\}, \tag{4}$$

with each parameter stands for buffer occupancy, video bitrate, playback deadline, actual arrival time, and available bandwidth respectively. Given state u_k, an a_k action is taken to select the video rate for the next segment, i.e., $v_{k+1} = a_k(u_k)$. Note, if the segment is not available at the server side, a wait action will be taken.

The buffer occupancy q_k denotes the buffer level when segment k is just completely downloaded, and it is measured in second. It increases when a segment is pushed from the server and descends when segments are consumed by playing. Therefore, the buffer occupancy evolution can be written as:

$$q_{k+1} = \max\left\{q_k + T_s - \frac{v_{k+1} \cdot T_s}{bw_{k+1}}, 0\right\}, \tag{5}$$

where T_s is the duration of one segment.

For parameters $\hat{t}_k$ and t_k, they are used to characterize the time attributes of each segment. Moreover, we define T_d as the start-up delay, the playback deadline of segment $k+1$ can be written as:

$$\hat{t}_{k+1} = (k+1) \cdot T_s + T_d. \tag{6}$$

On the other hand, in live streaming, a segment is pushed to the client as soon as the previous segment has been totally sent out. Therefore, the actual arrival time of segment k is determined by both the previous segment arrival time t_k and the transmission duration of segment k:

$$t_{k+1} = t_k + \frac{v_{k+1} \cdot T_s}{bw_{k+1}}. \tag{7}$$

Generally, it is difficult to estimate the statistic of the bandwidth accurately. However, it has been widely known that the Markov channel models are useful tools to describe the variations of bandwidth. Thus, we also apply the Markov model to describe the available bandwidth. According to the Markov property, the state at any time instance only depends on its previous state. Considering for taking an action under a certain state, all the factors in the next system state can be calculated using (5)–(7) except bandwidth itself. Therefore, given a previous state u_k and an action a_k, the transition probability of the MDP is related to the transition probability of bandwidth which can be given as:

$$P(u_{k+1}|u_k, a_k) = P(bw_{k+1}|bw_k). \tag{8}$$

4 Dynamic Reward Function

In this section, we propose a reward function to measure how good an action is. Previous works [6]–[8] pointed out several factors that have impacts on user experience. In this paper, four system factors have been considered, including video rate and its smoothness, buffer occupancy, and playback deadline constraint. The impact of these factors on user experience is denoted by R_1, R_2, R_3, R_4 respectively. Then, the reward for an action can be evaluated by

$$R(u_k, a_k) = aR_1 + bR_2 + cR_3 + dR_4, \tag{9}$$

where a, b, c, d are weights for each factors.

In the following, we will derive the four reward functions included in $R(u_k,a_k)$ under three scenarios, since the factors a user concerns are generally different under different streaming scenarios.

4.1 Scenario 1: $q_k < T_s$

When the buffer occupancy is lower than T_s, i.e., the segment duration, the risk of buffer underflow is high. In this case, a low bitrate is preferred so as to avoid playback freeze. From the video quality perspective, the reward value gained from the video rate obeys the logarithmic relationship and R_1 is defined as:

$$R_1 = \ln(v_k - B_{\min} + \varepsilon), \tag{10}$$

where $B_{\min}$ is the lowest available bitrate and ε is an arbitrary small positive number. On the other hand, rate switch among continuous segments will bring negative effect on user visual perception. Therefore, the smoothness of video rate on user experience R_2 can be presented as:

$$R_2 = \ln(|v_k - v_{k-1}| + \varepsilon). \tag{11}$$

At last, when $q_k < T_s$, more attention should be paid to avoid buffer underflow. One of the most effective method is to select the lowest bitrate. Therefore, we can simply define R_3 and R_4 as:

$$R_3 = -\infty, \tag{12}$$

$$R_4 = -\infty. \tag{13}$$

4.2 Scenario 2: $q_k > q_{high}$

We predefined a threshold q_{high} used to avoid buffer overflow. When the buffer occupancy is larger than q_{high}, the risk of buffer overflow is high and the risk of buffer underflow is low. In this case, a high rate is preferred. For the video rate and its smoothness on user experience, they are defined the same as Scenario 1. For buffer occupancy, we prefer to select a video rate that can drag the buffer occupancy to be no higher than q_{high}, and R_3 is written as:

$$R_3 = q_{high} - q_k. \tag{14}$$

At last, R_4 measures the difference between the download time and playback time. When $\hat{t}_k > t_k$, some video segments has been buffered. On the other hand, if $\hat{t}_k < t_k$ the buffer is empty and playback freeze happens. Therefore, we can define R_4 as:

$$R_4 = \begin{cases} \hat{t}_k - t_k, \hat{t}_k - t_k \geqslant T_s \\ -\infty, \hat{t}_k - t_k < T_s \end{cases}. \tag{15}$$

