Impact of flow-level dynamics on QoE of video streaming in wireless networks

Yuedong Xu, Salah-Eddine Elayoubi, Eitan Altman, Rachid El-Azouzi

Impact of Flow-level Dynamics on QoE of Video Streaming in Wireless Networks

Yuedong Xu†, Salah Eddine Elayoubi*, Eitan Altman‡, Rachid El-Azouzi†
†University of Avignon, 339 Chemin des Meinajaries, Avignon, France
‡INRIA Sophia Antipolis, 2004 Route des Lucioles, France
*Orange Labs, 38 rue du General Leclerc, 92130 Issy-Les-Moulineaux, France
Email: {yuedong.xu@gmail.com, salaheddine.elayoubi@orange.com
eitan.altman@inria.fr, rachid.elazouzi@univ-avignon.fr}

***Abstract*—The Quality of Experience (QoE) of streaming service is often degraded by frequent playback interruptions. To mitigate the interruptions, the media player prefetches streaming contents before starting playback, at a cost of delay. We study the QoE of streaming from the perspective of flow dynamics. First, a framework is developed for QoE when streaming users join the network randomly and leave after downloading completion. We compute the distribution of prefetching delay using partial differential equations (PDEs), and the probability generating function of playout buffer starvations using ordinary differential equations (ODEs). Second, we extend our framework to characterize the throughput variation caused by opportunistic scheduling at the base station in the presence of fast fading. Our study reveals that the flow dynamics is the fundamental reason of playback starvation. The QoE of streaming service is dominated by the average throughput of opportunistic scheduling, while the variance of throughput has very limited impact on starvation behavior.**

I. Introduction

Streaming services are witnessing a rapid growth in mobile networks. According to Allot Communications [1], HTTP streaming service made up 37 percent of mobile broadband traffic during the second half of 2010. This presents new challenges for operators that are used to classify services into real-time (voice-like) and elastic (data-like) services. Indeed, classical QoS metrics in mobile networks are blocking rates for real-time traffic and average user throughput for elastic one, and operators dimension their networks for satisfying targets on those metrics [2]. However, the particular nature of streaming applications, halfway between real-time and elastic services, is raising the following difficult questions in wireless environments. First, which QoS metrics best represent the QoE perceived by users. Second, how to predict these QoE metrics for a given traffic intensity and to dimension the network.

The first step towards defining QoE and predicting it is to understand how streaming is played. In general, media players at the devices are equipped with a playout buffer that stores arriving packets. As long as there are packets in the buffer, the video is played smoothly. Once the buffer empties, the spacing between packets does not follow the original one. These *starvations* cause large *jitters* and are particularly annoying for end users that see frozen images. One feasible way to avoid starvations is to introduce a start-up (also called prefetching) delay before playing the stream, and a rebuffering delay after each starvation event. Then after a number of media frames accumulate in the buffer, the media player starts to work. This leads to two important sets of QoE metrics: starvation properties (probability, frequency, etc.) and startup/re-buffering delays.

Once the behavior of media streaming service is understood, the particularity of offering it over wireless networks is considered. Indeed, the wireless channel is subject to a large variability due to fading, mobility, etc. On top of this, it is a shared channel where multiple users are served simultaneously and cell capacity is divided among them. This introduces two variability time scales: flow level (tens of seconds) driven by the departures/arrivals of calls and wireless channel variability time scale (milliseconds) driven by the fast fading.

Related Literature

Video delivery over wireless networks has been studied for more than 15 years. Large attention was given to enhance coding in order to combat errors due to wireless channel variability. Recent work considered flow level performance in cellular networks delivering real time video. [2] studied QoS when integrating elastic and video traffic in cellular networks; video QoS was expressed by a blocking rate, while average throughputs and delays represent QoS for elastic traffic. [3] derived the Erlang-like capacity region for a traffic mix including real time video, the aim being to dimension the network for ensuring a target QoS. [4] derived the stability region of the network and showed how it is impacted by real-time video traffic. Recently, QoE for new streaming services was studied, taking into account the initial buffering period. [7] considered a G/G/1 queue where the arrival and service rates are characterized by their first two moments. [8] considered a channel that oscillates between *good* and *bad* states following the extended Gilbert model. [9] considered a P2P video streaming based on random linear network coding; this neglects chunk scheduling and peer selection inherent in P2P systems. An M/M/1 queue model has been adopted in [5], allowing to derive explicit formula for QoE metrics.

As of the tools used for deriving QoE metrics: [7] adopted a diffusion approximation where the discrete buffer size is replaced with a Brownian motion whose drift and diffusion coefficients are calculated. [8] presented a probabilistic analysis based on an a priori knowledge of the playback and arrival curves. [9] calculated bounds on the playback interruption

probability based on the adopted M/D/1 buffer model. Explicit formula of the distribution of the number of starvations has been obtained in [5], [6] based on the Ballot theorem.

The above mentioned works on QoE estimation are very useful for catching the impact of variability of the wireless channel due to fast fading or even user's mobility. However, the underlying models fail to capture the large variations due to flow dynamics. For instance, the diffusion approximation in [7] supposes that the drift and diffusion coefficients are constant over time, which is not true when the number of concurrent flows changes during playback process. The assumption of Poisson packet arrivals in [9], [5] also fails to take into account these flow dynamics. Note that the analysis of [9] has been generalized to a two-state Markovian arrival process, but this corresponds more to a bursty traffic due to a Gilbert channel model than to flow dynamics.

Our Work

To the best of our knowledge, this paper is the first attempt to assess the impact of flow dynamics on the QoE of streaming. We model the system as two queues in tandem. The first queue, representing the scheduler of the base station, is modeled as a processor sharing queue, while the second represents the playout buffer whose arrival rates are governed by the output process of the base station queue. We consider a static channel (no fast fading) with Constant Bit Rate (CBR) streaming, and derive the prefetching delay distribution and the starvation probability generation function using Partial Differential Equations (PDEs) as well as Ordinary Differential Equations (ODEs) constructed over the Markov process describing the flow dynamics. We next extend the model to include a fast fading channel and show that the impact of flow dynamics is preponderant over the channel variability due to fast fading. Extensive simulations show that our models are accurate enough to be used in QoE prediction. Our analysis also sheds light on the novel QoE enhancement strategies. The results presented here can be used by the base station to "recommend" the prefetching parameters to the media player, and to guide the admission control and the scheduling algorithms.

The remainder of this paper is organized as follows. Section II describes the system model and the QoE metrics. Section III presents the analytical framework of QoE with flow dynamics. Section IV extends the mathematical framework to include the impact of fast fading. The accuracy of our models is validated in Section V. Section VI eventually concludes the paper.

II. Problem Description and Model

In this section, we first describe our motivation, then define the metrics of QoE for media streaming service, and present a queueing model for the playout buffer at a user.

A. Motivation and Network Description

We consider a wireless data network that supports a number of flows.When a new flow "joins" the network, it requests the streaming service from a media server. After the connection has been built, the streaming packets are transmitted through the base station (BS). The streaming flows have *finite* sizes, which means that a flow "leaves" the network once the transmission completes. Note that each active user cannot watch more than one streams at the mobile device simultaneously. Hence, we use the terms "flow" and "user" interchangeably.

In wireless data networks, a streaming flow may traverse both wired and wireless links, whereas the BS is the bottleneck for the sake of limited channel capacity. In other words, the queue of an *active* flow is always backlogged at the BS. This assumption holds because most of Internet streaming servers use TCP/HTTP protocols to deliver streaming packets. The TCP protocol in the transport layer exploits the available bandwidth by pumping as more packets as possible to the BS. The BS can easily perform per-flow congestion control to limit TCP sending rate to avoid buffer overflow (a small number of concurrent flows in total). The adaptive coding and modulation in the physical layer, and ARQ scheme at the MAC layer can effectively avoid TCP packet loss. Due to these reasons, we do not consider TCP packet losses in our system.

