Computing The Quality of Experience in Network Modeled by a Markov Modulated Fluid Model

Zakaria Ye∗, Rachid EL-Azouzi†, Tania Jimenez∗ and Yuedong Xu†
∗University of Avignon, 339 Chemin des Meinajaries, Avignon, France
†Department of Electronic Engineering, Fudan University, China
Email: {zakaria.ye, rachid.elazouzi, tania.jimenez}@univ-avignon.fr
ydxu@fudan.edu.cn

Abstract—The amount of streaming traffic has grown enormously in the last years and is expected to continue to grow. However, the Internet still provides only best-efforts and therefore the traffic streaming are distorted by fluctuations in the available bandwidth on the Internet. In this paper we develop an analytical approach to study Quality of user Experience (QoE) for streaming by considering a MMFM (Markov Modulated Fluid Model) that accurately approximates the queuing behaviour for network traffic exhibiting LRD (Long Range Dependence) behaviour. We consider the starvation probability (the probability of interruption in video playback) as well as start-up delay (initial waiting time) as the QoE metrics. We compute the distribution of starvation as well as start-up delay using partial differential equation (PDEs) and solve them using the Laplace Transform. We illustrate the results with the cases of the two-state Markov Modulated Fluid Model (MMFM) that is the most used in multimedia applications. We compare our analytical model with simulation results using ns-3 [17] under various operating parameters.

I. INTRODUCTION

According to different sources such as [1], watching streaming adds gobbles up over half the traffic on the Internet in North America. Since the smartphones introduction, mobile networks are witnessing an exponential traffic growth. This creates regimes where Internet and wireless networks are pushed to operate close to their performance limits, dictated by current architectural considerations. Much effort has been expended and significant progress has been made in recent years to increase the capacity of mobile networks, there is little progress for dealing with the user satisfaction, which is strongly related to the Quality of Experience (QoE). That is the big challenge operators face because they have to look at both the server side and the client side to make a link between the quality of service of the network and the client satisfaction which depends on the QoE. The question is which QoS metrics best represent the QoE perceived by the users and how to predict these QoE metrics for a given traffic intensity. Despite the rich literature on non-interactive real-time services such video streaming and QoE measurement, the traditional methods of measuring user experience (e.g. opinion scores) and quality of video (e.g., Peak Signal-to-Noise Ratio) have been evolved to other metrics that capture new effects by taking into account both quality and experience [5], [10]. Further the interplay between quality of video and user experience can be different for different types of content [6]. An empirical study in [11] has identified critical metrics that affect the QoE through the user engagement: Buffering Ratio, Rate of Buffering, startup delay, Rendering Quality and average Bit Rate. There are recent works addressed this problem [6] to classify important metrics that affect the QoE. Development of new models as function of these metrics can help operators and content publishers to better invest their network and server resources toward optimizing these metrics that really matter for QoE. The first step towards defining QoE and predicting it is to understand how streaming is played. Media players at the devices are equipped with a playback 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 re-buffering 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. However, the expected QoE depends on the wireless networks that is subject to a lot of constraints like bandwidth limitation and rate fluctuations due to the frequent changes of channel states and mobility.

QoE analysis over wireless networks has been studied for many years. In [2] authors studied the QoE in a shared fast-fading channel using an analytical framework based on Takacs Ballot theorem. They used a GI/D/1 queue to model the system, so they supposed that the arrival process is independent and identically distributed (i.i.d). In [14] the analysis of buffer starvation using M/M/1 queue is performed. They used a recursive method to compare it with the Ballot theorem method even if the recursive method did not offer explicit result. They supposed an i.i.d arrival process that is a Takacs Ballot theorem. They used a GI/D/1 queue to model the system, so they supposed that the arrival process is independent and identically distributed (i.i.d). In [14] the analysis of buffer starvation using M/M/1 queue is performed. They used a recursive method to compare it with the Ballot theorem method even if the recursive method did not offer explicit result. They supposed an i.i.d arrival process that is a rough model of streaming services over the wireless networks. As the performance measures depends on the autocorrelation structure of the traffic., a general consensus exists about the limitation of the Poisson process to model the traffic behaviour. In [16] authors developed an analytical framework to investigate the impacts of network dynamics on the user perceived video quality, they modeled the playback buffer by
a G/G/1 queue and used the diffusion approximation method to compute the QoE. The QoE of streaming from the perspective of flow dynamics is studied in [7]. Indeed a tagged user sees the others and his throughput is governed by the number of users that are present in the network. This study reveals that the flow dynamics is the fundamental reason of playback starvation.

