Analytical QoE Models for Bit-Rate Switching in Dynamic Adaptive Streaming Systems

Yuedong Xu, Yipeng Zhou, and Dah-Ming Chiu, *Fellow, IEEE*

Abstract—Video streaming service in wireless networks is increasingly using dynamic selection of video bit-rates to provide a high quality of user experience (QoE). The bit-rate switching mechanism, performed at client side, plays a key role in determining QoE metrics. In this paper, we present the first analytical framework to compute starvation probability of playout buffer, continuous playback time and mean video quality, given the bit-rate switching logics. Wireless channel is modeled as a continuous time Markov process, and playout buffer is modeled as a fluid queue with Markov modulated fluid arrival. We construct a set of ordinary differential equations (ODEs) to characterize the dynamics of starvation probability and expected continuous playback time with regard to buffer length, and simple models to analyze mean bit-rate for different bit-rate switching algorithms. Our framework is very general in that by adding appropriate parameters, it can be utilized to predict the QoE metrics of dynamic adaptive streaming with a variety of features: i) buffer-aware bit-rate switching ii) (im)patience of the user, and iii) receiver-side flow control.

Index Terms—Bit-rate switching, quality of experience, starvation, ordinary differential equations

1 Introduction

Video streaming now makes up 39 percent of all mobile traffic, and it grew 93 percent in the first half of 2011 as reported by Allot Communications [1]. Providing a high-quality user experience has become a very challenging task for both network operators and content providers. The main difficulty originates from heterogeneous and time-varying bandwidth of streaming users, especially in wireless networks. This motivates the shift from traditional streaming protocols to novel adaptive streaming technique, where dynamic adaptive streaming over HTTP (DASH) is a prominent application. DASH is a client-driven streaming technology on top of TCP/HTTP that does not introduce specially designed streaming architecture at server side. In DASH, a video is usually encoded into multiple files of different bit-rates. For each quality level, the video is divided into a number of segments lasting several seconds in terms of playback time. DASH allows a user to choose a high bit-rate during a streaming session if current throughput is large, and a low one otherwise. The small video segments of different qualities are aligned seamlessly to guarantee fluent playback (i.e., no abrupt jump of video frames).

The idea of DASH has been applied to commercial products including Microsoft Smooth Streaming, Akamai HD, Adobe OSMF, Netflix, etc. Their key feature is *bit-rate switching* algorithm, i.e., the way that receiver adjusts the quality of requested video segments. Due to commercial reasons, the above DASH systems are closed, leaving their switching behavior unknown to research community. Akhshabi et al. in [10] conducted measurements on several DASH systems by altering link bandwidth. They showed that the bit-rate switching in some systems were more aggressive than the others. Whereas the measurement study is done case by case, and is unable to unravel the performance tradeoff of more general bit-rate selection strategies. Furthermore, external measurements are confined by default parameters of commercial systems, thus lack of flexibility.

A theoretical understanding of the performance of bit-rate switching is very importance in designing better switching algorithms. A streaming user needs to predict the QoE before selecting a switching algorithm during the playback process. Intuitively, choosing an algorithm with higher average bit-rate (interchangeable with playback rate) may cause the hazard of playback interruptions. This hazard becomes more serious when the playout buffer length is small upon the time of decision making. However, these intuitions have not been scrutinized in the literature. Their precise qualitative and quantitative properties remain unclear. The biggest obstacle is how the key features of adaptive streaming can be subtracted from technical details. To this goal, we dissect an adaptive streaming system into three modules, the arrival process, the bit-rate switching algorithm and the QoE metrics. The arrival process is regarded as the external input that can be measured at the client side. The switching algorithm determines the QoE metrics.

Our context is the delivery of an adaptive streaming session over a wireless link that experiences slow channel fading or is shared by multiple flows. We model the channel rate (or throughput) variation as a finite state, continuous time Markov process, similar to previous works on DASH [7], [8], [9]. *Without loss of generality, the worst channel rate is assumed to be less than the lowest video bit-rate*. We consider two representative strategies in which one is called

- *Y. Xu is with the School of Information Science and Technology, Fudan University, Shanghai, P.R. China. E-mail: ydxu@fudan.edu.cn.*
- *Y. Zhou is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, P.R. China. E-mail: ypzhou@szu.edu.cn.*
- *D.-M. Chiu is with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong. E-mail: dmchiu@ie.cuhk.edu.hk.*

Manuscript received 11 Dec. 2012; revised 1 Nov. 2013; accepted 7 Jan. 2014. Date of publication 24 Feb. 2014; date of current version 27 Oct. 2014. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMC.2014.2307323

"buffer-oblivious" (BO) switching and the other is called "buffer-aware" (BA) switching. BO switching selects bit-rate only based on the measured channel rate. In general, if the lowest bit-rate is larger than a channel rate, it is the default bit-rate for this channel state. Otherwise, the user may request the highest bit-rate that can be supported by the current channel rate. BA switching introduces a value named "SWITCH_DOWN threshold". When the playout buffer (interchangeable with queue) length is below this threshold, BA behaves the same as the corresponding BO strategy. When it is above the threshold, the requested bit-rate may ramp up to a higher level, but not switch to a lower level. The SWITCH_DOWN action is taken when the queue length is below SWITCH_DOWN threshold. BO and BA strategies reflect key features of bit-rate switching of existing DASH systems according to measurement studies [6], [10]. Video streaming service possesses a specific feature related to user behavior. The user might terminate the streaming session before the video is completely played. In order to avoid the wasted prefetching, streaming vendors usually deploy receiver side flow control scheme. If the downloading progress is ahead of the playback progress than a certain threshold, user pauses the request of new video chunks. User engagement and flow control are parameterized via the "watching time" and the "flow control threshold" respectively. The buffer-oblivious switching with flow control is abbreviated as BOFC.

In this paper, we answer a fundamental question concerning the performance of client-driven adaptive streaming service. *Given a bit-rate switching algorithm and initial conditions (e.g., buffer length and channel state), how can a user predict the expected QoE metrics including starvation probability, continuous playback time and mean video bit-rate?* Knowing this answer enables the user to make wise switching from a set of candidate switching algorithms. Our second purpose is to reveal how the QoE metrics are influenced by the user's watch time, and the flow control and the SWITCH_DOWN thresholds. We propose a unified framework to efficiently compute the QoE metrics. The playout buffer is modeled as a Markov modulated fluid queue. Inspired by ruin analysis in actuarial science [21], we formulate a set of ODEs with regard to (w.r.t.) the buffer length for BO switching. These ODEs are solved by using Laplace transform (LT), and the solutions are interpreted as the starvation probability and the mean continuous playback time. This method is further extended to analyze BOFC and BA switchings. To the best of our knowledge, we present the *first* analytical framework for dynamic adaptive streaming. Our study is useful not only in providing qualitative and quantitative understandings of bit-rate switching strategies, but also in designing better switching algorithms.

We hereby summarize our major qualitative observations of bit-rate switching logics from an engineering point of view.

-*Starvation probability*. It decreases exponentially w.r.t. the initial buffer length. The starvation probabilities of BO, BA and BOFC are sorted in an ascending order.

-*Asymptotic starvation probability (long enough watch time)*. For BO switching, it is less than 1 if mean arrival rate is larger than mean playback bit-rate of a strategy, and is 1 otherwise. For BOFC switching, it is always 1. For BA switching, it is less than 1 if mean arrival rate is greater than the highest achievable bit-rate in the strategy, and is always 1 otherwise. In a word, even if mean arrival rate is greater than mean playback rate, buffer emptiness can happen certainly in BOFC and BA switchings.

-*Continuous playback period*. It is a metric to reflect the frequency of starvations when mean arrival rate is below mean playback rate of a strategy. For BO switching, it is proportional to initial queue length. The proportional coefficient is the same at all different channel states. The continuous playback times of BO, BA and BOFC strategies are sorted in a descending order.

-*Mean video bit-rate*. The impact of flow control on mean video bit-rate is uncertain. BA switching has a larger mean bit-rate than BO and BOFC strategies.

-*Downloaded but unwatched data in the absence of flow control*. It increases with the increase of user's watch time first, and then decreases as the watch time further increases. The maximum wasted prefetching happens when the user terminates video playback at the instant that the whole video is just downloaded.

The remainder of this paper is organized as follows: In Section 2, we describe the motivation of this work and the mathematical model. Section 3 computes QoE metrics of BO switching. Section 4 computes QoE metrics of BA switching. We provide explicit results for a two-state channel process in Section 5. In Section 6, we compare models with simulations. We discuss potential limitations in Section 7. Section 8 describes the related works. Section 9 concludes this paper.

2 Motivation and Model

In this section, we describe the bit-rate switching mechanisms as well as the QoE metrics, and present the mathematical model.

2.1 Buffer-Oblivious Switching

The basic idea of DASH-like adaptive streaming is to enable end user to select chunk bit-rate dynamically that matches his bandwidth during the playback process. We use the terms SWITCH_UP and SWITCH_DOWN to define actions of increasing or decreasing the requested bit-rate respectively. As one can imagine, bit-rate selection strategy is a set of IF-THEN decisions depending on bandwidth variation and buffer length. Though intuitively simple, the bit-rate switching is difficult to design and to analyze. There are three main reasons. First, the bit-rate switching introduces new QoE metrics. Second, the switching behavior is coupled with the model of bandwidth variation. Third, the bit-rate switching has some very heuristic realizations that are difficult to model.

In this work, we present the first theoretical study on the performance of bit-rate switching. To facilitate the analysis, we assume that bandwidth estimation is accurate. This assumption is confirmed by a recent study in [2] that uses a machine-learning based TCP throughput prediction algorithm. More discussions on the estimation of wireless channel model can be found in Section 7. We next commence our study with a set of important terms.

Definition 1 (Bit-rate Switching Strategy). *A bit-rate switching strategy $\mathcal{S}$ consists of I channel rate and bit-rate pairs, given that there are I channel rate levels.*

A video is encoded into multiple versions of different bit-rates by content provider. In [2], each video has 51 bit-rates ranging from 100 Kbps to 5.1 Mbps. Though more bit-rate levels enable more fine-grained switching, the operational cost is greatly increased and the scalability is impaired. De Cicco et al. in [6] measure Akamai's service using five video levels from 300 Kbps (320×180p) to 3.5 Mbps (1,280×720p) in wired Internet. In addition, different bit-rate sets are used in different measurement studies. Here, we consider five bit-rates in the set {240, 360, 480, 600, 720} Kbps. The bit-rates at the lower end were especially chosen for small handhold devices on mobile networks (close to those in [11]). We use levels from 1 to 5 to denote the bit-rates in the ascending order.

Here, we consider a representative scheme that sheds light on how the commercial systems work.

Definition 2 (Buffer-Oblivious Switching). *The SWITCH_UP and SWITCH_DOWN actions are triggered by channel rate variation.*

-An illustration. The channel rate of a mobile user is time varying in the set {150, 300, 500, 700} Kbps. The user can choose all the possible bit-rate levels for every channel rate. However, this increases the complexity for the user to make the switching decision. A conventional wisdom is to exclude infeasible strategies and then confine to a small set of candidate strategies. As a commonly adopted rule, the selected bit-rate cannot be larger than the throughput if it is not the lowest bit-rate. Given the sets of bit-rates and channel rates, we want to compare two heuristic strategy profiles: $\mathcal{S}_1$ = (1, 1, 3, 4) and $\mathcal{S}_2$ = (1, 1, 2, 4). Here, $\mathcal{S}_1$ selects 480 Kbps video bit-rate and $\mathcal{S}_2$ is more conservative to select 360 Kbps when the channel rate is 500 Kbps. As a consequence, $\mathcal{S}_1$ may have a higher video bit-rate, but risks a higher probability of playback interruption compared with $\mathcal{S}_2$. If the user can predict the possible outcomes of $\mathcal{S}_1$ and $\mathcal{S}_2$, he will choose the appropriate strategy to optimize his QoE at each time point of requesting video chunks.