4.3 Scenario 3: $T_s \leqslant q_k \leqslant q_{high}$

When buffer occupancy satisfies $T_s \leqslant q_k \leqslant q_{high}$, i.e., the buffered video time is between one segment duration and the overflow threshold, the risk of both buffer overflow and underflow is low. The video rate and its smoothness on user experience are defined the same as in Scenarios 1 and 2. For buffer occupancy, we prefer to select a video rate that can keep the buffer occupancy stable in range of $[T_s, q_{high}]$ and R_3 is written as:

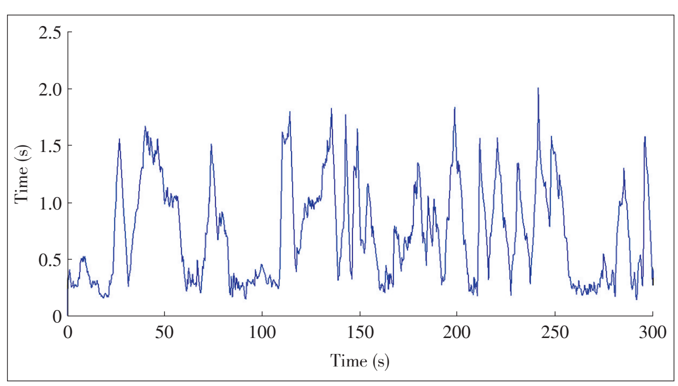
$$R_3 = \frac{T_s + q_{high}}{2}. \tag{16}$$

R_4 here is simply computed as the same as that in Scenario 2.

5 Experiments

In our experiment, same with Netflix, the server provides five different versions of video bitrates {300 kbit/s, 700 kbit/s, 1.5 Mbit/s, 2.5 Mbit/s, 3.5 Mbit/s}. Each version is an equal-length segment, with the length of 0.3 s. For the start-up delay, we set $T_d = 2$ s which is the length of seven segments. This setting can not only ensure a low startup delay, but also provide a continuous video playback as will be shown in the experiment results. For the buffer overflow threshold, we set $q_{high} = 1.8$ s, i.e., the length of six segments, which can be used to avoid that the buffered video time is higher than initial buffered video data so as to switch to a higher video rate. For performance comparison, we extend the previous SDP-based rate adaption scheme proposed in [6] to deliver live content using a regular request-response mode for requesting the video segment one by one. We also implement the bandwidth based K-Push scheme [13] where the client issues a request every K segments with the same rate. We evaluate the three rate adaptation schemes on real-world bandwidth trace. For our proposed method, the state is updated according to environment change and the best action has been chosen for deciding the video bitrate.

We first evaluate the difference between the playback deadline and download time of each segment in our scheme. The result is shown in **Fig. 1**. From the result we can find that when $T_d = 2$ s, the time difference for all segment is higher than zero, i.e. no playback freeze happens. On the other hand, whenever the time difference is high (approach to 1.8 s), it will quickly



▲**Figure 1. Time difference between the playback deadline and download time.**

decrease. This is because high time difference means the segment needs to wait for a long time before being played. Therefore, a high video rate can be selected.

At last, we compared the performance of all the three schemes. The results (**Fig. 2**) demonstrate that the video rate follows the principle of designing reward function well and that the rate decreases when buffer occupancy is low thus preventing playback freeze. Moreover, smoothness is focused if buffered video is adequate to keep continuous video playback. Compared with the SDP approach, our proposed scheme obtains a much smoother video rate. This is because in the SDP approach, the reward function is designed to stabilize the buffer occupancy without considering the smoothness of video rate well. From Fig. 2b we can find that the video rate is mainly switched between 1.5 Mbit/s and 2.5 Mbit/s in the SDP approach, and there are also a few segments whose rate are assigned to be 700 kbit/s or 300 kbit/s due to the low buffer occupancy. At last, when compared with the bandwidth based K-Push scheme, we can find that both have close performance on the smoothness of video rate. This is mainly because in the bandwidth based K-Push scheme, consecutive K segments are assigned the same video rate and bandwidth variations can be smoothed well. However, since video rate is switched every K segments, the bandwidth based K-Push scheme is insensitive to bandwidth variation and playback freeze happens as Fig. 2d shows.