To the best of our knowledge, this paper is the first attempt to address the QoE analysis in MMFM for finite media size. In particular we assume that the arrival of packets at player buffer characterized by an MMFM, which accurately approximates the traffic exhibiting LRD behaviour and mimics the real behaviour of multimedia traffic with short-term and long-term correlation [15]. In comparison to related works, our whole analysis is on transient regime. We construct sets of Partial Differential Equations (PDEs) to derive the starvation probability as function of the file size as well as the prefetching threshold. Moreover we provide relevant results to understand on how the starvation probabilities are impacted by the variation of traffic load and prefetching threshold. We finally do extensive simulations to show the accuracy of our model using ns3.

II. System Model Description

We consider a single user receiving a media file with finite size $Z$ in a streaming manner. Generally, media files are divided into blocks of frames, and when a user makes a request, the server segments this media into frames and transfers them to the user through wired or wireless links or both. When frames traverse the internet, their arrivals are not deterministic due to the dynamics of the available bandwidth. One of the main characteristic of wireless traffic and Internet traffic in general is the rate fluctuation that can be happened in many situations. Data packet arrivals in Ethernet, WLAN and cellular networks are found to be correlated over both short and long-time scales. These features generally result from the arrival of bursts of packets of comparable size, often leading to high instantaneous arrival rates. Hence video is streamed through the Internet with fluctuating speed. In this paper we assume that frames arrive to the play-out buffer with a rate that can take values from finite set $S = \{\lambda_i, i = 1, 2, ..., L\}$. The rate of arrival frames is governed by a Continuous-Time Markov Chain (CTMC). This approach predicts the starvation probability as function of the file size as well as the prefetching threshold. Moreover we provide relevant results to understand on how the starvation probabilities are impacted by the variation of traffic load and prefetching threshold. We finally do extensive simulations to show the accuracy of our model using ns3.

III. Analysis of the Queuing System Model

A. Laplace Transform of the Starvation Probability

We compute the Laplace transform of the probability of starvation given the Continuous-Time Markov Chain $\{I(t), t \geq 0\}$. We define $H_{ij}(x,t)$ to be the probability of starvation at state $j$ before time $t$, given the initial state $i$ and the initial queue length $x$.

$$H_{ij}(x,t) = P\{\tau \leq t, I(\tau) = j | X(0) = x, I(0) = i\} \quad (1)$$

for $i, j = 1, 2, ..., L$, $x > 0$ and $t \geq 0$. It is clear that the CTMC cannot be in a state $j$ at time $\tau$ if $r_j > 0$. Hence we have

$$H_{ij}(x,t) = 0 \quad \text{for all} \quad t \geq 0, \quad x > 0 \quad \text{if} \quad r_j > 0.$$

$$H_{ij}(x,t) = 0 \quad \text{for all} \quad t > 0, \quad x > 0 \quad \text{if} \quad r_j = 0.$$

Let $\pi = (\pi_1, \pi_2, ..., \pi_L)$ be the steady state probability vector of the CTMC $\{I(t), t \geq 0\}$ where $\pi_i$ is the probability to be in the state $i$ at the stationary regime. The expected input and output rates in steady state are $\sum_{i \in S} \pi_i \lambda_i$ and $\sum_{i \in S} \pi_i \mu_i$ respectively. The buffer is stable if $\sum_{i \in S} \pi_i \lambda_i < \sum_{i \in S} \pi_i \mu_i$. Conditioning on the first transition from the state at time 0 we get

$$H_{ij}(x,t) = \sum_{k \neq i} q_{ik} \Delta t H_{kj}(x + r_i \Delta t, t - \Delta t)$$

$$+ (q_{ii} \Delta t + 1) H_{ij}(x + r_i \Delta t, t - \Delta t) + o(\Delta t) \quad (2)$$

Taking the limit $\lim_{\Delta t \to 0} \frac{H_{ij}(x,t) - H_{ij}(x,t - \Delta t)}{\Delta t}$ and algebraic simplification yields the partial differential equation

$$\frac{\partial H_{ij}(x,t)}{\partial t} - r_i \frac{\partial H_{ij}(x,t)}{\partial x} = \sum_{k=1}^{L} q_{ik} H_{kj}(x,t) \quad (3)$$
This can be written in a matrix form as
\[
\frac{\partial H_j(x,t)}{\partial t} - R \frac{\partial H_j(x,t)}{\partial x} = QH_j(x,t)
\]
with the initial conditions
\[
H_{ij}(0,t) = \begin{cases} 
1 & \text{if } i = j \text{ and } t \geq 0 \\
0 & \text{if } i \neq j \text{ and } r_i < 0 
\end{cases}
\]
\[
H_{ij}(x,0) = 0 \text{ for all } i \neq j \text{ and } x \geq 0,
\]
where \( H_{ij}(x,t) = [H_{1j}(x,t), H_{2j}(x,t), ..., H_{Lj}(x,t)] \).