2.2 QoE Metrics and Our Motivation

The QoE of adaptive streaming introduces new features compared with non-adaptive streaming. According to [11] (with certain modifications), the QoE metrics of adaptive streaming include

- *Starvation probability*. Denoting the probability that a streaming user sees frozen images.

- *Average bit-rate*. Denoting the mean video quality over the entire session.

- *Bit-rate stability*. Describing the jittering of video quality during the entire session.

- *Start-up delay*. Denoting the waiting duration between the time that the user requests streaming service and the time that media player starts to play.

When starvation event happens for sure, the starvation probability (i.e., equal to 1) is insufficient to capture the severity of QoE degradation. Hence, in addition to [11], we define a new metric as the following:

- *Continuous playback time*. Denoting the expected playback time between two consecutive starvations. If it is small, the starvation events happen more frequently.

Among these five QoE metrics, the starvation behaviors including the starvation probability and the continuous playback time remain to be the most annoying factors. The start-up delay becomes relatively less important because the user can choose the lowest video bit-rate at the initial stage so as to greatly reduce the waiting time. The analysis of start-up delay can be done in the same way as the continuous playback time. The former measures the duration that buffer length increases from 0 to a certain value. The latter depicts the duration that the buffer length reduces from the initial value to 0. The bit-rate instability refers to a phenomenon that serious bit-rate fluctuation harms the user perceived video quality. It mainly arises in the situations that bandwidth estimation does not work (e.g., with very fast changing bandwidth [2] and with several competing dynamic adaptive streaming sessions [3]). Here, we are looking into a single adaptive streaming session in a wireless channel whose channel rate varies at a large time scale. The frequency of bit-rate switching is decided by the stationary channel fading process. Hence, we focus on the modeling of starvation behavior and average video bit-rate.

Today, vendors of DASH systems keep their bit-rate switching schemes as their proprieties. The research community is not clear how user perceived video quality is influenced by bit-rate switching algorithm, initial prefetching, user behavior, and configuration of thresholds. This motivates our study to develop an analytical framework for a better understanding of the above features under channel variation.

2.3 A Unified Queueing Model

We consider a wireless channel whose channel rate between media server and streaming user varies in the set $\mathcal{I} = \{1, \ldots, I\}$. The channel rate variation is modeled as a finite state Markov (FSM) process similar to recent works on DASH improvement [7], [8], [9] and a number of works in wireless networking. The Markov chain is irreducible. In other words, the channel rate is possible to get to any state from any state during the streaming session. The transition rate from state i to j is denoted as α_{ij}. Let $\alpha_i = \sum_{j \neq i} \alpha_{ij}$. We assume that the channel rate at state i is deterministic, denoted by a_i in bits per second (bps) for all $i \in \mathcal{I}$. The channel rate changes like a step function over time. In $\mathcal{I}$, the channel rate a_i has $a_1 < \cdots < a_i \cdots < a_I$. We denote by $\mathcal{R} = \{1, \ldots, L\}$ the set of bit-rate levels of a video at the server. The bit-rate of level l streaming is denoted by r_l in bps that has $r_1 < \cdots < r_l \cdots < r_L$, $(l \in \mathcal{R})$.

We begin with the model of BO switching. BOFC and BA switchings are presented later on. Let l_i be the default bit-rate level at the ith channel state. In BO switching, r_{l_i} is a single value in $\mathcal{R}$. Thus, a BO strategy is expressed as $\mathcal{S} = (l_1, \ldots, l_I)$. The starvation of playout buffer may happen at state i that has $a_i < r_{l_i}$. For analytical convenience, the arrival rate is scaled by the video bit-rate. Let $b_i^{l_i} := a_i / r_{l_i}$ be the scaled arrival rate given the channel rate a_i and the bit-rate r_{l_i}. Define a new variable $c_i^{l_i} := b_i^{l_i} - 1$ that reflects the slope of buffer change at the ith channel state.

We define two sets, $\underline{\mathcal{I}}$ and $\overline{\mathcal{I}}$, to differentiate the channel states with negative and positive $c_i^{l_i} (\forall i \in \mathcal{I})$ respectively.

In what follows, we present a unified framework for BO switching. Denote by $Q(t)$ the length of playout buffer measured in seconds of playback at time t. At $t = 0$ (the time of starting video playback), we suppose that q seconds of content has been prefetched. Let $N_e(t)$ record the number of events of channel variation by time t. Denote by A_k the time that event k takes place with $A_0 = 0$. When the media player starts the playback, the queueing process $\{Q(t); Q(t) > 0, t \geq 0\}$ is given by:

$$Q(t) = q - t + \sum_{k=1}^{N_e(t)} b_{I_k}^{l_{I_k}} (A_k - A_{k-1}) + b_{I_{N_e(t)}}^{l_{I_{N_e(t)}}} (t - A_{N_e(t)}),$$

if time axis starts at the instant of video playback.

Define $\tau = \inf\{t \geq 0 | Q(t) < 0\}$ as the time of observing empty buffer. Let $U_i(q) := E[\tau | \tau < \infty, I(0) = i, Q(0) = q]$ be the mean continuous playback time if the initial channel state is i and the initial buffer length is q. When starvation event happens for sure, $U_i(q)$ is an important measure for the severity of starvations. A small $U_i(q)$ means that the starvation events happen frequently.

We define $V_i^{l_i}(q, t)$ to be the expected starvation probability before time t, given the initial channel state i and the initial queue length q. Then, there has

$$V_i^{l_i}(q, t) = E[\mathbf{1}_{\tau(q) \leq t} | I(0) = i, Q(0) = q].$$

As t approaches ∞, $V_i^{l_i}(q, \infty)$ represents the asymptotic starvation probability, also called absolute starvation probability. Let T_w be the random variable of watch time that follows an exponential distribution with mean $1/\theta$. For simplicity, we denote $V_i^{l_i}(q)$ by

$$V_i^{l_i}(q) = E[\mathbf{1}_{\tau(q) \leq T_w} | I(0) = i, Q(0) = q]. \quad (1)$$

3 Buffer-Oblivious Bit-Rate Switching

In this section, we present theoretical models for the starvation probability, the mean continuous playback time and the average video bit-rate.

3.1 Computing Starvation-Related Metrics

Our purpose is to compute the probability of empty buffer where the channel rate is governed by the external Markov process. At state i, the queue length changes in an infinitesimal slot h by $Q(t + h) = Q(t) + c_i^{l_i} h, \ \forall i \in \mathcal{I}$. In the slot [0, h], the downloaded video content is $a_i h / r_l$ seconds if the user does not terminate the playback. Then, the probability that the user continues to watch video in $[0, h]$ is given by $e^{-\delta_i h}$, where $\delta_i^{l_i} := \theta a_i / r_l$.

In BO switching, the requested bit-rate level l_i is unique at state i. Hence, $V_i^{l_i}(q)$ is reduced to the notation $V_i(q)$. Similarly, we use b_i (resp. c_i, δ_i) to replace $b_i^{l_i}$ (resp. $c_i^{l_i}$, $\delta_i^{l_i}$). If we take a snapshot of the system, four events may occur at $[0, h]$: (1) no change of state; (2) change of state; (3) departure of user; (4) occurrence of more than one event. Conditioned on these events, the dynamics of $V_i(q)$ is obtained by

$$V_i(q) = (1 - \alpha_i h) V_i(q + c_i h) e^{-\delta_i h} + \sum_{j \neq i} \alpha_{ij} h e^{-\delta_i h} V_j(q + c_i h) + o(h). \quad (2)$$

Since $e^{-\delta_i h} = 1 - \delta_i h + o(h)$ as $h \to 0$, Eq. (2) yields

$$V_i(q) = \left(1 - \alpha_i h - \delta_i^i h\right) V_i(q + c_i h) + \sum_{j \neq i} \alpha_{ij} h V_j(q + c_i h) + o(h). \quad (3)$$

As $h \to 0$, $o(h)/h \to 0$, the following ODEs hold:

$$c_i \dot{V}_i(q) = (\alpha_i + \delta_i) V_i(q) - \sum_{j \neq i} \alpha_{ij} V_j(q), \quad \forall i \in \mathcal{I}, \quad (4)$$

where $\dot{V}_i$ is the derivative of V_i over q.

User impatience. Suppose that the user watches an exponentially distributed time with the mean $1/\theta$. We take Laplace transform over Eq. (4). Let $\hat{V}_i(s)$ be the LT of $V_i(q)$. There has

$$c_i s \hat{V}_i(s) = (\alpha_i + \delta_i) \hat{V}_i(s) - \sum_{j \neq i} \alpha_{ij} \hat{V}_j(s) + c_i V_i(0). \quad (5)$$

The constant $V_i(0)$ is interpreted as the starvation probability with no initially prefetched content. Obviously, $V_i(0) := 1 \ \forall i \in \underline{\mathcal{I}}$ and $0 \leq V_i(0) < 1 \ \forall i \in \overline{\mathcal{I}}$ hold in our setting. Define a Matrix $\mathbf{M}_V$ that has

$$\mathbf{M}_V(i, j) := \begin{cases} -(\delta_i + \alpha_i)/c_i, & \text{if } i = j; \\ \alpha_{ij}/c_i, & \text{otherwise}. \end{cases} \quad (6)$$

Define a vector $\mathbf{V}(0) := \{V_1(0), \cdots, V_I(0)\}'$. Then Eq. (4) can be rewritten in a matrix form

$$\hat{\mathbf{V}}(s) = (s\mathbf{I} + \mathbf{M}_V)^{-1} \cdot \mathbf{V}(0). \quad (7)$$

The starvation probability vector can be derived by taking inverse LT over Eq. (7)

$$\mathbf{V}(q) = \mathcal{L}^{-1}\{(s\mathbf{I} + \mathbf{M}_V)^{-1} \cdot \mathbf{V}(0)\}. \quad (8)$$

The solution to $\mathbf{V}(q)$ needs the knowledge of the unknowns $\mathbf{V}(0)$.

Theorem 1. *Let* $\mathrm{adj}(sI + \mathbf{M}_V)$ *be the adjugate matrix of* $sI + \mathbf{M}_V$. *The starvation probabilities* $V_i(0)$ *are solved by*

- $V_i(0) = 1$ *for all* $i \in \underline{\mathcal{I}}$;
- $\mathrm{adj}(sI + \mathbf{M}_V) \cdot \mathbf{V}(0)|_{s=\xi_k} = 0$ *where* $\{\xi_k\}$ *is the* kth *positive root of the determinant* $|sI + \mathbf{M}_V|$.

Proof. Please refer to the supplementary file, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TMC.2014.2307323. □

To obtain starvation probabilities, we need to compute the roots of a polynomial determinant, the inverse matrix in Eq. (7), and the simple inverse LT in (8). Due to a small number of channel states, the computational complexity is very low.

We next present a set of propositions that shed light on the important properties of $V_i(q)$. According to Theorem 1, the right hand of Eq. (7) does not contain the factors of non-negative roots. Hence, by taking inverse LT, $V_i(q)$ only

consists of the exponential terms with negative exponents. Formally, we have the following propositions:

Proposition 1. *Let $\xi_1, \ldots, \xi_K$ be the negative roots of the determinant $|sI + \mathbf{M}_V| = 0$. Let $v_{i,1}, \ldots, v_{i,K}$ be the coefficients calculated from the inverse LT $\mathcal{L}^{-1}\{\hat{\mathbf{V}}(s)\}$. Then, the starvation probability $V_i(q)$ is expressed in the form $V_i(q) = \sum_{k=1}^{K} v_{i,k} \cdot \exp(\xi_k q), \forall i \in \mathcal{I}$.*

Proposition 2. *When the watching time is long enough (i.e., $\theta \to 0$), $V_i(q), \forall q < \infty, i \in \mathcal{I}$ is 1 if the average arrival rate is less than the average bit-rate (i.e., $\sum_i a_i \pi_i < \sum_i r_i \pi_i$).*

Remark 1. Proposition 1 manifests that the starvation probability decreases exponentially with the increase of initial buffer length. Hence, the user can adopt a more aggressive bit-rate switching algorithm if the buffer length is large, and a more conservative one if otherwise. Proposition 2 shows that if mean arrival rate is less than mean playback rate, the asymptotic starvation probability does not depend on the current buffer length and the current channel state.