5 Conclusions

In this paper, we have studied the rate adaption problem for live streaming over HTTP/2 by the Markov theory. To track the dynamic characteristics of the streaming system, we have defined a system state and several system variables are taken into account, including video rate, buffer occupancy, available bandwidth, playback deadline, and download time for each segment, and then the system transition probability is derived. We also have designed a dynamic reward function considering both the quality of user experience and dynamic system variables. Therefore, the rate adaption problem is formulated into a Markov decision based optimization problem and the best streaming policy is obtained. At last, the experiments by bandwidth trace have demonstrated the high effectiveness of our proposed rate adaption scheme.

References

[1] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http," in *Proc. 2nd Annual ACM Conference on Multimedia Systems*, San Jose, USA, 2011, pp. 157–168. doi: 10.1145/1943552.1943574.

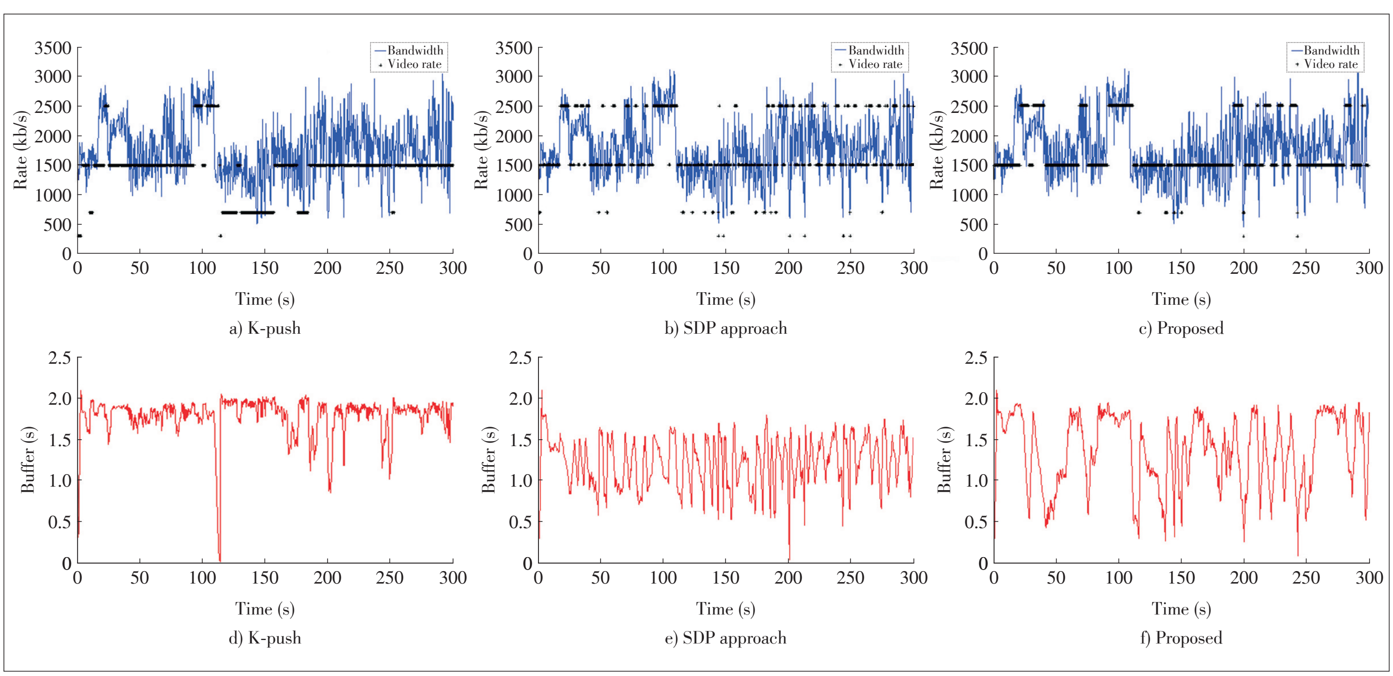
[2] T. Stockhammer, "Dynamic adaptive streaming over http-: standards and design principles," in *Proc. 2nd Annual ACM Conference on Multimedia Systems*, San Jose, USA, 2011, pp. 133–144. doi: 10.1145/1943552.1943572.