Streaming flows may experience fast fading and normalized signal-to-noise ratio (NSNR) scheduling is usually adopted to achieve multiuser diversity with the consideration of fairness [11], [12]. The scheduling duration is commonly around 2ms [10]. NSNR selects the user that has the largest ratio of SNR compared with its mean SNR. It is similar to the well-known proportional fair (PF) scheduler in that they both attempt to achieve channel access-time fairness. We consider NSNR instead of PF for two reasons. First, the moments of throughput of PF do not have explicit results, even asymptotic ones (see [12] and references therein) when the channel capacity is computed according to the Shannon theorem. Second, NSNR needs the knowledge of the average SNR that can be obtained from the history information. The PF scheduler may cause the slow throughput convergence problem if the initial average throughput is not configured wisely.

At the user side, incoming bits are reassembled into video *frames* step by step. These video frames are played with a deterministic rate, e.g. 25 frames per second (fps) in the TV and movie-making businesses. The size of a frame is determined by the video codec, i.e. a high definition video streaming or a complex video scenario require more bits to render each frame. We consider constant bit-rate (CBR) streaming in which a codec's output data is consumed with the constant rate (i.e. the same size of frames).

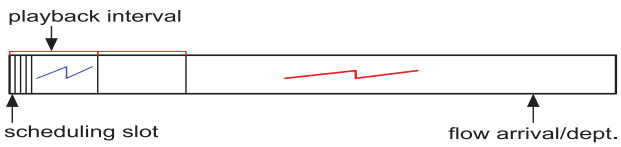

Fig. 1. Illustration of 3 different time scales

We highlight the properties of the streaming system briefly to facilitate the mathematical modeling. In our system, there exist three time scales shown in Fig.1: i) the scheduling duration (e.g. 2ms); ii) playback interval (e.g. 40ms for a video frame rate of 25fps), and iii) duration of flow dynamics (lasting about tens of seconds). The scheduler and the media player do not work at the same granularity of time scale and job size.

B. QoE Metrics

There exist five industry-standard video quality metrics. Authors in [13] summarize them into five terms: *join time*, *buffering ratio*, *rate of buffering events*, *average bitrate* and *rendering quality*. The first three metrics reflect the fundamental tradeoff in designing the prefetching process. The last two metrics are concerned with source coding. For analytical convenience, we redefine the QoE metrics regarding "prefetching" process.

- **Start-up delay:** The start-up delay denotes the duration (measured in seconds) between the time that a user initiates a session and the time that the media player starts playing video frames. In the initial prefetching phase, the player starts until the duration of received video reaches the *start-up threshold* measured in seconds of video segment. The users are impatient to wait for a very long start-up delay. Once the starvation event happens, the player pauses and resumes until the rebuffered video duration reaches the *rebuffering threshold*. We use the term *rebuffering delay* to differentiate the rebuffering time from the initial start-up delay.
- **Starvation probabilities:** When the playout buffer of a user becomes empty before the video has been completely played, we call this event a *starvation*. The starvation is very annoying to users. We adopt the starvation probability to evaluate the influence of the start-up threshold. In addition, if the rebuffering process is taken into account, we analyze the probabilities of having a certain number of starvations.

Note that the start-up delay and the starvation probabilities can be used to compute the QoE metrics in [13]. The expected number of starvations is the sum of the products of the number of starvations and its probability. The expected buffering time equals to the product of the start-up delay in each rebuffering and the mean number of starvation events (including the initial prefetching).

C. Basic Queueing Model of Playout Buffer

We consider a wireless cellular network that supports up to K simultaneous flows. The purpose of admission control is to avoid the overloading of the cell. We make the following assumptions:

- **Single user type and static channel:** We begin with the case where streaming users coexist in a static channel, as this provides an easier route to understand the developed QoE evaluation model. The impact of fast fading is added in section IV. We also consider that all the flows have the same SNR, and hence, in a static channel case, identical throughput. The extension to multiple user classes is presented in Section VI.
- **Exponentially distributed video duration:** The video duration, measured in seconds, is exponentially distributed with mean $1/\theta$. Though the exponential distribution is not the most realistic way to describe video duration, it reveals the essential features of the system, and is the first step for more general distributions (i.e. hyper-exponential distribution in Section VI).
- **Processor sharing at the BS:** The scheduling slot is very small (e.g. $\leq$2ms in 3G LTE) compared with the service interval between two video frames (e.g. 40ms at 25fps) in the playout buffer. This property enables us to treat the BS as an egalitarian processor sharing queue where all the flows are served simultaneously. Hence, the per-flow throughput, depicted in continuous time, is a deterministic step-wise function of the number of active users in the static channel (e.g. [17]).
- **Continuous time playback:** The service of video contents is regarded as a continuous process, instead of a discrete rendering of adjacent video frames spaced by a fixed interval. This assumption is commonly used (see [18]) and is validated by simulations in this work.

We denote by λ the arrival rate of new video streams. Let $Bitrate$ be the playback speed of video streams in bits per-second, and C (in bps) be the capacity of the static wireless channel. Given the exponential distribution of video duration, the file size F (measured in bits) is also exponentially distribution with mean $1/\theta_F = Bitrate/\theta$. Therefore, the dynamics of coexisting flows in the cell can be depicted as a continuous time Markov chain with a finite state space.

We concentrate on one "tagged" flow in order to gain the insight of dynamics of the playout buffer. At any time t, the tagged flow sees i other flows in a finite space $S := \{0, 1, \cdots, K-1\}$. We denote by $\{I(t); t \geq 0\}$ the external environment process that influences the throughput of the tagged flow. The environmental change refers to the join of a new flow, or the departure of an existing flow. From the assumption of Poisson flow arrival and exponentially distributed flow size, we can see that $\{I(t); t \geq 0\}$ is a homogeneous, irreducible and recurrent Markov process. Let $\{\pi_i; i \in S\}$ be the stationary distribution of environmental states that will be computed in the following sections. The throughput of the tagger user is $b_i := \frac{C}{Bitrate \cdot (i+1)}$ in seconds of video contents at state i. Let $N_e(t)$ be the number of changes in the environment by time t. Denote by A_l the time that the l^{th} environmental change takes place with $A_0 = 0$ and by $I_l := I(A_l)$ the state to which the environment changes after time A_l.

We denote by $Q(t)$ the length of playout buffer *measured in seconds* of video contents at time t. In the prefetching phase, $Q(t)$ is expressed as

$Q_a(t) = \sum_{l=1}^{N_e(t)} b_{I_l}(A_l - A_{l-1}) + b_{I_{N_e(t)}}(t - A_{N_e(t)})$.

Denote by q_a the start-up threshold. The start-up delay T_a is defined as $T_a = \inf\{t \geq 0 | Q_a(t) \geq q_a\}$. The cumulative distribution of T_a is expressed as $\Psi_i(t; q_a) = \mathbb{P}\{T_a < t | I(0) = i\}$ if the tagged flow is in state i upon arrival.