Time till starvation. If the watch time is long enough (i.e., $\theta \to 0$), $V_i(q)$ equals to 1 when mean arrival rate is less than mean playback rate of $\mathcal{S}$. Then, $V_i(q)$ is insufficient to reflect the severity of starvations. We therefore introduce $U_i(q)$ to denote the mean playback time between two consecutive starvations. $U_i(q)$ reflects the frequency of discontinuous playbacks. Similarly, we derive the ODEs for $U_i(q)$. In slot $[0, h]$, the stored video length changes from q to $q + c_i h$. The expected time till starvation increases by h. This yields the dynamics of $U_i(q)$ as

$$U_i(q) = (1 - \alpha_i h)(h + U_i(q + c_i h)) + \sum_{j \neq i} \alpha_{ij} h (h + U_j(q + c_i h)) + o(h), \forall i \in \mathcal{I}. \quad (9)$$

By letting $h \to 0$, we obtain another set of ODEs

$$c_i \dot{U}_i(q) = \alpha_i U_i(q) - \sum_{j \neq i} \alpha_{ij} U_j(q) - 1, \quad \forall i \in \mathcal{I}. \quad (10)$$

Similarly, we take LT over Eq. (10) and obtain $\forall i \in \mathcal{I}$

$$c_i s \hat{U}_i(s) = \alpha_i \hat{U}_i(s) - \sum_{j \neq i} \alpha_{ij} \hat{U}_j(s) + c_i U_i(0) - 1/s. \quad (11)$$

The inverse LT in the matrix form is expressed as

$$\mathbf{U}(q) = \mathcal{L}^{-1}\{(s\mathbf{I} + \mathbf{M}_V)^{-1} \cdot (\mathbf{U}(0) - \{1/(c_i s)\})\} \quad (12)$$

with $\delta_i = 0$. Note that the boundary conditions of Eq. (10) are different from those in Theorem 1. When $q = 0$ and $c_i < 0$, the buffer emptiness happens at $t = 0$. When q is infinitely large, $E_i[\tau]$ also approaches infinity, which cannot serve as the boundary condition. According to the general result of G/G/1 queue and ruin theory (see Theorem 3.1 in [19]), the busy period exhibits the following asymptotic property,

$$\lim_{q \to \infty} \frac{U_i(q)}{q} = \frac{1}{1 - \sum_{i=1}^{I} \pi_i b_i}, \quad \forall i \in \mathcal{I}. \quad (13)$$

This implies that the continuous playback duration has a linear growth rate w.r.t. the prefetched contents in the playout buffer. Hence, $\hat{\mathbf{U}}(s)$ must be 0 if s is set to the positive root of the determinant $|sI + \mathbf{M}_V|$ (with $\delta_i = 0$). Otherwise, $U_i(q)$ is an exponential function of q with positive exponents. The solution to $\mathbf{U}(0)$ is derived using the similar method as Theorem 1 except that

- $U_i(0) = 0$ for all $i \in \underline{\mathcal{I}}$;
- $\mathrm{adj}(sI + \mathbf{M}_V) \cdot (U(0) - \frac{1}{c_i s})|_{s=\xi_k} = 0$ where $\{\xi_k\}$ is the k^{th} positive root of the determinant $|sI + \mathbf{M}_V|$.

Remark 2. The mean continuous playback time of BO switching strategy is proportional to the initial buffer length when the mean arrival rate is below the mean bit-rate. Eq. (13) shows that the slop of $U_i(q)$ is the same no matter what the initial channel state is.

3.2 Impact of Receiver-Side Flow Control

Denote by ϕ_a the expected buffer filling level, which is the difference between the current downloading and the playback progresses. We define Phase-A to be the scenario $q < \phi_a$, and Phase-B to be the scenario $q \approx \phi_a$. When ϕ_a is reached, the user stops requesting a new video chunk, thus keeping buffer length around ϕ_a. This is to say, the arrival rate at state i is $\min\{a_i, r_i\}$. The user departure rate is computed as $\delta_i = \theta$ in Phase-B.

Recall that $\underline{\mathcal{I}}$ (resp. $\overline{\mathcal{I}}$) is the set of states with $a_i < r_i$ (resp. $a_i > r_i$). The streaming user switches from Phase-B to A as soon as the bandwidth changes to any state in $\underline{\mathcal{I}}$. Denote by $\psi_{ij} (i \in \overline{\mathcal{I}}, j \in \underline{\mathcal{I}})$ the probability of phase switching from B to A at state j, given the initial channel state i at the time of entering Phase-B. According to the properties of embedded Markov process, there exists

$$\psi_{ij} = \frac{\alpha_{ij}}{\alpha_i + \theta} + \frac{\sum_{k \in \overline{\mathcal{I}}, k \neq i} \alpha_{ik} \psi_{kj}}{\alpha_i + \theta}, \quad \forall i \in \overline{\mathcal{I}}. \quad (14)$$

The above set of linear equations yield ψ_{ij}. Then, the starvation probability $V_i(\phi_a)$ is obtained by

$$V_i(\phi_a) = \sum_{j \in \underline{\mathcal{I}}} \psi_{ij} V_j(\phi_a), \quad \forall i \in \overline{\mathcal{I}}. \quad (15)$$

In Phase-A, the solution of $V_i(q)$ is also obtained from Eq. (8). However, the right-side boundary conditions in Theorem 1 do not hold any more because q cannot be larger than ϕ_a. Eq. (8) gives rise to

$$\mathbf{V}(\phi_a) = \mathcal{L}^{-1}\{(s\mathbf{I} + \mathbf{M}_V)^{-1}\}|_{q=\phi_a} \cdot \mathbf{V}(0). \quad (16)$$

Next, we will show how the starvation probabilities are derived in the presence of receiver-side flow control. According to Eq. (16), we express $V_i(\phi_a)$ as functions of $V_i(0)$. There are I linear equations where $V_i(\phi_a)$ for $i \in \mathcal{I}$ and $V_i(0)$ for $i \in \overline{\mathcal{I}}$ are unknowns. According to Eq. (15), $V_i(\phi_a)$ for $i \in \overline{\mathcal{I}}$ are substituted by $V_i(\phi_a)$ for $i \in \underline{\mathcal{I}}$. Hence, there are I unknowns remained in a set of I linear equations.

Solving $V_i(\phi_a)$ and $V_i(0)$ is not intuitive. Though the above analysis gives rise to a set of I linear equations with I unknowns. Some coefficients of these unknowns in the linear equations are extraordinarily large. As a result, the linear equations built from Eqs. (15) and (16) are ill-conditioned such that they cannot be solved directly. Here,

we present a simple heuristic approach to handle this difficulty via two steps. First, we re-arrange the linear equations. The starvation probability $V_i(\phi_a)$ in Eq. (16) contains the exponential terms $\exp(\xi_k\phi_a)$, $k=1,\ldots,I$. Here, $\{\xi_k\}$ are ranked in an increasing order. When $\exp(\xi_k\phi_a)$ is large enough (our default value is 10^5), the coefficient in front of $\exp(\xi_k\phi_a)$ should be small enough because the starvation probability is only in the small range [0, 1]. Hence, we let this coefficient be 0. It contains the unknowns in the original set of linear equations. By letting this coefficient be 0, we obtain a new linear equation with the same set of unknowns. We suppose that there are l roots with large enough $\exp(\xi_k\phi_a)$ for $I-l < k \leq I$. Then, we obtain a set of l new linear equations containing the unknown variables. Note that for each operation, we select a different equation from the original set of linear equations. Secondly, we select $I-l$ equations from the original set of linear equations built from Eqs. (15) and (16) after eliminating the largest l items $\exp(\xi_k\phi_a)$ for $I-l < k \leq I$. Now we obtain a new set of I linear equations with the unknowns $V_i(0)$ for $i \in \overline{\mathcal{I}}$ and $V_i(\phi_a)$ for $i \in \underline{\mathcal{I}}$ that can be easily solved.

We continue to provide the asymptotic starvation probability of BOFC switching.

Proposition 3. *The starvation probability of BOFC switching is 1 if the watch time is infinite and there is at least one state $i \in \mathcal{I}$ with $c_i < 0$.*

In contrast to Proposition 2, the starvation event can always happen in BOFC switching even if the mean arrival rate is greater than the mean bit-rate of a strategy. This is because the queue length is always finite (not growing infinitely as $t \to \infty$) in BOFC switching.

We next proceed to compute the expected time till starvation. The dynamics of $U_i(q)$ in Phase-A follows Eq. (9). In Phase-B, $U_i(\phi_a)$ changes at $[0,h]$ by

$$U_i(\phi_a)=(1-\alpha_i h)(h+U_i(\phi_a))+\sum_{j\in\mathcal{I},j\neq i}\alpha_{ij}(h+U_j(\phi_a))+o(h). \tag{17}$$

Taking the limit $h \to 0$ on the right side, we obtain

$$\alpha_i U_i(\phi_a)-\sum_{j\neq i}\alpha_{ij}U_j(\phi_a)=1, \quad \forall i \in \overline{\mathcal{I}}. \tag{18}$$

Here, Eq. (18) consists of the right-side boundary conditions. Taking the inverse LT over Eq. (12), we obtain the expressions of $U_i(\phi_a)$ with unknowns $U_j(0)$ for $j \in \overline{\mathcal{I}}$. Combined with the linear equations in Eq. (18), we can solve $U_i(0)$ for all $i \in \mathcal{I}$. Then, the starvation duration with arbitrary initial state $q \in [0,\phi_b]$ can be obtained accordingly. Note that we still meet the ill-conditioned linear functions when solving $U_i(0)$. We adopt the same technique as that in computing $V_i(0)$.

We next summarize the starvation behaviors of BO and BOFC switching in the following proposition.

Proposition 4. *The starvation probability and the mean continuous playback time satisfy: $V_i(q)|_{\phi_a\to\infty} \leq V_i(q)|_{\phi_a<\infty}$ and $U_i(q)|_{\phi_a\to\infty} \geq U_i(q)|_{\phi_a<\infty}$.*

Remark 3. The implications of Proposition 4 are as follows. The buffer length of BO switching increases continuously as long as the arrival rate is greater than the playback rate. When the channel rate reduces to the lowest level in the future, there are more packets in the buffer. In BOFC switching, the user cannot take the opportunity of good channel conditions to download more data. Because of fewer packets stored, flow control leads to a larger starvation probability or a smaller continuous playback time.

3.3 Comparing Average Bit-Rates w/o Flow Control

The average bit-rate is an important metric to depict user satisfaction. However, it is difficult to analyze over the entire streaming session. We need to consider the obscure low bit-rate requests in the beginning of streaming and after each starvation [10]. Therefore, we investigate the average bit-rate without considering the bit-rates at the short startup and rebuffering stages. We sample the average bit-rate at the steady state in order to mitigate the influence of the initial conditions (e.g., channel state, queue length). In addition, the discussion of the mean video bit-rate is meaningful only when it can be supported by the mean arrival rate.