[3] C. Zhou, X. Zhang, L. Huo, and Z. Guo, "A control-theoretic approach to rate adaptation for dynamic http streaming," in *2012 IEEE Visual Communications and Image Processing*, San Diego, USA, 2012, pp. 1–6. doi: 10.1109/VCIP.2012.6410740.

[4] C. Zhou, C.-W. Lin, X. Zhang, and Z. Guo, "A control-theoretic approach to rate adaption for dash over multiple content distribution servers," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 4, pp. 681–694, Apr. 2014. doi: 10.1109/TCSVT.2013.2290580.

[5] A. Zambelli, "IIS smooth streaming technical overview," Microsoft Corporation, USA, vol. 3, 2009.

[6] S. Garcia, J. Cabrera, and N. Garcia, "Quality-control algorithm for adaptive streaming services over wireless channels," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 1, pp. 50–59, Feb. 2015. doi: 10.1109/JST-



▲Figure 2. Performance comparison. a)–c) are video bitrates of three rate adaption approaches; d)–f) are buffer occupancy of three rate adaption approaches.

SP.2014.2331912.
[7] M. Xing, S. Xiang, and L. Cai, "A real-time adaptive algorithm for video streaming over multiple wireless access networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, pp. 795–805, Apr. 2014. doi: 10.1109/JSAC.2014.140411.
[8] T. Andelin, V, Chetty, D. Harbaugh, S. Warnick, and D. Zappala, "Quality selection for dynamic adaptive streaming over http with scalable video coding," in *Proc. 3rd Multimedia Systems Conference*, Chapel Hill, USA, 2012, pp. 149–154. doi: 10.1145/2155555.2155580.
[9] S. Xiang, L. Cai, and J. Pan, "Adaptive scalable video streaming in wireless networks," in *Proc. 3rd Multimedia Systems Conference*, Chapel Hill, USA, 2012, pp. 167–172. doi: 10.1145/2155555.2155583.
[10] M. Xing, S. Xiang, and L. Cai, "Rate adaptation strategy for video streaming over multiple wireless access networks," in *Proc. IEEE Global Communications Conference*, Atlanta, USA, 2013, pp. 5745–5750.
[11] C. Mueller, S. Lederer, C. Timmerer, and H. Hellwagner, "Dynamic adaptive streaming over http/2.0," in *IEEE international Conference on Multimedia and Expo (ICME)*, San Jose, USA, 2013, pp. 1–6. doi: 10.1109/ICME.2013.6607498.
[12] T. C. Thang, H. T. Le, A. T. Pham, and Y. M. Ro, "An evaluation of bitrate adaptation methods for http live streaming," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, pp. 693–705, Apr. 2014. doi: 10.1109/JSAC.2014.140403.
[13] S. Wei and V. Swaminathan, "Low latency live streaming over http2.0," in *Proc. Network and Operating System Support on Digital Audio and Video Workshop*, Singapore, Singapore, 2014, pp. 37–37.

Manuscript received: 2017-01-01

Biographies

XIE Lan (xielan@pku.edu.cn) received the B.S. degree from Beijing University of Technology, China in 2015. She is currently a master degree student at the Institute of Computer Science and Technology, Peking University. Her research interests include dynamic HTTP streaming and VR video streaming. She received Beijing Outstanding Graduate Award in 2015.

ZHANG Xinggong (zhangxg@pku.edu.cn) received the Ph.D. degree from Peking University, China in 2011. He is currently a professor at the Institute of Computer Science and Technology, Peking University. His research interests include video conferencing, dynamic HTTP streaming, information-centric networking and VR video streaming. He received the First Prize of the Ministry of Education Science and Technology Progress Award in 2006 and the Second Prize of the National Science and Technology Award in 2007.

HUANG Cheng (huang.cheng5@zte.com.cn) received the M.S. degree from Southeast University, China in 2006. He is currently a senior system engineer and multimedia standard manager at ZTE Corporation. His research interest include video coding, storage, transport, and multimedia systems.

DONG Zhenjiang (dong.zhenjiang@zte.com.cn) received his M.S. degree in telecommunication from Harbin Institute of Technology, China in 1996. He is the deputy head of the Service Institute of ZTE Corporation and standing director of Chinese Association for Artificial Intelligence. His research interests include cloud computing, mobile internet, natural language processing, and multimedia analysis.