When the media player starts the playback, the queueing process $\{Q(t); t \geq 0\}$ is given by

$Q_b(t) = q - t + \sum_{l=1}^{N_e(t)} b_{I_l}(A_l - A_{l-1}) + b_{I_{N_e(t)}}(t - A_{N_e(t)})$, if the time axis starts at the instant of playing. Define $c_i := b_i - 1$ for all $i \in S$. Define

$$T_b := \inf\{t \geq 0 | Q_b(t) < 0\} \tag{1}$$

to be the time of observing empty buffer. Denote by $T_e (T_e < \infty)$ the completion time of downloading of the tagged flow. If T_b is less than T_e, a starvation event happens at the playout buffer. Then, the ultimate starvation probability is computed as $W_i(q_a) = \mathbb{P}\{T_b < T_e | I(0) = i, Q_b(0) = q_a\}$. When the playback begins at state i, and stops upon an empty queue. The ultimate starvation probability is the weighted sum of

starvation probabilities at all the ergodic entry states. Next, we proceed the computations using an approach that is inspired by the ruin analysis in actuarial theory [14], [15].

III. Impact of flow level dynamics on QoE

In this section, we model the starvation probability and the prefetching delay in a static channel where the media flows join and leave the system dynamically. The key idea is to investigate the queueing process of one "tagged" flow on the basis of differential equations.

A. Markov models of flow dynamics

Our purpose here is to construct two Markov chains to characterize the dynamics of the number of active flows. The first one models flow dynamics before the "tagged" flow joins in the network. Based on this Markov process, we can compute the stationary distribution of the number of active flows observed by the "tagged" flow at the instant when it is admitted. The second one describes the flow dynamics after the tagged flow is admitted. This Markov process enables us to investigate how the playout buffer of the tagged user changes.

We first look into the flow dynamics before the tagged flow joins. When the normalized SNR scheduler is used, the per-flow throughput is proportional to the reciprocal of flow population. Given the Poisson arrival rate and the exponentially distributed service time, we can model the dynamics of flow population as a finite-state Markov chain $\mathbf{Z}_a := \{0, 1, \cdots, K\}$ (illustrated in our technical report [19]). The transition rate from i to $i-1$ is $\mu_i := C\theta_F$. Note that the network capacity is a constant in the static channel. Hence, we let $\mu_i = \mu$ for $i = 1, 2, \cdots, K$ and $\mu_0 = 0$. Define $\rho := \frac{\lambda}{\mu}$ to be the load of the channel. Let z_i^a be the stationary probability that there exist i flows. We give the expression of z_i^a $(i \in S \cup \{K\})$ directly because it is easy to compute: $z_0^a = \frac{1-\rho}{1-\rho^{K+1}}$, and $z_i^a = \frac{\rho^i(1-\rho)}{1-\rho^{K+1}} \forall i = 1, \cdots, K$. The tagged user cannot be admitted at state K due to the admission control at the BS. Therefore, if it joins in the network successfully, it will observe i other flows with the probability $\pi_i := \frac{z_i^a}{1-z_K^a} = \frac{\rho^i(1-\rho)}{1-\rho^K}, \quad \forall i \in S$.

After the tagged flow joins in the network, the Markov process $\mathbf{Z}_a$ has been altered. The states are the number of flows observed by the tagged user, and the transition rates are conditioned on the presence of the tagged flow. Therefore, we model the flow dynamics observed by the tagged flow through a finite-state Markov chain $\mathbf{Z}_b := \{0, 1, \cdots, K-1\}$ in Fig.2. Denoted by ν_i the transition rate from state i to $i-1$. The per-flow throughput at state i is $\frac{C}{(i+1)}$ so that there has $\nu_i := \frac{iC\theta_F}{(i+1)} = \frac{i}{i+1}\mu$ for all $i \in S$.For the simplicity of notations, we denote by λ_i the transition rate from state i to $i+1$. It is obvious to have $\lambda_i = \lambda$ for all $i \neq K-1$ and $\lambda_{K-1} = 0$.

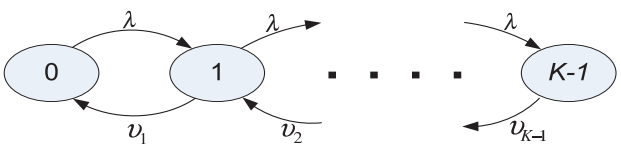

Fig. 2. Flow dynamics observed by tagged flow

B. Modeling prefetching delay distribution

We want to know how long time the tagged user needs to wait in the prefetching phase. Recall that q_a is the start-up threshold. The prefetching time is only meaningful when the video duration is **longer** than q_a. In the prefetching phase, because the playout buffer does not serve video frames, the queue length of the tagged flow evolves in an infinitesimal time interval $[0, h]$ with $h(>0)$ $Q(t+h) = Q(t) + b_i h$. The distribution of the prefetching time is difficult to solve directly. We resort to the following duality problem:

> Duality Problem: What is the starvation probability by time t if the queue is depleted with rate $b_i (i \in S)$ and the duration of prefetched contents is q_a?

In the duality problem, the queue dynamics in $[0, h]$ is modified as $\tilde{Q}(t+h) = \tilde{Q}(t) - b_i h$. We define $U_i(q, t)$ $(\forall i \in S)$ to be the probability of starvation before time t, conditioned on the entry state i and the initially prefetched content q. We use differential equations to obtain $U_i(q, t)$. In the infinitesimal time interval $[0, h]$, there are four possible events
- no change of the concurrent flows;
- arrival of one flow;
- departure of one flow (not the tagged one);
- occurrence of more than one events.

Conditioned on the events occurred in $[0, h]$, we have

$$\begin{aligned} U_i(q,t) \;=\; & (1-\lambda_i h-\nu_i h)U_i(q-b_i h, t-h) \\ & +\lambda_i h U_{i+1}(q-b_i h, t-h) \\ & +\nu_i h U_{i-1}(q-b_i h, t-h) + o(h), \;\; \forall i \in S. \end{aligned} \tag{2}$$

The above equations yield for $i \in S$

$$\begin{aligned} 1/h \cdot (U_i(q,t) - U_i(q-b_i h, t-h)) = & \lambda_i U_{i+1}(q-b_i h, t-h) \\ & -(\lambda_i+\nu_i)U_i(q-b_i h, t-h) \\ & +\nu_i U_{i-1}(q-b_i h, t-h) + o(h)/h. \end{aligned} \tag{3}$$

When $h \to 0$, the left side of eq.(3) is the partial differentials of $U_i(q, t)$ over q and t. In other words, eq.(3) yields a set of linear partial differential equations (PDEs)

$$\begin{aligned} (\partial U_i/\partial t) \;=\; & -b_i \cdot (\partial U_i/\partial q) - (\lambda_i+\nu_i)U_i(q,t) \\ & +\lambda_i U_{i+1}(q,t) + \nu_i U_{i-1}(q,t), \;\; \forall i \in S, \end{aligned} \tag{4}$$

with the initial condition: $U_i(q, 0) = 0, \;\forall\; q > 0$; and the boundary conditions at both sides $\forall\; t \geq 0$: $U_i(0, t) = 1$, and $\lim_{q\to\infty} U_i(q, t) = 0, \; \forall t \geq 0$. The initial condition means that the starvation cannot happen at time 0 for $q > 0$. The right-side boundary condition says that the starvation will not happen before t if the initial prefetching is large enough. Hence, the c.d.f. of start-up delay is the solution of linear PDEs by letting q be q_a. We next analyze the probability that the prefetching process starts at state i and ends at state j, for all $i, j \in S$. Define

$$V_{i,j}(q; q_a) := \mathbb{P}\{I(T_a) = j | I(0) = i, Q(0) = q\}. \tag{5}$$

We can use the approach of obtaining $U_i(q, t)$ to solve $V_{i,j}(q; q_a)$. Note that we now back to the queueing dynamics

$Q(t+h) = Q(t) + b_i h$. In the time interval $[0, h]$, there exists for all $i, j \in S$

$$V_{i,j}(q; q_a) = (1 - \lambda_i h - \nu_i h) V_{i,j}(q + b_i h; q_a) + \lambda_i h V_{i+1,j}(q + b_i h; q_a) + \nu_i h V_{i-1,j}(q + b_i h; q_a). \quad (6)$$

It is easy to see that $V_{i,j}(q; q_a)$ is the solution of the following differential equation

$$b_i \dot{V}_{i,j}(q; q_a) = (\lambda_i + \nu_i) V_{i,j}(q; q_a) - \lambda_i V_{i+1,j}(q; q_a) - \nu_i V_{i-1,j}(q; q_a), \ \forall i, j \in S, \quad (7)$$

with the boundary conditions for all $i, j \in S$: $V_{i,j}(q_a; q_a) = 1$ if $i = j$; $V_{i,j}(q_a; q_a) = 0$, otherwise.