Average bit-rate (BO). Recall that π_i is the stationary probability of being at state i. If we observe the system for an infinitesimal duration h, the downloaded video content is ha_i in bits, and is hb_i in seconds of video playback at state i. Then, the average bit-rate is calculated as the average amount of bits divided by the average video duration:

$$E[R_{bo}]=\sum_i \pi_i a_i \Big/ \sum_i \pi_i b_i. \tag{19}$$

Average Bit-rate (BOFC). When flow control is enforced, the downloaded video content is hr_{l_i} in bits, and is h in seconds for the duration h at state i. Thus, the average bit-rate of BOFC switching is expressed as

$$E[R_{bofc}]=\sum_i \pi_i r_i. \tag{20}$$

Remark 4. There is no explicit answer on whether the receiver side flow control can bring a higher average bit-rate or not, (i.e., which one is bigger in eqs. (19) and (20).)

Wasted prefetching. When a user terminates the playback before the end of the stream, the downloaded but not watched data is regarded as a "waste". The original purpose of flow control is to avoid the excessive downloading of streaming users. It is important to quantify how much traffic can be "saved" using flow control. We suppose that mean channel rate is larger than mean playback rate in strategy $\mathcal{S}$ (otherwise the wasted data will be very small). Similar to the preceding analysis, we concentrate on the unwatched video data at the steady state of channel process. Let $1/\theta$ be the video length. We reuse T_w to denote the watch time, i.e., $T_w \leq 1/\theta$. In the duration T_w, the total volume of downloaded data is $\sum_i \pi_i a_i T_w$ in bits if there is no flow control. Therefore, the volume of downloaded but unwatched data is

$$\min\left\{\sum_i \pi_i a_i T_w, \sum_i \pi_i r_i \Big/ \theta\right\} - \sum_i \pi_i r_i T_w. \tag{21}$$

When flow control is enforced, the average wasted prefetching is no larger than $\phi_a \sum_i \pi_i r_i$, that is, the product of flow control threshold and average bit-rate.

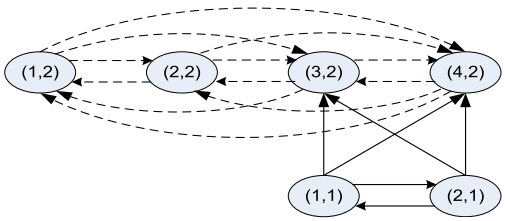


Fig. 1. Markov process of bit-rate switching with $q > \phi_b$.

Remark 5. For the BO switching, the amount of wasted prefetching increases w.r.t. the user's watching time before the completion of downloading the whole video, and decreases afterwards. The maximum waste happens at the time that the user terminates video playback when whole video is just completely downloaded.

4 Buffer-Aware Bit-Rate Switching

In this section, we investigate the (dis)advantages of requesting bit-rates higher than the channel rates when the buffer filling level is large.

4.1 Buffer-Aware Bit-Rate Switching

The default BO switching algorithm is "conservative" in that the requested bit-rate is usually below the channel rate. An interesting question is what are the benefits and hazards of being more aggressive to use bit-rates higher than the channel rate when the buffer length is larger. We hereby introduce a switching algorithm similar to the one used by Akamai HD Networks. Denote by ϕ_b the SWITCH_DOWN threshold in seconds of video content. We define two phases regarding the buffer length q, *Phase-A* and *Phase-C*. Phase-A refers to the stage $q < \phi_b$, and Phase-C refers to the stage $q \geq \phi_b$. In Phase-A, the BA switching behaves the same as the BO switching. In Phase-C, the requested video bit-rate may increase according to the same rules in Phase-A if the channel rate ramps up. If the channel rate decreases to a level below the current playback rate, the SWITCH_DOWN action is not triggered immediately. It occurs when the queue length is below ϕ_b. To highlight, one channel rate is mapped into one bit-rate uniquely in BO switching, and is mapped into multiple bit-rates in BA switching when the queue length is large (i.e., $\geq \phi_b$).

In Fig. 1, we illustrate all the possible switchings when the buffer length is larger than ϕ_b. This example looks at a system with four channel conditions $\{150, 300, 500, 700\}$ Kbps. Two different bit-rates $\{240, 480\}$ Kbps labeled as $\{1, 2\}$ are considered. Then, the strategy at Phase-A is $\mathcal{S} := (1, 1, 2, 2)$ with $c_1 < 0$ and $c_i > 0$ for $i > 1$. We also call the bit-rates in $\mathcal{S}$ the *default* video qualities.

The strategy at Phase-C becomes much more complicated. We define a state as a tuple (i, l) where i denotes the channel state and l denotes the bit-rate level. Note that all the state transitions in Fig. 1 are Markovian. The requested bit-rate may increase as the channel rate becomes larger. Hence, the transition rate from (i, l) to (j, l_j) is α_{ij} for any l, given $i < j$. If $i > j$, the requested bit-rate does not change, causing the transition from (i, l) to (j, l) with the rate α_{ij}. In

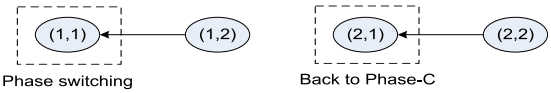


Fig. 2. Rule based switching as buffer length decreases to ϕ_b.

Fig. 2, we illustrate the rule based state transitions when the buffer length reduces to the SWITCH_DOWN threshold ϕ_b. The user at state $(1, l)$ (here $l = 2$) switches to state $(1, 1)$ and enters Phase-A. While at state (i, l) with $c_i > 0$, it switches to a new state (i, l_i). In this new state, the arrival rate is greater than the playback rate so that the buffer length continues to stay at Phase-C. For instance, the state $(2, 2)$ changes to the state $(2, 1)$ when the buffer length reduces to ϕ_b. It does not enter Phase-A, but returning to Phase-C again. Our purpose here is to study the impact of aggressive bit-rate switching on the QoE metrics. The number of Markov states at Phase-C is in the order of $L \times I$. The dynamics of state transitions at Phase-C is more complicated than that at Phase-A.

4.2 Computing Starvation Behaviors

The starvation analysis of BA switching is much more challenging because the set of system transitions in Figs. 1 and 2 are not Markovian when putting them together. We first need to examine how the buffer length crosses between Phase-A and C.

Starvation probability. In Phase-A, our purpose is to compute the probability of playback interruption. The switching scheme is the same as the BO scheme. Then, the ODEs of starvation probability are also the same as Eq. (4). But the boundary condition $\lim_{q\to\infty} V_i(q) = 0$ does not hold any more because the queue length cannot exceed ϕ_b at Phase-A. This leads to a different method to compute the starvation probabilities with no initial prefetching $V_i(0), (\forall 1 \leq i \leq I)$.

In Phase-C, as long as the mean throughput cannot support the highest bit-rate level of $\mathcal{S}$, the playout buffer length will decrease to ϕ_b very likely. We need to compute the transition probability from Phase-C to Phase-A. To obtain this result, we first need to derive the probability that the threshold ϕ_b is hit. Let (i^*, l^*) be the targeted state that the buffer length q decreases to ϕ_b. Denote by $W_{il}^{i^*l^*}(q)$ the probability that the buffer length starts to change at state (i, l) from q and hits ϕ_b at state (i^*, l^*) first. Conditioned on all possible events (i.e., change of state, departure of user, and no change of state) in the slot $[0, h]$, we obtain the dynamics of $W_{il}^{i^*l^*}(q)$ for $q \geq \phi_b$ by

$$\begin{aligned} W_{il}^{i^*l^*}(q) = (1-\alpha_i h)e^{-\delta_i^l h} W_{il}^{i^*l^*}\left(q + c_i^l h\right) \\ + \sum_{j>i} \alpha_{ij} h \cdot e^{-\delta_i^l h} W_{j,l_j}^{i^*l^*}\left(q + c_i^l h\right) \\ + \sum_{j<i} \alpha_{ij} h e^{-\delta_i^l h} W_{jl}^{i^*l^*}\left(q + c_i^l h\right). \end{aligned} \tag{22}$$

The first item in the right side of Eq. (22) denotes the case of no state change. The second one represents the transition from state (i, l) to state (j, l_j) with $a_j > a_i$. The third one reflects the change of channel condition without degrading

the requested bit-rate. Using the same approach, we obtain a set of ODEs for all (i,l)

$$c_i^l \dot{W}_{il}^{i^*l^*}(q) = \big(\alpha_i + \delta_i^l\big) W_{il}^{i^*l^*}(q) - \sum_{j>i} \alpha_{ij} W_{j,l_j}^{i^*l^*}(q) - \sum_{j<i} \alpha_{ij} W_{jl}^{i^*l^*}(q). \tag{23}$$

with the initial and boundary conditions

$$\begin{cases} W_{il}^{i^*l^*}(q) = 0 & \text{if } r_{l^*} < a_{i^*}; \\ W_{il}^{i^*l^*}(q) = 0 & \text{if } l > l^*; \\ W_{i^*l^*}^{i^*l^*}(\phi_b) = 1 & \text{if } r_{l^*} > a_{i^*}; \\ W_{il}^{i^*l^*}(\phi_b) = 0 & \text{if } (i,l) \neq (i^*,l^*) \text{ and } r_l > a_i; \\ W_{il}^{i^*l^*}(q)|_{q\to\infty} = 0 & \forall i, i^* \in \mathcal{I}; l, l^* \in \mathcal{R}. \end{cases} \tag{24}$$

The first condition of Eq. (24) says that ϕ_b cannot be hit at a state with $r_{l^*} < a_{i^*}$. The second one shows that a user at state (i,l) cannot hit ϕ_b at state (i^*,l^*) with $l > l^*$. The requested bit-rate can decrease only when q reduces to ϕ_b. The third one manifests the scenario that the buffer length decreases to ϕ_b for sure at state (i^*,l^*). The fourth one states that ϕ_b is reached at a state other than (i^*,l^*). The last one gives an asymptotic probability of hitting ϕ_b. As the buffer length is infinitely large, the user has already departed (or finished downloading) without reaching ϕ_b. The probabilities of hitting ϕ_b at different states can be solved in the similar way as the starvation probabilities of BO switching. The first step is to take LT over Eq. (23). Next, we compute the left-boundary conditions $W_{il}^{i^*l^*}(q)$ using Theorem 1. The only difference lies in that we compute $W_{il}^{i^*l^*}$ starting from the highest l^* iteratively.

However, hitting ϕ_b does not necessarily mean the switching from Phase-C to Phase-A. If the hitting event is at a state i with $a_i < r_{l_i}$ (e.g., $a_1 < r_1$ in Fig. 1), the switching from Phase-C to A takes place immediately. If ϕ_b is reached at the state (i,l) that has $a_i > r_{l_i}$, the new state becomes (i,l_i) and the buffer length increases above ϕ_b again. The phase switching does not happen instantly, but might be possible in the future. We denote by $P_{C\to A}^{i,j}(\phi_b)$ the probability that the playout buffer enters Phase-C at the state (i,l_i) in the beginning, and returns to Phase-A at the state (j,l_j). It is obvious to see $i \in \overline{\mathcal{I}}$ and $j \in \underline{\mathcal{I}}$. Otherwise, the playout buffer cannot enter Phase-C at the channel state i and leave Phase-C at the channel state j. Therefore, if the entry state to Phase-C is (i,l_i) (with $r_{l_i} < a_i$), there are three possibilities. One is that the user has left the system at Phase-C. The second is that the playout buffer leaves Phase-C at the state $(j,l_j)|_{r_{l_j}>a_j}$ with probability $\sum_{k=1}^{L} W_{i,l_i}^{j,k}(\phi_b)$. The third is that the buffer length increases first. After some time, it decreases until ϕ_b at the state (m,k) that has $r_k > a_m$ and $a_m > r_{l_m}$. Since there has $a_m > r_{l_m}$, according to the switching rules, the buffer length increases again at the new state (m,l_m) (i.e., the request bit-rate level decreases from k to l_m). Then, the phase switching probability in the future is $P_{C\to A}^{m,j}(\phi_b)$. Summing up all the possibilities, we obtain the switching probability from Phase-C to Phase-A through a set of linear functions

$$P_{C\to A}^{i,j}(\phi_b) = \sum_{k=1}^{L} W_{il_i}^{jk}(\phi_b) + \sum_{m=1}^{I}\sum_{k=1}^{L} W_{il_i}^{mk}(\phi_b) \cdot \mathbf{1}_{r_{l_m}<a_m} \cdot P_{C\to A}^{m,j}(\phi_b), \quad \forall i \in \overline{\mathcal{I}}, j \in \underline{\mathcal{I}}. \tag{25}$$

It is easy to see that $P_{C\to A}^{i,j}(\phi_b)$ can be solved by inverting a matrix of coefficients.