The Laplace Stieltjes Transform (LST) of \( H_{ij}(x,t) \) is
\[
\tilde{H}_{ij}(x,\omega) = \int_0^{\infty} e^{-\omega t} dH_{ij}(x,t) = E\{e^{-\omega \tau}; I(\tau) = j|X(0) = x, I(0) = i\}
\]
for \( i, j = 1, ..., L \) and \( \tilde{H}_{ij}(x,\omega) = [\tilde{H}_{ij}(x,\omega)] \). Taking the LST of Equation (4) and using the fact that \( H_{ij}(x,0) = 0 \) for all \( x > 0 \), we find
\[
R \frac{d\tilde{H}_{ij}(x,\omega)}{dx} = (\omega I - Q)\tilde{H}_{ij}(x,\omega)
\]
For a fixed value of \( \omega \), we try
\[
\tilde{H}_{ij}(x,\omega) = e^{s(\omega)x}\phi(\omega)
\]
as a solution to Equation (5). Substituting in (5) we get
\[
Rs(\omega)\phi(\omega) = (\omega I - Q)\phi(\omega)
\]
where the scalar \( s(\omega) \) and the vector \( \phi(\omega) \) are to be determined. The theorem 3.3 from [8] gives
\[
\tilde{H}_{ij}(x,\omega) = \sum_{s_k(\omega) \in E^-} a_{kj} e^{s_k(\omega)x} \phi_k^j(\omega)
\]
where the coefficients \( a_{kj} \) are obtained by solving
\[
\sum_{s_k(\omega) \in E^-} a_{kj} \phi_k^j(\omega) = \begin{cases} 
1 & \text{if } i = j \\
0 & \text{if } i \neq j
\end{cases}
\]
\( s_k(\omega) \) are the roots with negative real parts of \( \Delta(s,\omega) = det(Q + sR - \omega I) \) and \( \phi_k(\omega) \) are the corresponding eigenvectors satisfying the equation
\[
(Q + s(\omega)R - \omega I)\phi(\omega) = 0
\]
**B. Laplace Transform of the Start-up delay**

We consider the previous system during the prefetching process and we define \( X_s(t) \) to be the buffer content size at time \( t \). Let
\[
T_x = \inf\{t \geq 0 : X_s(t) \geq x\}
\]
be the first time that the buffer content reaches \( x \). \( T_x \) is the time that the system will take to accumulate \( x \) content in the buffer. This distribution is difficult to solve directly, so we resort to the same duality problem as in [4]:

**Duality problem:** What is the starvation probability by time \( t \) if the queue is depleted with rate \( \lambda_i \) and the duration of prefetching contents is \( x \)?

This duality problem allows us to compute the prefetching delay as a probability of starvation. We define \( U_{ij}(x,t) \) to be the probability of starvation before time \( t \) at the state \( j \), conditioning on the initial state \( i \) and the initial prefetching content \( x \), i.e., the start-up threshold.
\[
U_{ij}(x,t) = P\{T_x \leq t, I(T_x) = j|I(0) = i, X_s(0) = x\}
\]
for \( i, j = 1, 2, ..., L \), \( x > 0 \) and \( t \geq 0 \).

Conditioning on the first transition from the state at time 0 we get
\[
U_{ij}(x,t) = \sum_{k \neq i} q_{ik} \Delta t U_{kj}(x - \lambda_i \Delta t, t - \Delta t)
\]
\[+ (q_{ij} \Delta t + 1) U_{ij}(x - \lambda_j \Delta t, t - \Delta t) + o(\Delta t)
\]
Taking the limit \( \lim_{\Delta t \to 0} U_{ij}(x,t) - U_{ij}(x,t-\Delta t) \) and algebraic simplification yields the partial differential equation
\[
\frac{\partial U_{ij}(x,t)}{\partial t} + \lambda_i \frac{\partial U_{ij}(x,t)}{\partial x} = \sum_{k=1}^{L} q_{ik} U_{kj}(x,t)
\]
This can be written in a matrix form as
\[
\frac{\partial U_{ij}(x,t)}{\partial t} - R \frac{\partial U_{ij}(x,t)}{\partial x} = QU_{ij}(x,t)
\]
with the same initial conditions as in section III-A where \( R = diag\{-\lambda_1, -\lambda_2, ..., -\lambda_L\} \) and \( U_{ij}(x,t) = [U_{1j}(x,t), U_{2j}(x,t), ..., U_{Lj}(x,t)] \).
\[
\tilde{U}_{ij}(x,\omega) = \sum_{s_k(\omega)} a_{kj} e^{s_k(\omega)x} \phi_k^j(\omega)
\]
where the coefficients \( a_{kj} \) are obtained by solving
\[
\sum_{s_k(\omega)} a_{kj} \phi_k^j(\omega) = \begin{cases} 
1 & \text{if } i = j \\
0 & \text{if } i \neq j
\end{cases}
\]
\( s_k(\omega) \) are the roots of \( det(Q + sR - \omega I) \) and \( \phi_k(\omega) \) are the corresponding eigenvectors.