We interpret the boundary condition in the following way. If there exist $I(0) = i$ and $Q(0) = q_a$, the prefetching duration is 0 and the prefetching process ends at state i. Hence, $V_{i,j}(q_a; q_a)$ is 1 iff i equals to j. We use Laplace Transform (LT) to solve eq.(7). Define a matrix $\mathbf{M}_V$ as below:

$$\begin{pmatrix} -\frac{\lambda_0+\nu_0}{b_0} & \frac{\lambda_0}{b_0} & 0 & \cdots & 0 & 0 \\ \frac{\nu_1}{b_1} & -\frac{\lambda_1+\nu_1}{b_1} & \frac{\lambda_1}{b_1} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \cdots & \frac{\nu_{N-1}}{b_{N-1}} & -\frac{\nu_{N-1}+\lambda_{N-1}}{b_{N-1}} \end{pmatrix}$$

Define $\mathbf{1}_j$ to be a column vector in which the j^{th} element is 1 and all other elements are 0. Taking LT of eq.(6), we obtain

$$b_i s V_{i,j}(s) = (\lambda_i + \nu_i) V_{i,j}(s) - \lambda_i V_{i+1,j}(s) - \nu_i V_{i-1,j}(s) + b_i V_{i,j}(0; q_a), \ \forall i, j \in S, \quad (8)$$

where $V_{i,j}(s)$ is the laplace transform of $V_{i,j}(q; q_a)$. Thus, we obtain the vector of probabilities that the prefetching process ends at state j

$$\mathbf{V}_j(0; q_a) = [\mathcal{L}^{-1}\{(sI + \mathbf{M}_V)^{-1}\}(q_a)]^{-1} \cdot \mathbf{1}_j, \ \forall j \in S, \quad (9)$$

where $\mathcal{L}^{-1}\{\}$ denotes the inverse LT. For the computation of eq. (9), interested readers can refer to the technical report [19].

C. Modeling starvation probability

The modeling of starvation probabilities should take into account the departure of the tagged flow. Recall that the CTMC in Fig. 2 assumes the persistent tagged flow, which is not suitable for the playback process. Before solving the starvation probabilities, we first modify the original CTMC by adding an absorbing state **A** (illustrated in our technical report [19]). The state **A** denotes the event that the tagged flow completes its downloading. Because of the exponentially distributed video duration, the transition from state i to state **A** is Poisson. Denote by $\mathbf{Z}_c := \{0, 1, \cdots, K-1, \mathbf{A}\}$ this absorbing Markov chain. Denote by φ_i the transition rate from state i to **A**. At state i, the bandwidth of a flow is $\frac{C}{i+1}$, resulting in $\varphi_i := \frac{\mu}{i+1}$. Define $c_i := b_i - 1$. The queue length of the tagged flow changes in an infinitesimal interval h according to the rule $Q(t+h) = Q(t) + c_i h$. If $c_i > 0$, the bandwidth is sufficient for continuous playback of the tagged flow and i other flows. For mathematical convenience, we suppose that q is 0^- if buffer starvation happens.When the tagged flow enters the absorbing state, it has downloaded the whole file with a non-empty playout buffer. Thus, the starvation probability at state **A** is 0 for any $q \geq 0$. Let $W_i(q)$ be the starvation probability with q seconds of contents in the playout buffer at state i. We derive a system of ordinary differential equations for $W_i(q)$. In an infinitesimal interval $[0, h]$, there are five possible events:
- all four events the same as those in the prefetching stage;
- the tagged flow entering the absorbing state.

The above events give rise to the following bunch of equations

$$W_i(q) = (1 - (\lambda_i + \mu_i)h) W_i(q + c_i h) + \lambda_i W_{i+1}(q + c_i h) + \nu_i W_{i-1}(q + c_i h) + o(h). \quad (10)$$

When $h \to 0$, we obtain

$$c_i \dot{W}_i(q) = (\lambda_i + \mu_i) W_i(q) - \lambda_i W_{i+1}(q) - \nu_i W_{i-1}(q). \quad (11)$$

Similarly, we solve the above ODEs using LT. Define $\hat{W}_i(s)$ as the LT of $W_i(q)$, and $\hat{\mathbf{W}}(s)$ as the vector $\{\hat{W}_0(s), \hat{W}_1(s), \cdots, \hat{W}_{K-1}(s)\}$. They yield for all $i \in S$

$$c_i(s\hat{W}_i(s) - W_i(0)) = (\lambda_i + \mu_i)\hat{W}_i(s) - \lambda_i \hat{W}_{i+1}(s) - \nu_i \hat{W}_{i-1}(s).$$

We write the above equations in matrix form:

$$(sI + \mathbf{M}_W) \cdot \hat{\mathbf{W}}(s) = \mathbf{W}(0) \quad (12)$$

where $\mathbf{M}_W$ is expressed as the following:

$$\begin{pmatrix} -\frac{\lambda_0+\mu_0}{c_0} & \frac{\lambda_0}{c_0} & 0 & \cdots & 0 & 0 \\ \frac{\nu_1}{c_1} & -\frac{\lambda_1+\mu_1}{c_1} & \frac{\lambda_1}{c_1} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \cdots & \frac{\nu_{N-1}}{c_{N-1}} & -\frac{\mu_{N-1}+\lambda_{N-1}}{c_{N-1}} \end{pmatrix}$$

and $\mathbf{W}(0)$ is the vector of initial state. In order to compute $\hat{W}_i(s)$, we first need to find out $W_i(0)$ ($\forall i \in S$), the starvation probability without prefetching. However, eq.(12) does not give explicit expressions for $W_i(0)$.

Theorem 1: Let $\mathrm{adj}(sI + \mathbf{M}_W)$ be the adjugate matrix of $sI + \mathbf{M}_W$. $\{W_i(0)\}$ are solved by
- $W_i(0) = 1$ if $c_i < 0$, $\forall i \in S$;
- $W_i(0) = 0, \forall i \in S$ if $c_{K-1} \geq 0$;
- $\mathrm{adj}(sI + \mathbf{M}_W) \cdot \mathbf{W}(0)|_{s=\xi_l} = 0$, $\forall\ l = 0, 1, \cdots, m$, $m < K-1$, if $\{\xi_l\}$ are positive roots of the determinant $|sI + \mathbf{M}_W|$.

Proof: Please refer to the technical report [19]. ∎

Once the vector $\mathbf{W}(0)$ is solved, the LTs of starvation probabilities can be computed by $\hat{\mathbf{W}}(s) = [sI + \mathbf{M}_W(s)]^{-1} \cdot \mathbf{W}(0)$. Using inverse LT, we obtain the starvation probability $\mathbf{W}(q) := \mathcal{L}^{-1}\{[\mathbf{M}_W(s)]^{-1}\}(q) \cdot \mathbf{W}(0)$. Substituting q by q_a, we derive the starvation probabilities $\mathbf{W}(q_a)$.