Next, we combine the analysis in Phase-A and Phase-C together. Recall that $P_{C\to A}^{i,j}(\phi_b)$ is the probability in which the playout buffer leaves Phase-A at the ith channel state, and re-enters Phase-A at the jth channel state. Therefore, the starvation probability $V_i(\phi_b)$ is the sum of starvation probabilities when the playout buffer re-enters Phase-A,

$$V_i(\phi_b) = \sum_{j\in\underline{\mathcal{I}}} P_{C\to A}^{i,j}(\phi_b) V_j(\phi_b), \quad \forall i \in \overline{\mathcal{I}}. \tag{26}$$

We proceed to compute the probability of starvation in Phase-A. The dynamics of $V_i(q)$ remains the same as that of the buffer-oblivious switching scheme in Eq. (8). However, the initial state $V_i(0)$ cannot be solved by letting q be infinity. Similar to the scenario with receiver-side flow control, the right boundary condition in Theorem 1 does not hold any longer. Then, we utilize Eqs. (16) and (26) to solve the unknowns $V_i(0), (\forall i \in \overline{\mathcal{I}})$, and $V_j(\phi_b), (\forall j \in \underline{\mathcal{I}})$. Note that the ill-conditioned linear functions are solved through the same technique as that used in the BOFC switching.

We next analyze a special case that the user's watching time is long enough (i.e., $\theta \to 0$). If the average channel rate is less than $r_I^{l_I}$, the highest rate in $\mathcal{S}$, the queue length in Phase-C will return to ϕ_b. The asymptotic starvation probability is given in the following proposition.

Proposition 5. *The asymptotic starvation probability (i.e., $\theta \to 0$) of BA switching is 1 if the following conditions hold: i) at least one state with $c_i^{l_i} < 0$ in Phase-A, and ii) the average channel rate less than the maximum bit-rate of $\mathcal{S}$, i.e., r_{l_I}.*

Time till starvation. It has been shown that the probability of starvation is 1 when $\theta = 0$ and $\sum_{i\in\mathcal{I}} a_i\pi_i < \max\{r_l, l \in \mathcal{S}\}$. Under this circumstance, it is interesting to show how the threshold ϕ_b influences the continuous playback duration. To make our method easily understandable, we consider a simpler case. The bit-rates and the available bandwidth satisfy $a_1 < r_{l_1}$ and $r_{l_i} < a_i$ for $i > 1$. That is, $\underline{\mathcal{I}} = \{1\}$ and $\overline{\mathcal{I}} = \{2,\ldots,I\}$. Then, the switching from Phase-C to A happens only at $i = 1$.

In Phase-C, we denote by $U_{il}(q)$ the expected continuous playback time before the buffer length reduces to ϕ_b. Here, the initial state is (i,l) and the initial buffer length is q. Following the same approach, we take a look at the dynamics of $U_{il}(q)$ for $q \geq \phi_b$ at an infinitesimal slot $[0,h]$.

$$U_{il}(q) = (1-\alpha_i h)\big(h + U_{il}\big(q + c_i^l h\big)\big) + \sum_{j>i} \alpha_{ij} h\big(h + U_{j,l_j}\big(q + c_i^l h\big)\big) + \sum_{j<i} \alpha_{ij} h\big(h + U_{jl}\big(q + c_i^l h\big)\big).$$

By letting $h \to 0$, we obtain for all $i \in \mathcal{I}$

$$c_i^l \dot{U}_{il}(q) = \alpha_i U_{il}(q) - \sum_{j>i} \alpha_{ij} U_{jl_j}(q) - \sum_{j<i} \alpha_{ij} U_{jl}(q) - 1, \quad (27)$$

where $\dot{U}_{il}(q)$ is the derivative of $U_{il}(q)$ over q. The initial and boundary conditions satisfy

$$\begin{cases} U_{il}(\phi_b) = 0 & \text{if } r_l > a_i; \\ \frac{U_{il}(q)}{q}|_{q\to\infty} < \infty & \forall i \in \mathcal{I}; l \in \mathcal{S}. \end{cases} \quad (28)$$

The left boundary condition means that the continuous playback time is 0 before q hits ϕ_b if the currant channel rate a_i is not able to support the bit-rate r_l. The second item is the right side boundary condition. The continuous playback time is actually the busy period of G/G/1 queue. It does not grow exponentially with q. Let $\hat{U}_{il}(s)$ be the LT of $\dot{U}_{il}(q)$. We take LT over Eq. (27) and obtain

$$\left(s - \frac{\alpha_i}{c_i^l}\right)\hat{U}_{il} + \sum_{j>i} \frac{\alpha_{ij}}{c_i^l} \hat{U}_{jl_j} + \sum_{j<i} \frac{\alpha_{ij}}{c_i^l} \hat{U}_{jl} = U_{il}(0) - \frac{1}{c_i^l s}. \quad (29)$$

The solution approach follows the one used in buffer-oblivious switching. We can solve $U_{il}(q)$ for the highest bit-rate $l \in \mathcal{S}$ first and for other bit-rates step by step.

Our next step is to compute the expected time that the queue length resides in Phase-C. Denote by $X_i (i > 1)$ the expected time of the queue length in Phase-C when it enters Phase-C with the initial channel state i. Here, we recall $\underline{\mathcal{I}} = \{1\}$ and $\overline{\mathcal{I}} = \{2, \ldots, I\}$. If the buffer length reduces to ϕ_b at channel state $j = 1$, it switches to Phase-A immediately (i.e., $X_1 = 0$). If the buffer length decreases to ϕ_b at any state $j > 1$, it continues to stay at Phase-C, and the new requested bit-rate is changed to be l_j. Therefore, we obtain a set of linear equations as follows:

$$X_i = U_{il_i}(\phi_b) + \sum_{j \in \overline{\mathcal{I}}} \sum_{k=1}^{L} W_{il_i}^{jk}(\phi_b) X_j, \quad i > 1. \quad (30)$$

At last, we calculate the time till starvation in Phase-A. The solution to $U_i(q)$ can be solved by the LTs in Eq. (12). Whereas the right boundary conditions are replaced by

$$U_i(\phi_b) = U_1(\phi_b) + X_i, \quad \forall i > 1. \quad (31)$$

Eq. (16) contains I linear equations with the $2I-1$ unknowns $U_2(0), \ldots, U_I(0)$, $U_1(\phi_b), \ldots, U_I(\phi_b)$ when ϕ_a is replaced by ϕ_b. Submitting $I-1$ linear Equations of (31) to Eq. (16), We obtain I linear equations with the remaining unknowns $U_1(\phi_b)$ and $U_2(0), \ldots, U_I(0)$. After solving these unknowns, we obtain $U_i(q)$ for any $q \le \phi_b$, $i \in \mathcal{I}$.

4.3 Computing Average Bit-Rate

Finding the average bit-rate of BA switching is very intuitive. It takes full advantage of the available bandwidth when the average channel rate is less than the highest bit-rate in $\mathcal{S}$. Meanwhile, it does not result in the continuous increase of buffer occupancy. Thus, the average bit-rate $E[R_{ba}]$ under this scenario is given by

$$E[R_{ba}] = \min\Big\{ \sum\nolimits_i \pi_i a_i,\ r_{l_I} \Big\}. \quad (32)$$

5 Illustration of a Two-State Channel

In this section, we provide the explicit QoE metrics in a wireless channel with two states {Good, Bad}. We denote by 1 the bad channel state, and by 2 the good one. Suppose that the video content is encoded into L versions. Let l_1 be the default bit-rate level at the bad state, and l_2 be that at the good state. The bit-rates are r_{l_1} and r_{l_2} respectively. We suppose that there have $a_1 < r_{l_1} < r_{l_2} < a_2$. This implies that the starvation events may happen at the 1^{st} channel state. When evaluating the mean bit-rate, we suppose that the mean arrival rate is above the mean video bit-rate.

5.1 Buffer-Oblivious Switching: BO and BOFC

Due to tight page limit, we only provide the flowcharts and the results. More details can be found in the supplementary file, available online. For BO switching, we need four steps:

- Step 1. Calculating the matrix $\mathbf{M}_V$, the determinant $sI + \mathbf{M}_V$ and the roots s_1 and s_2 ($s_1 > 0$ and $s_2 < 0$);
- Step 2. Calculating inverse LTs $V_i(q)$ in Eq. (8) and $U_i(q)$ in Eq. (12);
- Step 3. Calculating $V_2(0)$ using Theorem 1;
- Step 4. Calculating $U_2(0)$ based on Eq. (13).

The starvation probabilities are obtained as follows:

$$V_1(q) = \exp(s_2 q) \quad \text{and} \quad V_2(q) = V_2(0) \cdot \exp(s_2 q), \quad (33)$$

where ($c_1 < 0$ and $c_2 > 0$)

$$s_{1,2} = \frac{\alpha_{12}+\delta_1}{2c_1} + \frac{\alpha_{21}+\delta_2}{2c_2} \pm \sqrt{\left(\frac{\alpha_{12}+\delta_1}{2c_1} - \frac{\alpha_{21}+\delta_2}{2c_2}\right)^2 + \frac{\alpha_{12}\alpha_{21}}{c_1 c_2}}$$

and

$$V_2(0) = \left(-\frac{\alpha_{21}+\delta_2}{2c_2} + \frac{\alpha_{12}+\delta_1}{2c_1} + \sqrt{\left(\frac{\alpha_{12}+\delta_1}{2c_1} - \frac{\alpha_{21}+\delta_2}{2c_2}\right)^2 + \frac{\alpha_{12}\alpha_{21}}{c_1 c_2}} \right) \frac{c_1}{\alpha_{12}}.$$

The mean continuous playback times are given by

$$U_1(q) = -\frac{(\alpha_{12} + \alpha_{21})q}{\alpha_{12} c_2 + \alpha_{21} c_1}, \quad (34)$$

$$U_2(q) = \frac{c_1 - c_2}{\alpha_{12} c_2 + \alpha_{21} c_1} - \frac{(\alpha_{12} + \alpha_{21})q}{\alpha_{12} c_2 + \alpha_{21} c_1} \quad (35)$$

(confined to $\alpha_{12} c_2 + \alpha_{21} c_1 < 0$). Since the stationary distribution of channel rates is $\pi_1 = \frac{\alpha_{21}}{\alpha_{12}+\alpha_{21}}$ and $\pi_2 = \frac{\alpha_{12}}{\alpha_{12}+\alpha_{21}}$, the mean video bit-rates of BO switching is obtained by $E[R_{bo}] = \frac{\alpha_{21} a_1 + \alpha_{12} a_2}{\alpha_{21} b_1 + \alpha_{12} b_2}$.

Step 1 and 2 are the same for the BO and BOFC switchings. But the right boundary to compute the unknown $V_2(0)$ is different. Replacing q by ϕ_a in $V_i(q)$ obtained via Step 2 for $i = 1, 2$, we derive two linear equations regarding three unknowns, $V_1(\phi_a)$, $V_2(0)$ and $V_2(\phi_a)$. We next cast about for the connection between $V_1(\phi_a)$ and $V_2(\phi_a)$. In this two-state wireless channel, the probability that the buffer returns from Phase-B to A is given by $\psi_{21} := \frac{\alpha_{21}}{\alpha_{21}+\theta}$. Then, there has $V_2(\phi_a) = \psi_{21} V_1(\phi_a)$. We can compute three unknowns using three linear equations, and obtain $V_i(q)$ for $i = 1, 2$ subsequently.

The BOFC continuous playback time also contains three unknowns, $U_1(\phi_a)$, $U_2(0)$ and $U_2(\phi_a)$. Step 2 calculates $U_i(q)$ with the unknown $U_2(0)$. Replacing q by ϕ_a in $U_i(q)$ for $i = 1, 2$, we obtain two linear equations. From Eq. (18), the duration that queue length stays in Phase-B satisfies $U_2(\phi_a) - U_1(\phi_a) = 1/\alpha_{21}$. We can then solve all the unknowns and obtain $U_i(q)$, $i = 1, 2$ and $q \leq \phi_b$.

Given the channel rate distribution, the BOFC mean video bit-rate is given by $E[R_{bofc}] = \frac{\alpha_{21} r_1 + \alpha_{12} r_2}{\alpha_{12} + \alpha_{21}}$.

5.2 Buffer-Aware Switching: BA

The buffer dynamics at Phase-A is the same as that of BO switching. Following Step 1 and 2 of BO switching, we obtain the expressions of $V_i(q)$ (resp. $U_i(q)$) for $i = 1, 2$ with the unknown $V_2(0)$ (resp. $U_2(0)$). Substituting q by ϕ_b, we obtain two linear equations with three unknowns, $V_1(\phi_b)$, $V_2(0)$ and $V_2(\phi_b)$ (resp. $U_1(\phi_b)$, $U_2(0)$ and $U_2(\phi_b)$).

We next inspect the buffer dynamics at Phase-C. The system state must be $(2, 2)$ when the queue length enters Phase-C from Phase-A. It must be at state $(1, 2)$ when the queue length returns to Phase-A. Then, the buffer dynamics of BA switching at Phase-C is also similar to that of BO switching. One difference is that the queue length decreases to ϕ_b in BA switching, but decreases to 0 in BO switching. The other is that the requested bit-rate is r_{l_2}. Hence, the buffer dynamics of BA switching is greatly simplified in the two-sate wireless channel.

At Phase-C, we substitute $c_1, c_2, \delta_1, \delta_2$ by their counterparts $c_1^2, c_2^2, \delta_1^2, \delta_2^2$. With abuse of notations, we let $\hat{s}_1$ and $\hat{s}_2$ be the roots of the determinant $sI + \mathbf{M}_V$ that have $\hat{s}_1 > 0$ and $\hat{s}_2 < 0$. We use $\hat{V}_i(q)$ (resp. $\hat{U}_i(q)$) to denote the starvation probability (resp. the mean starvation duration) of BO switching when c_1, c_2, δ_1 and δ_2 are replaced by c_1^2, c_2^2, δ_1^2 and δ_2^2. The probability that the queue length switches to Phase-C from Phase-A, and switches back to Phase-A is computed by $P_{C\to A}^{2,1} = W_{2,2}^{1,2}(\phi_b) = \hat{V}_2(0)$. The above equation establishes the following relationship: $V_2(\phi_b) = P_{C\to A}^{2,1} V_1(\phi_b)$. Thus, the unknowns $V_1(\phi_b)$, $V_2(0)$ and $V_2(\phi_b)$ are solved, and $V_i(q)$ is derived for any $q \leq \phi_b$. Similarly, the expected playback time at Phase-C satisfies

$$X_2(\phi_b) = \hat{U}_2(0) = \frac{c_1^2 - c_2^2}{\alpha_{12} c_2^2 + \alpha_{21} c_1^2}$$

when the initial conditions are $q = \phi_a$ and $i = 2$. This means that $U_i(q), i = 1, 2$, satisfy $U_2(\phi_b) = U_1(\phi_b) + X_2(\phi_b)$. We can then solve the unknowns and obtain $U_i(q)$ for any $q \leq \phi_b$, $i = 1, 2$.

Given the stationary distribution of channel rates, the average video bit-rate is obtained by

$$E[R_{ba}] = \min\left\{\frac{\alpha_{12} a_2 + \alpha_{21} a_1}{\alpha_{12} + \alpha_{21}}, r_{l_2}\right\}.$$

It is easy to validate $E[R_{ba}] > E[R_{bo}]$ and $E[R_{ba}] > E[R_{bofc}]$ under the condition $\alpha_{12} c_2 + \alpha_{21} c_1 < 0$.

Remark 6. We summarize our direct observations as follows:

- $V_i(q)$ decreases *exponentially* w.r.t. to the initially prefetched content q.
- $U_i(q)$ of BO switching is a linear function of q .
- $U_i(q)$ of BOFC and BA switchings do not increase linearly w.r.t. q.
- The average video bit-rate of BA switching is the highest.

6 Evaluation

6.1 Simulation Setup

The mathematical models of QoE metrics are computed by MATLAB. We further use MATLAB to simulate the event-driven bit-rate switching system where all the events are triggered by a timer. These events keep track of the variations of channel rate, the requests of video chunks, the completions of downloading chunks, and the buffer starvations. The timer generates random time stamps that record the transitions of channel rate from one state to the other. We then monitor the playout buffer length based on the packet arrival rate and the playback rate in packets. A starvation event happens when all the packets have been served. We take into account the practical restriction of requesting video chunks. The bit-rate switching is not performed at the packet level, but at the chunk level. Once a chunk is requested, its bit-rate remains the same even if the channel rate changes during the downloading. The request of a new chunk takes place only after the completion of the current chunk. If there is no flow control, the client requests video chunks immediately one after the other. When flow control is introduced, the client schedules the time of requesting new chunks. The chunk size is recommended to be 2 s in [2] and by Akamai. We run each set of experiments for 3,000 times.

The wireless channel is modeled by a continuous time Markov process with the set of rates $\mathcal{I} = \{150, 300, 500, 700\}$ Kbps. The channel state transition matrices (two examples) are given by

$$\alpha_{\mathbf{A}} = \begin{bmatrix} 0 & 0.05 & 0 & 0 \\ 0.03 & 0 & 0.03 & 0 \\ 0 & 0.03 & 0 & 0.02 \\ 0 & 0 & 0.06 & 0 \end{bmatrix},$$

$$\alpha_{\mathbf{B}} = \begin{bmatrix} 0 & 0.03 & 0.01 & 0 \\ 0.06 & 0 & 0.02 & 0.01 \\ 0.01 & 0.05 & 0 & 0.02 \\ 0.01 & 0.02 & 0.06 & 0 \end{bmatrix}.$$

The set of bit-rates is $\mathcal{R} = \{240, 360, 480, 600\}$ Kbps. When the channel rate is 500 Kbps, the client can request either 480 or 360 Kbps chunks. Under the rule that the requested bit-rate (except the lowest one) is less than the channel rate, we choose candidate strategies $\mathcal{S}_1 = (1, 1, 3, 4)$ and $\mathcal{S}_2 = (1, 1, 2, 4)$. For notional convenience, we let the more aggressive strategy $\mathcal{S}_1$ be the default strategy $\mathcal{S}$. If not mentioned explicitly, the simulations are for $\mathcal{S}$. In $\mathcal{S}$, the mean arrival rate is larger than the mean bit-rate with $\alpha_{\mathbf{A}}$, and less than the mean bit-rate with $\alpha_{\mathbf{B}}$. We adopt the transition matrix $\alpha_{\mathbf{A}}$ when analyzing starvation probability, and $\alpha_{\mathbf{B}}$ when analyzing continuous playback time. This is because the expected continuous playback time is infinite for any initial buffer length when mean arrival rate is greater than mean playback rate in BO switching. The default mean watch time of a user is set to $1/\theta = 1{,}000$ seconds.

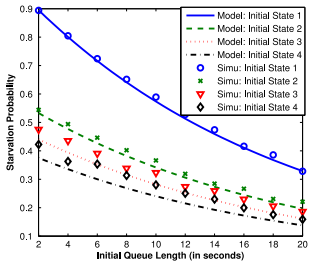


Fig. 3. Starvation probability (BO) versus initial queue length.

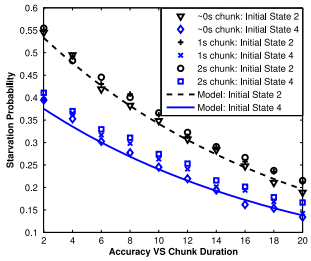


Fig. 4. Accuracy of model (BO) versus chunk durations (~0, 1, and 2s).

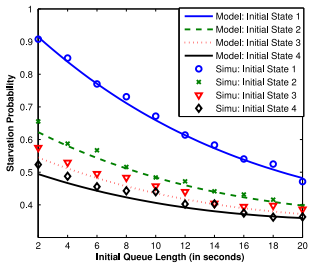


Fig. 5. Starvation probability (BOFC with $\phi_a = 30$) versus initial queue length.

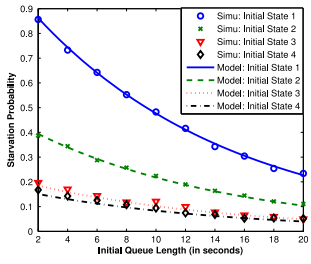


Fig. 6. Starvation probability (BO) of $\mathcal{S}_2$ versus initial queue length.

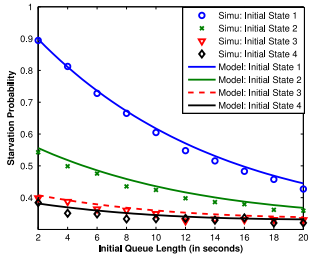


Fig. 7. Starvation probability (BOFC) of $\mathcal{S}_2$ versus initial queue length.

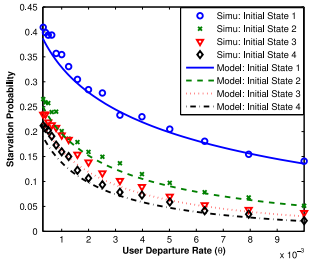


Fig. 8. Starvation probability (BO) versus view time.

6.2 Buffer-Oblivious Switching

Set 1: Starvation probability versus initial queue length. Here, we reveal the relationship between the starvation probability and the initial buffer length for BO switching in Fig. 3. As q increases from 2 to 20, the starvation probability decreases exponentially. When q is small, a slight increase of q leads to a fast reduction of the starvation probability. When q is large, further increasing q gives smaller reduction of starvation probability.

We next examine the accuracy of our model. As shown in Fig. 3, the proposed model underestimates the starvation probability in the simulation by $3 \sim 5$ percent. The main reason lies in that the bit-rate switching is realized at packet granularity in the model, but at chunk granularity in the simulation. When the channel rate varies, the on-going chunk is usually unfinished. The on-the-fly packets will still be transferred. If the channel rate decreases from a high to a low level, the unfinished chunk is of high video quality. Then, the size of the fractional chunk is large in bits, but is transmitted by a low channel rate. This increases the risk of meeting starvations. To validate our explanation, we reproduce the above experiments with different chunk durations. Fig. 4 illustrates the starvation probabilities when the chunk duration is chosen to be one packet (nearly 0 s), 1 and 2 s. The prediction error with 2 s chunk is larger than that with 1 s chunk. The simulation and the model match very well with infinitesimal chunk duration.

In Fig. 5, flow control is introduced. We set ϕ_a to be 30 s. When q increases from 2 to 20 s, the starvation probability also decreases. The cross comparison between Figs. 3 and 5 manifests that flow control leads to a much higher starvation probability. At the point $q = 20\,\mathrm{s}$ and $i = 4$, the starvation probability of BO is around 15.9 percent, but that of BOFC is around 36.3 percent in the simulation.