Next, we build a bridge to interconnect the prefetching threshold and the starvation probability function $W_i(q)$. For a given prefetching threshold q_a, the starvation event takes place only when the video duration T_{video} is longer than q_a. Thus a flow with $T_{video} > q_a$ can be regarded as a tagged flow. When the prefetching process is finished, the tagged flow enters the playback process. Conditioned on the distribution of entry states $\{\pi\}$, the distribution of the states that the playback process begins (or the prefetching process ends) is computed by $\pi \cdot \mathbf{V}(0; q_a)$. Then, the starvation probability with the prefetching threshold q_a is obtained by

$$P_s(q_a) = \mathbb{P}\{T_{video} > q_a\} \cdot \pi \cdot \mathbf{V}(0; q_a) \cdot \mathbf{W}(q_a) = \exp\big(-\theta q_a\big) \cdot \pi \cdot \mathbf{V}(0; q_a) \cdot \mathbf{W}(q_a). \quad (13)$$

D. Modeling P.G.F. of starvation events

When a starvation event happens, the media player pauses until q_b seconds of video contents are re-buffered. A more interesting but challenging problem is how many starvations may happen in a streaming session. In this section, we come up with an approach to derive the probability generating function of starvation events.

We define a *path* as a sequence of prefetching and starvation events, as well as the event of completing the downloading. Obviously, the probability of a path depends on the number of starvations. We illustrate a typical path with L starvations in figure 3 that starts from a prefetching process and ends at a playback process. We denote by I_l^A the beginning state of the l^{th} prefetching, by I_l^B the beginning state of the l^{th} playback, and by I_e the end of downloading. The end of a prefetching process is exactly the beginning of a playback process. The end of a playback process is also the beginning of a subsequent prefetching process if the video has not been downloaded completely. This path contains a sequence of events happening at the states $\{I_1^A, I_1^B, I_2^A, I_2^B, \cdots, I_{L+1}^A, I_{L+1}^B, I_e\}$. The process between I_l^A and I_l^B is the l^{th} prefetching process, while that between I_l^B and I_{l+1}^A is the l^{th} playback process, $(1 \le l \le L)$. The first starvation takes place at the instant that the second prefetching process begins. The starvation event (e.g. $I_l^B,\ 1 \le l \le L$) cannot happen at the state i that has $c_i \ge 0$.

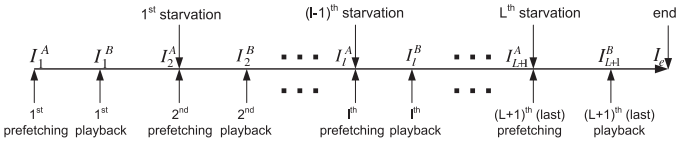

Fig. 3. A path with L starvations

The sample path in figure 3 demonstrates a roadmap to find the p.g.f. of starvation events. We need to compute the transition probability along the path with all possible states. Recall that the transition probabilities from state I_l^A to I_l^B have been computed in section III-B. The only missing part is the transition probabilities from state I_l^B to I_{l+1}^A.

Denote by $X_{i,j}(q)$ the probability that a playback process starts at state i and meets with the empty buffer at state j with the prefetching threshold q. Define a matrix $\mathbf{X}(q) := \{X_{i,j}(q); i,j \in S\}$. Denote by $\mathbf{X}_j(q)$ the vector of probabilities that the starvation takes place at state j with the prefetching threshold q, i.e. $\mathbf{X}_j(q) := [X_{0,j}(q), \cdots, X_{K-1,j}(q)]^T$. Let $\mathbf{X}_j(0) := [X_{0,j}(0), \cdots, X_{K-1,j}(0)]^T$ be the vector of those probabilities without the prefetching. Using the same argument, we get the differential equation of $X_{i,j}(q)$, $\forall i,j \in S$,

$$c_i \dot{X}_{i,j}(q) = (\lambda_i + \mu_i) X_{i,j}(q) - \lambda_i X_{i+1,j}(q) - \nu_i X_{i-1,j}(q). \quad (14)$$

The solution of eq.(14) is directly given by

$$\mathbf{X}_j(q) = \mathcal{L}^{-1}\{[s\boldsymbol{I} + \mathbf{M}_W]^{-1}\}(q) \cdot \mathbf{X}_j(0). \quad (15)$$

The computation of $\mathbf{X}_j(q)$ requires the knowledge of the boundary condition $\mathbf{X}_j(0)$.

Theorem 2: The probability that the playback process starts at state i and ends at state j without prefetching is computed by

- $X_{i,j}(0) = 0,\ i \ne j$ and $X_{i,j}(0) = 1$ if $c_i < 0$;
- $X_{i,j}(0) = 0, \forall\ i \in S$ if $c_{K-1} \ge 0$;
- The third item of Theorem 1.

Proof: The proof is the same as that of Theorem 1. ∎

When replacing q by q_a, we obtain the probability $X_{ij}(q_a)$ that the first starvation happens at state j with i other flows observed by the tagged flow at the beginning of the playback process. The starvation probability in a rebuffering process is calculated by $X_{ij}(q_b)$, given the rebuffering threshold q_b.

The probability of having L starvations can be expressed as the product of the probabilities from the first prefetching to the last playback. The probability vector from I_1^A to I_1^B is obtained by

$$\{\mathbb{P}_{I_1^A \to I_1^B}\} = \pi \cdot \exp\big(-\theta q_a\big) \cdot \mathbf{V}(0; q_a),\ \forall I_1^A, I_1^B \in S. \quad (16)$$

The probability vector from I_1^A to I_2^A is,

$$\begin{aligned}\{\mathbb{P}_{I_1^A \to I_2^A}\} &= \{\mathbb{P}_{I_1^A \to I_1^B}\} \cdot \mathbf{X}(q_a) \\ &= \pi \cdot \exp\big(-q_a\theta\big) \cdot \mathbf{V}(0; q_a) \cdot \mathbf{X}(q_a), \forall I_1^A, I_2^A \in S \quad (17)\end{aligned}$$

Recall that the starvation happens at state I_2^A, and the rebuffering process ends at state I_2^B with the prefetched video duration q_b. We next compute the probability of having only one starvation denoted by $\mathbb{P}_{1\text{starv}}$. The possible paths include $\{I_1^A, I_1^B, I_2^A, I_e\}$ and $\{I_1^A, I_1^B, I_2^A, I_2^B, I_e\}$. The first part of $\mathbb{P}_{1\text{starv}}$ refers to the case that the remaining video duration is less than the rebuffering threshold q_b. The second part refers to the case that the remaining video duration is longer than q_b and there is no starvation after the rebuffering process. $\mathbb{P}_{1\text{starv}}$

$$\begin{aligned}&= \{\mathbb{P}_{I_1^A \to I_2^A}\} \cdot \mathbf{1} \cdot \big(1 - \exp(-q_b\theta)\big) + \{\mathbb{P}_{I_1^A \to I_2^B}\} \cdot (1 - \mathbf{W}(q_b)) \\ &= \pi \cdot \exp\big(-q_a\theta\big) \cdot \mathbf{V}(0; q_a) \cdot \mathbf{W}(q_a) \cdot \big(1 - \exp(-q_b\theta)\big) + \\ &\pi \cdot \exp\big(-(q_a+q_b)\theta\big) \cdot \mathbf{V}(0; q_a)\mathbf{X}(q_a)\mathbf{V}(0; q_b) \cdot (1 - \mathbf{W}(q_b)). \quad (18)\end{aligned}$$

Here, the expression $(1 - \mathbf{W}(q_b))$ is the probability $I_2^A \to I_e$ in the first path and the expression $(1 - \mathbf{W}(q_b))$ is that of $I_2^B \to I_e$ in the second path. Similarly, we can deduce the probability of having $L(L > 1)$ starvations recursively by $\mathbb{P}_{\text{Lstarv}}$

$$\begin{aligned}&= \{\mathbb{P}_{I_1^A \to I_{L+1}^A}\} \cdot \mathbf{1} \cdot \big(1 - \exp(-q_b\theta)\big) + \{\mathbb{P}_{I_1^A \to I_{L+1}^B}\} \cdot (1 - \mathbf{W}(q_b)) \\ &= \pi \cdot \exp\big(-q_a\theta\big) \cdot \mathbf{V}(0; q_a)\mathbf{X}(q_a)\Big(\exp\big(-q_b\theta\big)\mathbf{V}(0; q_b)\mathbf{X}(q_b)\Big)^{L-1} \\ &\cdot \mathbf{1} \cdot \big(1 - \exp(-q_b\theta)\big) + \pi \cdot \exp\big(-(q_a + q_b)\theta\big) \cdot \mathbf{V}(0; q_a)\mathbf{X}(q_a) \cdot \\ &\cdot \Big(\exp\big(-q_b\theta\big)\mathbf{V}(0; q_b)\mathbf{X}(q_b)\Big)^{L-1} \cdot \mathbf{V}(0; q_b) \cdot (1 - \mathbf{W}(q_b)). \quad (19)\end{aligned}$$

Though the expression in eq.(19) looks complicated, it only involves duplicated products of matrices with dimension K that can be calculated easily.

IV. Impact of fast fading on QoE

This section models the starvation behavior of CBR streaming when users experience fast channel fading. We compute the first two moments of bit arrival process and show how these parameters can be fed into our analytical framework.

Network description. Due to the change of radio condition (e.g. user mobility, or a car passing by the user), the signal

strength is no longer a constant at different scheduling slots. To explore the multiuser diversity gain, the base station adopts the normalized SNR scheduling algorithm for allocating time slots to coexisting flows.

We begin with the scenario with a fixed population of i users (or flows) served by a single base station. In each slot, the users measure their channel qualities and feedback them to the BS. Based on the channel quality indications, the BS transmits to only one of the users every slot. Denote by $\gamma_{j,n}$ the instantaneous signal to noise ratio (SNR) of user j, $(1 \le j \le i)$, at slot n. As stated in most of previous work, we assume that all the users experience Rayleigh fast-fading. Denote by $\bar{\gamma}_j$ the average SNR of user j. Then, the received SNR of user j is an exponentially distributed random variable with the following probability density function $g_j(\gamma) = \frac{1}{\bar{\gamma}_j}\exp(-\frac{\gamma}{\bar{\gamma}_j})$. The NSNR scheduler selects the user that has the highest relative SNR for transmission, $j_n^* = \max_j\{\gamma_{j,n}/\bar{\gamma}_j,\ \ j = 1, 2, \cdots, i\}$, where j^* is the scheduled user at slot n. In this section, we consider the case of homogeneous average SNRs (i.e. $\bar{\gamma}_j = \bar{\gamma}$ for all j). Therefore, the NSNR scheduler is equivalent to the maximum sum rate (MSR) scheduler that gives the largest per-user throughput.Since the SNRs of different users are independently distributed, the scheduled SNR, denoted by γ^*, has the following probability density function [16] $g^*(\gamma) = \frac{i}{\bar{\gamma}}\exp(-\frac{\gamma}{\bar{\gamma}})\big(1-\exp(-\frac{\gamma}{\bar{\gamma}})\big)^{i-1}$. Denote by $f(\gamma)$ the data rate of a user with the SNR γ. Here, $f(\cdot)$ can be a linear function in the low-SNR regime and a logarithmic function in the high SNR regime if the modulation scheme is continuous. For discrete modulations, $f(\cdot)$ is a step function of γ. Without loss of generality, we let $f(\gamma) = \log_2(1+\gamma)$.

Analysis of throughput process.

The fast fading along with NSNR scheduling brings variation of bit arrivals to the receiver. Here, our purpose is to show that the throughput variance may have negligible influence on the computation of starvation behaviors. In what follows, we calculate the mean throughput and its variance measured in video duration. To achieve this goal, we shall obtain the mean throughput and its variance measured in bits first.

Denote by r_i^* the transmission rate of the user with the best SNR at a slot in each Hz when there are i active flows in the cell. Denote by r_i the transmission rate to *one particular flow* at a slot per Hz. Given the assumption that all the flows have the same average SNR, each flow has the equal probability of being scheduled. Hence, we can see

$$r_i := \begin{cases} r_i^* & \text{w.p. } \frac{1}{i}; \\ 0 & \text{w.p. } \frac{i-1}{i}. \end{cases}$$

For the r.v. r_i^*, its mean and variance are computed by

$E[r_i^*] = \int_0^\infty f(\gamma)\cdot g^*(\gamma)d\gamma,$

$\mathrm{Var}[r_i^*] = \int_0^\infty f(\gamma)^2\cdot g^*(\gamma)d\gamma - (E[r_i^*])^2.$

The above equations yield:

$E[r_i] = \frac{1}{i}E[r_i^*],$

$\mathrm{Var}[r_i] = E[r_i^2] - (E[r_i])^2 = \frac{1}{i}E[(r_i^*)^2] - \frac{1}{i^2}(E[r_i^*])^2$
$= \frac{1}{i}\mathrm{Var}[r_i^*] + (E[r_i^*])^2(\frac{1}{i} - \frac{1}{i^2}).$

Denote by D_s the duration of scheduling slot (usually 2ms), and by B the width of wireless spectrum in Hz. Then, the mean and the variance of per-flow throughput measured in the *duration of video contents* are $\frac{B\cdot D_s\cdot E[r_i]}{Bitrate}$ and $(\frac{B\cdot D_s}{Bitrate})^2\cdot \mathrm{Var}[r_i]$ respectively in one slot.

Let R_i be the r.v. of *per-flow throughput in one second* that is measured by the duration of video contents. In one second, the total throughput of a flow at one Hz is the sum of throughput in $\frac{1}{D_s}$ slots. Therefore, the r.v. R_i is the sum of $\frac{1}{D_s}$ i.i.d. r.v.s corresponding to the per-slot throughput. We can express the mean and the variance of R_i as follows:

$E[R_i] = \frac{1}{D_s}\cdot\frac{B\cdot D_s\cdot E[r_i]}{Bitrate} = \frac{B\cdot E[r_i^*]}{i\cdot Bitrate},$

$\mathrm{Var}[R_i] = \frac{1}{D_s}\cdot(\frac{B\cdot D_s}{Bitrate})^2\cdot\mathrm{Var}[r_i]$
$= \big(\frac{1}{i}\mathrm{Var}[r_i^*] + (E[r_i^*])^2(\frac{1}{i}-\frac{1}{i^2})\big)\cdot\frac{B^2\cdot D_s}{Bitrate^2}.$

In general, the frequency width B is 1$\sim$5 MHz, the bit-rate is usually greater than 200 Kbps, and D_s equals to 0.002s. Then, $\mathrm{Var}[R_i]$ is usually at the order of 10^{-2}. If starvation happens at state i, $E[R_i]$ is usually less than 1, which means that $\frac{B}{Bitrate}$ needs to be small. However, the small $\frac{B}{Bitrate}$ results in the small variance $\mathrm{Var}[R_i]$. This is to say, if the variance of bit arrival process is large, there might not exist starvations. On the contrary, if the starvations appear, the variance is usually small so that its impact on the starvation is negligible. For this reason, we directly use the framework without diffusion approximation to model the streaming QoE in a fast fading channel.