We next compare the risks of buffer starvation between $\mathcal{S}_1$ with $\mathcal{S}_2$. Under the same setting, $\mathcal{S}_2$ has much smaller starvation probabilities according to Figs. 6 and 7. For instance, when q is 20 s and the initial state is 4, the starvation probability of BO switching is only 5.1 percent. Hence, with our model, the user is able to determine whether to select $\mathcal{S}_1$ or $\mathcal{S}_2$, depending on the current queue length and the current channel state.

Set 2: Starvation probability versus user impatience. Figs. 8 and 9 show the starvation probabilities of BO and BOFC versus the view time respectively. The x-axis denotes θ, the inverse of the mean view time, which increases from 10^{-2} to 3×10^{-4}. This is equivalent to the increase of mean view time from 100 s to 3,333 s. The initial queue length is set to 30 s. In both experiments, the starvation probability

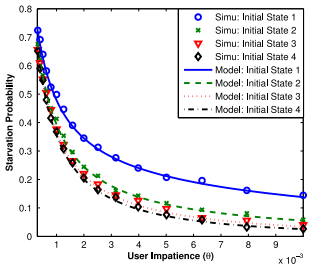


Fig. 9. Starvation probability (BOFC) versus view time.

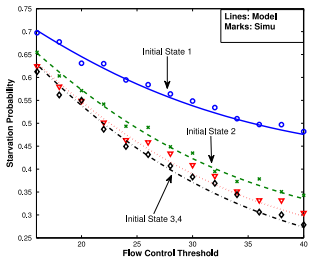


Fig. 10. Starvation probability (BOFC) versus flow control threshold.

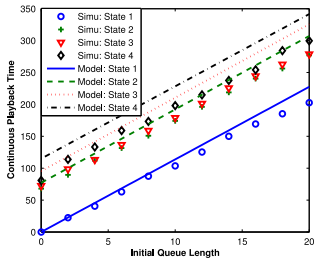


Fig. 11. Starvation duration (BO) versus initial queue length.

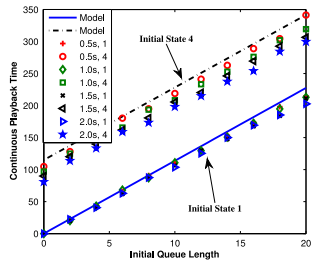


Fig. 12. Accuracy of model (BO) versus chunk duration.

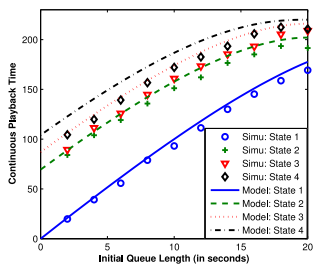


Fig. 13. Starvation duration (BOFC) versus initial queue length.

decreases with the increase of θ. It is very sensitive to the change of θ in certain ranges, which means that the viewers' behavior is an important factor to determine QoE. We next study the impact of flow control on the starvation probability via an example of $\phi_a = 30\,\text{s}$. When the mean view time is 1,000 s (i.e., $\theta = 10^{-3}$), the starvation probabilities of BO at all initial states are $[0.3545, 0.201, 0.193, 0.159]$, and those of BOFC are $[0.498, 0.413, 0.377, 0.363]$. Hence, one can see that flow control has a high risk of causing playback interruptions. We then compare BO and BOFC switching at two extreme points, $\theta = 10^{-2}$ and $\theta = 3 \times 10^{-4}$. Fig. 8 and 9 manifest that flow control has an almost negligible impact on the starvation probability for short view time, but greatly influences it for long view time.

Set 3: Starvation probability versus flow control threshold. Here, we investigate the role of the flow control threshold in determining the starvation probability in Fig. 10. The initial queue length is fixed to be 16 s. As ϕ_a increases from 16 to 40 s, the starvation probability decreases rapidly in the beginning, and continue to decrease with a smaller slope afterwards. Thus, selecting a very large threshold does not help much to reduce the starvation probability, but may cause excessive chunk requests in VoD service.

Set 4: Continuous playback time versus initial queue length. When the view time is large enough, the starvation probability is insufficient to reflect the severity of starvation. We plot the relationship between the continuous playback time and the initial buffer length of BO switching in Fig. 11. We observe that the continuous playback time increases linearly w.r.t. the initial buffer length. Fig. 11 also shows some gaps between model and simulation. These gaps are caused by the mismatch of the fluid model and the non-negligible chunk duration, the same reason for prediction error of starvation probability. Here, we explain why the prediction error of continuous playback time is more outstanding in this experiment. Due to the linearity relationship, the prediction error of the continuous playback time is proportional to the error of queue length. In this experiment, one second content can extend the continuous playback time by 11 s. A small prediction error of queue length magnifies the prediction error of continuous playback time. We next evaluate the impact of chunk duration on the accuracy of our model in Fig. 12. The prediction error becomes smaller if the chunk duration reduces. When it is no larger than 0.5 second, the simulations conform to our model accurately.

Fig. 13 plots the continuous playback time of BOFC switching with the default $\phi_a = 20\,\text{s}$. When the initial queue length increases from 0 to 20 s, the continuous playback time increases accordingly. An interesting observation is that the prediction error is becoming smaller and smaller. This is because the continuous playback time does not have a linear relationship with the initial queue length. As it is large, the continuous playback time becomes saturated, which reduces the impact of error of queue length on the continuous playback time. This set of experiments further validate that the continuous playback time of BO is larger than that of BOFC under the same initial condition.

6.3 Buffer-Aware Switching

We next evaluate the QoE metrics of buffer-aware bit-rate switching by varying the initial buffer length and the SWITCH_DOWN threshold. We let the set of bit-rates be $\mathcal{R} = \{240, 360, 480\}$ Kbps for simplicity.

Set 5: Starvation probabilities of BA switching. We let the default SWITCH_DOWN threshold be 30 s. The starvation

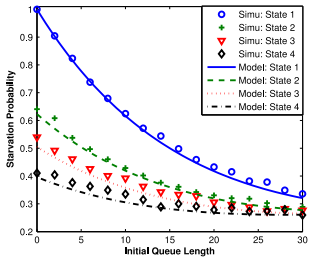


Fig. 14. Starvation probability (BA) versus initial buffer length.

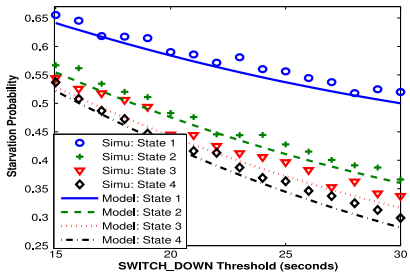


Fig. 15. Starvation probability (BA) versus SWITCH_DOWN threshold.

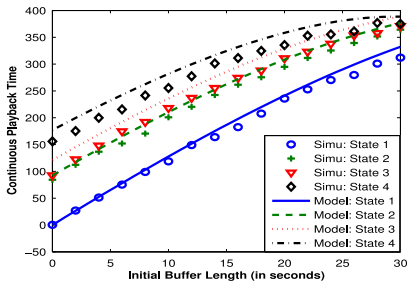


Fig. 16. Starvation duration (BA) versus initial buffer length.

probability in Fig. 14 exhibits the similar property as that in Figs. 3 and 5. When q increases from 0 to 30 s, the starvation probability decreases exponentially. The chunk duration is still 2 s while the prediction error of starvation probability is small in all the initial states.

We then evaluate the impact of SWITCH_DOWN threshold on the starvation probability. The initial buffer length is set to 15 seconds. In Fig. 15, the starvation probability decreases as ϕ_b increases from 15 to 30 s. The starvation probability is more sensitive to the increase of ϕ_b when ϕ_b is small. The absolute prediction error of the starvation probability is below 5 percent.

Set 6: Continuous playback time of BA switching. Similar to the BOFC switching, the starvation happens for sure when the view time is large enough and the mean arrival rate is less than 480 Kbps. Hence, we use the continuous playback time to test the severity of starvation. In Fig. 16, we show that the continuous playback time increases with the initial buffer length. We next evaluate the continuous playback time in Fig. 17 by increasing ϕ_b from 15 to 30 seconds. The initial buffer length is fixed to be 15 seconds. As ϕ_b increases, the continuous playback time increases accordingly. The percents in Fig. 17 denotes the maximum prediction error at each initial state. It is a ratio that the difference of values between model and simulation is divided by the value of model. We observe that the prediction

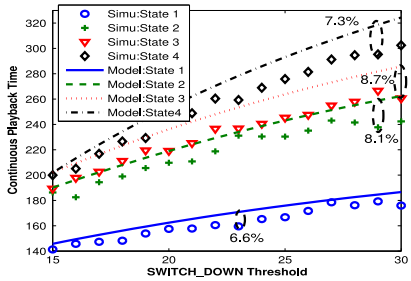


Fig. 17. Starvation duration (BA) versus SWITCH_DOWN threshold.

TABLE 1
Comparison of Average Video Bit-Rates

	BO	BOFC	BA
$\mathcal{S}_1$ (Simu)	361.015	359.660	379.120
$\mathcal{S}_1$ (Model)	367.636	362.727	382.955
$\mathcal{S}_2$ (Simu)	323.951	319.320	376.812
$\mathcal{S}_2$ (Model)	330.122	321.818	382.955

errors are small in general, and the maximum error is around 8 percent.

Set 7: Comparison of average video bit-rate. This set of experiments compares the average video bit-rates of various switching mechanisms. The channel transition matrix is set to α_A, and the set of video bit-rates is $\{240, 360, 480, 600\}$ Kbps. The chunk duration is 2 s and the initial queue length is 30 s. The flow control threshold and the SWITCH_DOWN threshold are both set to 30 s. The simulation stops when the buffer becomes empty or 1,000 chunks have been transmitted. By configuring a large initial queue length and large thresholds, the probability of starvation is small. Then, the average video bit-rate will be less influenced by the initial stage of video streaming. We measure the average video bit-rate of BO, BOFC and BA switching for different schemes, $\mathcal{S}_1$ and $\mathcal{S}_2$ (at Phase-A of BOFC and BA). Under this setting, the mean arrival rate is able to sustain the mean video bit-rate. Table 1 provides the average bit-rates from the models and the simulations. We have three observations. First, the models match the chunk based simulations very well. Second, for BO switching, choosing the more aggressive strategy $\mathcal{S}_1$ gives the user a better mean video quality. Third, BA switching has the similar mean video quality for both $\mathcal{S}_1$ and $\mathcal{S}_2$. The reason lies in that BA switching tries to fully utilize the channel bandwidth when the playout queue length is large. Even though $\mathcal{S}_2$ is less aggressive than $\mathcal{S}_1$ at Phase-A, BA switching makes the queue length of $\mathcal{S}_1$ and $\mathcal{S}_2$ stay at Phase-C at most of time. Thus, the average video bit-rates are very close in our experiment when either $\mathcal{S}_1$ or $\mathcal{S}_2$ are adopted at Phase-A.

7 Discussion and Limitations

In this section, we discuss how our model can be utilized to guide the design of adaptive streaming services and what the potential limitations may be.

What is a good switching algorithm? A good switching algorithm should avoid playback interruptions while maintaining high average bit-rate in our setting. However, it is very difficult to tell exactly what an optimal switching algorithm behaves. The difficulty lies in the quantization of user experience based on all the QoE metrics. The subjective

perception of a user also depends on his psychological factors, which is usually beyond mathematical modeling. Defining a QoE penalty function that reflects the tradeoff amongst various QoE metrics is an option in traditional video streaming services (e.g., [15], [16]). But this simplifies the realistic user satisfaction to a certain extent. Hence, we focus on how the objective metrics are obtained, instead of how they are mapped into a single subjective QoE score.