Markov model of flow dynamics To analyze the interaction between NSNR scheduling and the flow dynamics, a fluid-level capacity model is required. When the average SNR of all active users are the same, the per-flow throughput in each slot is i.i.d. and only depends on the quantity of flows (see section VI.A in [19]). Given the exponentially distributed video size, we can model the flow dynamics as a Markov process.

The Markov processes $\{\mathbf{Z}_a\}$, $\{\mathbf{Z}_b\}$, and $\{\mathbf{Z}_c\}$, contain transitions rates such as μ_i, ν_i and φ_i. However, it is not direct to feed the parameters of this section into the above Markov processes. In $\{\mathbf{Z}_a\}$, state i refers to the number of flows in the system. The departure rate is computed by $\mu_i = i\theta E[R_i]$ for $i \in S\cup\{K\}$, recalling that $E[R_i]$ is average per-user throughput in video duration per second. It is easy to obtain the stationary distribution of having i flows by

$$z_i^a = \frac{\lambda^i}{\prod_{l=1}^i\mu_l}\left[1+\sum_{j=1}^K\frac{\lambda^j}{\prod_{l=1}^j\mu_l}\right]^{-1}, \quad \forall i = 0,\cdots,K,$$

(with the convention that $\prod$ over an empty set is 1). When a tagged user joins in the system and is also admitted, it observes i other flows with the following stationary distribution $\{\pi\}$:

$$\pi_i = \frac{z_i^a}{1-z_K^a} = \frac{\frac{\lambda^i}{\prod_{l=1}^i\mu_l}}{1+\sum_{j=1}^{K-1}\frac{\lambda^j}{\prod_{l=1}^j\mu_l}}, \quad \forall i\in S.$$

The Markov processes $\{\mathbf{Z}_b\}$ and $\{\mathbf{Z}_c\}$ are conditioned on the existence of the tagged flow. At state i, the per-user throughput is $E[R_{i+1}]$ because there are i flows plus the tagged one. Hence, the transition rate ν_i is computed by $\nu_i := i\theta\cdot E[R_{i+1}]$ for all $i\in S$. The transition rate φ_i is expressed as $\varphi_i := \theta\cdot E[R_{i+1}]$. Define $\tilde{\mu}_i$ as the total departure rate at state i that has

$$\tilde{\mu}_i := \varphi_i + \nu_i = (i+1)\theta E[R_{i+1}] = \mu_{i+1}, \tag{20}$$

in the presence of the tagged flow. The constants b_i and c_i are obtained by

$$b_i = E[R_{i+1}] \quad \text{and} \quad c_i = b_i - 1, \ \ \forall i \in S. \tag{21}$$

Substituting the above parameters to the framework in section III, we can derive the approximated QoE metrics in a fast fading channel with flow dynamics.

V. Simulation

In this section, we compare the numerical experiments with the developed framework using MATLAB. Our models are shown to match the simulations accurately.

A. Flow dynamics without fast fading

We consider a network with maximum number of ten simultaneous streaming flows and the capacity of 2.5Mbps. Flows arrive to the network with a Poisson rate $\lambda = 0.12$. Let the video duration be exponentially distributed with the mean 60 seconds. Then, there have $\mu = 0.1302$ and $\rho = 0.9216$ at the playback rate 360Kbps, and $\mu = 0.0868$ and $\rho = 1.3824$ at the playback rate 480Kbps. The simulation lasts 5×10^5s.

Starvation probabilities: In this set of experiments, we will illustrate the overall starvation probability, the starvation probabilities when the playback process begins at different states, as well as the p.g.f. of starvation events. Fig. 4 shows the overall starvation probabilities with different settings of the start-up threshold. When it increases from 0 to 20s of video contents, the starvation probability decreases. The higher playback rate (e.g. 480Kbps) incurs larger starvation probabilities in comparison with the lower playback rate (e.g. 360Kbps). Our mathematical models match the simulations very well. Figure 5 compares the starvation probabilities when the playback process begins at different states. A higher state refers to more coexisting flows (or congestions), and hence causing a larger starvation probability. Note that the arrival rates at state 7 and 9 are less than 360Kbps. Without prefetching, the starvation event happens for sure. We evaluate the probabilities of having one or two starvations in the whole procedure in Fig.6. For clarity, we choose the same value for start-up and re-buffering thresholds. The starvation probabilities increase in the beginning and decrease afterwards when q_a (or q_b) increases from 0 to 30s. This is because there are many starvations with very small start-up threshold and few starvations with very large start-up threshold. Our analytical model predicts the starvation probabilities accurately.

Start-up delay: We illustrate the distribution of start-up delays in Fig.7. The start-up threshold is set to 10s. We highlight the c.d.f. curves when the tagged flow sees $3, 5$, and 7 other flows respectively after entering the network. We use MATLAB PDE function *pdepe* to compute the model in eq.(4) numerically. Fig.7 demonstrates accurate estimation of start-up delay in the simulation. When the cumulative probability is close to 1, the PDE model oscillates slightly. This is because $U_i(0,0)$ is discontinuous in the boundary and initial conditions. The elimination of this oscillation demands an iterative solution of linear PDEs, which will be our future work.

B. Rayleigh Fading Channel

Consider a wireless channel with frequency width of 1MHz. The average SNRs of users is 5dB. The base station allows at most 10 flows simultaneously, and schedules the transmission to one of them in every slot of duration 0.002s. The video duration is exponentially distributed with the mean of 90 seconds and the video bit rate is chosen to be 480Kbps. Then, the mean throughput are $\{3.5749,\ 2.3702,\ 1.7844,\ 1.4369,\ 1.2061,\ 1.0412,\ 0.9174,\ 0.8207,\ 0.7432,\ 0.6794\}$ times the playback rate at states from 0 to 9. In other words, the mean throughput at states 6$\sim$9 are insufficient to support the continuous playback. The variances at all states are $\{0.0083,\ 0.0144,\ 0.0144,\ 0.0134,\ 0.0124,\ 0.0114,\ 0.0105,\ 0.0098,\ 0.0091,\ 0.0086\}$, which are small enough. We consider two flow arrival rates, $\lambda = 0.07$ and $\lambda = 0.09$. For $\lambda = 0.07$, the traffic load ρ is greater than 1 at states 0$\sim$5 and less than 1 at states 6$\sim$9. For the latter case, there have $\rho > 1$ at all the states. Each set of simulation lasts 2×10^7 time slots.

In Fig.8 we compare the starvation probabilities measured from a Rayleigh fading channel, and those computed from the model without considering throughput variation. The simulation matches the model quite well, which means that the flow-level dynamics have a dominant impact on the playback interruption, while the impact of throughput variation due to Rayleigh fading is negligible. In Fig.9 we examine the starvation probabilities when the playback process begins at different states. We test two start-up thresholds, $q_a = \{5, 10\}$, and two flow arrival rates, $\lambda = \{0.07, 0.09\}$. One can observe that the starvation probabilities do not differ much in high states (e.g. 8 and 9). However, the starvation probabilities in the states with mean throughput around 1 are distinguishable, in which state 6 is an example. With $\lambda = 0.09$, a tagged flow sees the congested network (more other flows) with a higher probability, and also encounters a higher probability of starvation afterwards.

VI. Conclusion

In this work, we developed an analytical framework to compute the QoE metrics of media streaming service in wireless data networks. Our framework takes into account the dynamics of playout buffer at three time scales, the scheduling duration, the video playback variation, as well as the flow arrivals or departures. We show that the proposed models can accurately predict the distribution of prefetching delay and the probability generating function of buffer starvations. The analytical results demonstrate that the flow dynamics have dominant influence on QoE metrics compared to the jittering in the throughput due to the fast fading. Our result can be used by the streaming publishers to tune the video prefetching, and by the network operator to configure admission control and scheduling algorithms.

As of future works, we aim at studying the QoE with the variation of playback process. The more heterogeneous case where users of different radio conditions or different service types share the network resources is also left as an interesting future extension.

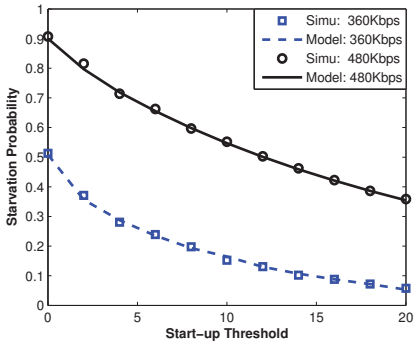

Fig. 4. Overall starvation probability VS start-up threshold

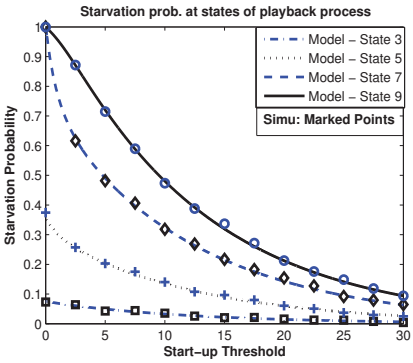

Fig. 5. Starvation probabilities at different playback states with a playback rate 360Kbps

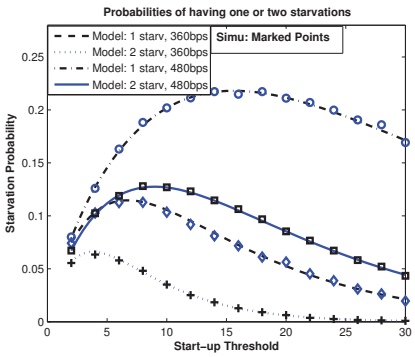

Fig. 6. Probability of observing one and two starvations

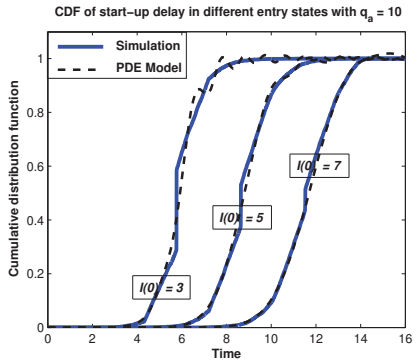

Fig. 7. CDF of start-up delay with $q_a = 10$

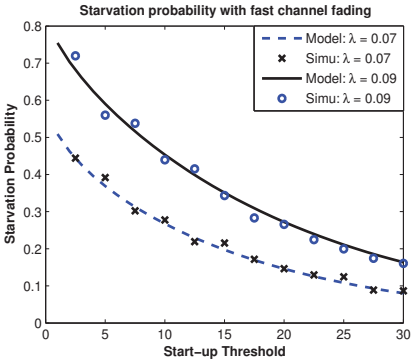

Fig. 8. Starv. probability VS start-up threshold with Rayleigh fading

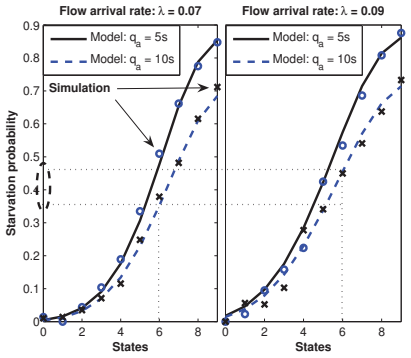

Fig. 9. Starvation probability under Rayleigh fading when different # of other users are observed initially

REFERENCES

[1] Allot MobileTrends. *http://www.allot.com*.

[2] S. Borst and N. Hegde, "Integration of Streaming and Elastic Traffic in Wireless Networks", Infocom 2007.

[3] L. Rong, S-E. Elayoubi and O. Ben Haddada, "Performance Evaluation of Cellular Networks Offering TV Services", *IEEE Transa. on Vehicular Tech.*, 2010.

[4] M. K. Karray, "Analytical evaluation of QoS in the downlink of OFDMA wireless cellular networks serving streaming and elastic traffic" *IEEE Trans. on Wireless Commun.*, 2010.

[5] Y.D. Xu, E. Altman, et. al, "Probabilistic Analysis of Buffer Starvation in Markovian Queues", *IEEE Infocom 2012*.

[6] Y.D. Xu, E. Altman, et. al, "QoE Analysis of Media Streaming in Wireless Data Networks", *IFIP Networking 2012*.

[7] Hao Luan, Lin X. Cai, and Xuemin (Sherman) Shen, "Impact of network dynamics on users' video quality: analytical framework and QoS provision" *IEEE Trans. on Multimedia*, Vol.12, No.1, pp:64-78, 2010.

[8] G. Liang and B. Liang, "Effect of delay and buffering on jitter-free streaming over random VBR channels", *IEEE Trans. on Multimedia*, Vol.10, No.6 pp:1128-1141, 2008.

[9] A. ParandehGheibi et al, "Avoiding Interruptions a QoE Reliability Function for Streaming Media Applications", *IEEE J. Sel. Areas Commun.*, Vol.29, No.5, pp:1064-1074, 2011.

[10] T. Bonald and A. Proutiere, "A Queueing Analysis of Data Networks", *Queueing Networks*, Springer, 2011.

[11] J.G. Choi and S. Bahk, "Cell-throughput analysis of the proportional fair scheduler in the single-cell environment," *IEEE Trans. Veh. Technol.*, Vol, 56, pp:766-778, 2007.

[12] G. Song and Y. Li, "Asymptotic throughput analysis for channel-aware scheduling," *IEEE Trans. Commun.*, Vol.54, No.10, pp.1827-1834, 2006.

[13] F. Dobrian, A. Awan, I. Stoica, et.al, "Understanding the Impact of Video Quality on User Engagement", *ACM SIGCOMM'2011*.

[14] S. Asmussen, *Ruin Probabilities*, World Scientific Pub. Vol.2, 2000.

[15] Y. Lu, S.M. Li, "On the probability of ruin in a Markov-modulated risk model", *Insurance: Mathematics and Economics*, No.37, pp:522-532, 2005.

[16] Y.J. Chang, F.T. Chien, and C.C. Kuo, "Cross-layer QoS Analysis of Opportunistic OFDM-TDMA and OFDMA Networks", *IEEE J. Sel. Areas Commun.*, Vol.25, 2007.

[17] S. Borst, "User-Level Performance of Channel-Aware Scheduling Algorithms in Wireless Data Networks", *Proc. of IEEE Infocom 2003*.

[18] B. Wang, W. Wei, Z. Guo and D. Towsley, "Multimedia Streaming via TCP: An Analytic Performance Study", *ACM TOMCCAP*, Vol.5, No.3, pp:1-23, 2004.

[19] Y.D. Xu, S-E. Elayoubi, E. Altman and R. Elzouzi, "Flow Level QoE of Media Streaming in Wireless Networks", *Technical Report*, 2012. Available at: **http://sites.google.com/site/yuedongxu**