In addition to the qualitative observations of QoE metrics, our models can be utilized to improve QoE. Suppose that the user aims to maximize the average bit-rate and to control the probability of starvation below a certain value. Finding an optimal switching rule is difficult because the user has to take account of all the future events. A practical approach is to define several candidate strategies (e.g., $\mathcal{S}_1$, $\mathcal{S}_2$ and so on). In each time point of chunk request, the user can predict the starvation probability, knowing the current queue length and current channel state. By judging the QoE metrics of different candidate strategies, the user can pick up the right one to optimize his QoE objective function.

Wireless channel variation. The analytical framework is based on the famous finite-state Markov model of wireless fading channel [7], [8], [9], [22]. The FSM channel model has more advantages than the arbitrary channel variation because the QoE metrics can be studied qualitatively and can be predicted in the former. The continuous FSM channel model can be generalized to a discrete one in which time is slotted. The only change in our analytical framework is that the differential equations are replaced by the corresponding difference equations.

The transition matrix of channel rates can be measured offline and online. The offline method is suitable for the office or residential environments in which the channel rate is time varying, but the Markov transition rates are identical at the same time period everyday. Then, the channel model is known as a priori. The online method measures the chunk throughput. The user needs certain time to construct the transition matrix. Estimating FSM channel model is not complicated since the channel rates are in a finite set. Online measurements can also be done by wireless bottlenecks such as the base station or the access point. They can explicitly notify the channel model to the user. In theory, our model is applicative even if the FSM channel model has changed. However, when the transition rates change over time, their estimation becomes very difficult. The error in the channel model will cause inaccurate prediction of QoE metrics.

Impact of chunk size. The video chunks usually last a couple of seconds (e.g., 2 s) in adaptive streaming. Our model is a fluid model, which means that the chunk duration is assumed to be infinitesimal. With this assumption, the channel variation and the bit-rate switching are synchronized. Otherwise, it will be extremely difficult to characterize the dynamics of the playout buffer because all the events are coupled. Recent studies in [2], [23] also adopt the same assumption in their buffer models.

When the chunk duration is not infinitesimal, the streaming user can only change the bit-rate after the on-going chunk has been completely transmitted. This phenomenon influences the accuracy of our models. In the SWITCH_UP and SWITCH_DOWN actions, the the sizes of chunks are identical in seconds, but quite different in bits. When the channel rate changes from a high bit-rate to a lower one, the size (in bits) of the unfinished chunk is large, but the new channel rate is small. It takes much more time than expected to complete the delivery. Then, the queue length becomes shorter than that computed by our model. Therefore, our model underestimates the starvation probability and overestimates the continuous playback duration. The precision of our model improves with the decrease of chunk duration. In the simulations, the absolute error of starvation probabilities is less than 5 percent with 2 s chunk duration. When it is 0.5 s, our model matches the simulation exactly.

8 Related Work

The dynamic adaptive streaming over HTTP is a very promising technology for delivering video streaming of variable qualities in the large scale. A series of efforts have been made on the standardization of DASH system [4], [5]. However, the principle of selecting bit-rates is not specified. There have been rising interests in the experimental evaluation on existing DASH systems, and the improved bit-rate switching algorithms. De Cicco et al. in [6] did an experimental study on Akamai adaptive streaming over HTTP. They investigated the bit-rate switching behaviors in the presence of abrupt bandwidth change, and one concurrent greedy TCP flow. In [10], Akhshabi et al. evaluated and compared the design tradeoffs of three commercial DASH systems, Microsoft Smooth Streaming, Netflix Player and Adobe OSMF. Similar experiments were conducted by Riiser et al. [11] in a mobile 3G environment. A QoE-aware DASH system was proposed in [12] that consisted of a novel bandwidth proving method based on RTTs and an empirical bit-rate switching logic. The bit-rate instability caused by competing DASH flows was studied in [3].

The mathematical modeling of streaming QoE has drawn considerable attentions in the past several years. All the models are developed for single bit-rate streaming. Recently, [13], [14] presented bounds of start-up delay to avoid starvation. Luan et al. in [15] modeled the playout buffer as a G/G/1 queue. By using diffusion approximation, they obtained the closed-form starvation probability for a finite file size. Xu et al. [16] obtained the distribution of the number of starvation events using the Ballot theorem approach and an iterative approach. They further extended the same approach to analyze QoE metrics in wireless downlinks in [17]. In [18], they developed an analytical framework to study the QoE metrics in wireless downlink with flow arrivals and departures. This paper stands out from state of the art works in the ways: i) the first analytical study concerning the design tradeoff of bit-rate switching, ii) the consideration of bandwidth variation, and iii) the incorporation of user (im)patience and receiver-side flow control.

9 Conclusion and Future Work

In this paper, we propose the first analytical framework to predict the QoE of adaptive streaming in wireless networks. Two bit-rate switching algorithms are analyzed, where the first one is based only on channel variation, and the second

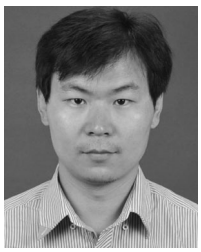
one is based on both channel variation and buffer length. We build the mathematical models for the starvation behavior and the mean video bit-rate. The proposed models can be parameterized to investigate the impact of initial content prefetching, user watch time, receiver-side flow control on QoE metrics. Our study has great practical value in guiding the design of bit-rate switching algorithm. It enables streaming user to predict the QoE of candidate switching strategies, thus selecting the appropriate one with certain QoE guarantee, given the current buffer length and channel condition. Our future study will explore the design of online bit-rate switching to achieve the optimal tradeoff amongst different QoE metrics.

Acknowledgments

This work is sponsored by Natural Science Foundation of China (No. 61402114), Shanghai Pujiang Program (No. 14PJ1401400) and Start-up Project for New Faculty of Fudan University.

References

[1] Allot communications. [Online]. Available: http://www.allot.com/index.aspx?id=3798&itemID=66228, July, 2011.
[2] G. Tian and Y. Liu, "Towards agile and smooth video adaption in dynamic HTTP streaming," in *Proc. ACM 8th Int. Conf. Emerging Netw. Experiments Technol.*, Nice, France, 2012, pp. 109–120.
[3] J. C. Jiang, V. Sekar, and H. Zhang, "Improving fairness, efficiency, and stability in HTTP-based adaptive video streaming with FESTIVE," in *Proc. ACM 8th Int. Conf. Emerging Netw. Experiments Technol.*, Nice, France, 2012, pp. 97–108.
[4] T. Stockhammer, "Dynamic adaptive streaming over HTTP: Design principles and standards," pp. 157–168., 2011.
[5] I. Sodagar and A. Vetro, "The MPEG-DASH standard for multimedia streaming over the internet," *IEEE MultiMedia Mag.*, vol. 18, no. 4, pp. 62–67, Apr. 2011.
[6] L. De Cicco, S. Mascolo, and V. Palmisano, "Feedback control for adaptive live video streaming," in *Proc. ACM 2nd Annu. ACM Conf. Multimedia Syst.*, 2011, pp. 145–156.
[7] 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. ACM 3rd Multimedia Syst. Conf.*, 2012, pp. 149–154.
[8] S. Y. Xiang and L. Cai, "Adaptive scalable video streaming in wireless networks," in *Proc. ACM 3rd Multimedia Syst. Conf.*, 2012, pp. 167–172.
[9] D. Jarnikov and T. Ozcelebi, "Client intelligence for adaptive streaming solutions," *Signal Process.: Image Commun.*, vol. 26, no. 7, pp. 378–389, 2011.
[10] S. Akhshabi, A. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP," in *Proc. ACM 2nd Annu. ACM Conf. Multimedia Syst.*, 2011, pp. 157–168.
[11] H. Riiser, H. S. Bergsaker, P. Vigmostad, P. Halvorsen, and C. Griwodz, "A comparison of quality scheduling in commercial adaptive HTTP streaming solutions on a 3G Network," in *Proc. ACM 4th Workshop Mobile Video*, 2012, pp. 25–30.
[12] K. Ricky, X. Luo, E. Chan, and R. Chang, "QDASH: A QoE-aware DASH system," in *Proc. ACM 3rd Multimedia Syst. Conf.*, 2012, pp. 11–22.
[13] G. Liang and B. Liang, "Effect of delay and buffering on jitter-free streaming over random VBR channels," *IEEE Trans. Multimedia*, vol. 10, no. 6, pp. 1128–1141, Oct. 2008.
[14] A. ParandehGheibi, M. Medard, A. Ozdaglar, and S. Shakkottai, "Avoiding interruptions a QoE reliability function for streaming media applications," *IEEE J. Sel. Area Commun.*, vol. 29, no. 5, pp. 1064–1074, May 2011.
[15] H. Luan, L. X. Cai, and X. Shen, "Impact of network dynamics on users' video quality: Analytical framework and QoS provision," *IEEE Trans. Multimedia*, vol. 12, no. 1, pp. 64–78, Jan. 2010.
[16] Y. D. Xu, E. Altman, R. Elzouzi, M. Haddad, S. Elayoubi, and T. Jimenez, "Probabilistic analysis of buffer starvation in markovian queues," in *Proc. IEEE INFOCOM*, Orlando, USA, 2012, pp. 1826–1834.
[17] Y. D. Xu, E. Altman, R. Elzouzi, S. E. Elayoubi, and M. Haddad, "QoE analysis of media streaming in wireless data networks," in *Proc. 11th Int. IFIP TC 6 Conf. Netw.*, Prague, Czech Republic, 2012, pp. 343–354.
[18] Y. D. Xu, S. Elayoubi, E. Altman, and R. Elzouzi, "Impact of flow-level dynamics on QoE of video streaming in wireless networks," in *Proc. IEEE INFOCOM*, Turin, Italy, 2013, pp. 2715–2723.
[19] N. Biuerle, "Some results about the expected ruin time in Markov-modulated risk models," *Insurance: Math. Economics*, vol. 18, pp. 119–127, 1996.
[20] G. H. Golub and V. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD, USA: Johns Hopkins Univ. Press, pp. 439–445, 1996.
[21] S. Asmussen, *Ruin Probabilities*, vol. 2, Singapore, World Scientific, 2000.
[22] P. Sadeghi, R. Kennedy, P. Rapajic, and R. Shams, "Finite-state Markov modeling of fading channels," *IEEE Signal Process. Mag.*, vol. 25, no. 5, pp. 57–80, Sep. 2008.
[23] T. Huang, R. Johari, and N. McKeown, "Downton abbey without the Hiccups: Buffer-based rate adaptation for HTTP video streaming," in *Proc. ACM SIGCOMM Workshop Future Human-Centric Multimedia Netw.*, 2013, pp. 9–14.



Yuedong Xu received the BS degree from Anhui University, the MS degree from Huazhong University of Science and Technology, and the PhD degree from The Chinese University of Hong Kong. He is a tenure-track associate professor in the School of Information Science and Technology, Fudan University, China. From late 2009 to 2012, he was a postdoc with INRIA Sophia Antipolis and Universite d'Avignon, France. His areas of interest include performance evaluation, control, optimization, and economic analysis of communication networks.

Yipeng Zhou received the BS degree from the University of Science and Technology of China, and the MPhil and PhD degrees both from The Chinese University of Hong Kong. He is an assistant professor in the College of Computer Science and Software Engineering, Shenzhen University. His research interest includes P2P/CDN content distribution, performance evaluation, network coding, and video streaming.

Dah Ming Chiu received the BS degree from Imperial College London and the PhD degree from Harvard University. After working in the industry (AT&T, DEC and Sun), he returned to academia in 2002. He is a full professor and chairman at the Department of Information Engineering, The Chinese University of Hong Kong. He was an associate editor of the *IEEE/ACM Transaction of Networking* from 2006 to 2012, and is a general chair of ACM SIGCOMM 2013. He became a fellow of the IEEE in 2008.