We also compute the probability of the prefetching end state process. For this purpose, we define
\[
V_{ij}(q,x) = P\{I(T_x) = j|I(0) = i, X_s(0) = q\}
\]
to be the probability that the prefetching stops at state \( j \) with \( x \) contents in the buffer given that the prefetching begins at state \( i \) with \( q \) contents in the buffer. In the time interval \([0,h]\) and conditioning on the first transition from the state at time 0 we get
\[
V_{ij}(q,x) = (1 + q_i h)V_{ij}(q + \lambda_i h, x)
\]
\[+ \sum_{k \neq i} q_{ik} h V_{kj}(q + \lambda_j h, x) + o(h)
\]
Algebraic simplification and letting \( h \to 0 \) yields the differential equation
\[
\lambda_i \frac{dV_{ij}(q,x)}{dq} = -\sum_{k=0}^{L} q_{ik} V_{kj}(q,x)
\]
with the boundary condition

\[ V_{ij}(x, x) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \]

Equation (13) can be written in matrix form as

\[ \text{diag}(\frac{1}{\lambda_i}) \tilde{V}(q, x) = -QV(q, x) \]

Let

\[ Q_v = \text{diag}(\frac{1}{\lambda_i}) (-Q) \]

Eq. (16) becomes

\[ \tilde{V}(q, x) = Q_v V(q, x) \]

Then \( V(q, x) \) is given by

\[ V(q, x) = \text{exp}(Q_v q) V(0, x) \]

Using Eq. (18) and the initial conditions

\[ V(q, x) = D_v \text{exp}(\Lambda_v (q - x)) D_v^{-1} V(x, x) \]

where \( D_v, \Lambda_v, D_v^{-1} = Q_v, \Lambda_v \) is the diagonal matrix containing all the eigenvalues of \( Q_v \) and \( D_v \) is an invertible matrix.

C. The Probability of Starvation and the Start-up Delay

In the previous sections we have derived explicit expressions for the Laplace-Stieltjes Transform of the probability of starvation and the start-up delay. In this section, we use a method to find the corresponding probability of starvation and start-up delay. The Laplace Stieljes Transform of \( H_{ij}(x, t) \) is

\[ \tilde{H}_{ij}(x, \omega) = E[e^{-\omega \tau}; I(\tau) = j | X(0) = x, I(0) = i] \]

\[ = \int_0^\infty e^{-\omega t} dH_{ij}(x, t) = \int_0^\infty e^{-\omega t} h_{ij}(x, t) \]

where \( h_{ij} \) is the probability density function of the distribution \( H_{ij} \).

Lemma 1 (Bromwich inversion integral). Given the Laplace transform \( \tilde{h} \), the function value \( h(t) \) can be recovered from the contour integral

\[ h(t) = \frac{1}{2\pi i} \int_{b-i\infty}^{b+i\infty} e^{\omega t} \tilde{h}(\omega) d\omega, \quad t > 0, \]

where \( b \) is a real number to the right of all singularities of \( \tilde{h} \), \( i^2 = -1 \), and the contour integral yields the value 0 for \( t < 0 \).

It’s shown in [12] that for real value functions, \( h \) has the following form

\[ h(t) = \frac{2e^{bt}}{\pi} \int_0^\infty Re(\tilde{h}(b + iu)) \cos(ut) du \]

According to the Bromwich inversion integral, \( h(t) \) can be calculated from the transform \( \tilde{h} \) by performing a numerical integration (quadrature). We use a specific algorithm based on the Bromwich inversion integral. It is based on a variant of the Fourier-series method - the trapezoidal rule - which proves to be remarkably effective. If we use a step size \( T \), then the trapezoidal rules gives

\[ h(t) \approx h_T(t) = \frac{T e^{bt}}{\pi} Re(\tilde{h}(b)) + \frac{2T e^{bt}}{\pi} \sum_{k=1}^\infty Re(\tilde{h}(b + ikh)) \cos(kht) \]

where \( Re(\tilde{h}(b)) = \tilde{h}(b) \) since \( b \) is real. Replacing \( \tilde{h}(\omega) \) by \( \frac{\tilde{h}(x, \omega)}{\omega} \) which is the Laplace transform of \( H_{ij}(x, t) \), we get the probability of starvation before time \( t \)

\[ P\{\tau \leq t\} = \frac{2T e^{bt}}{\pi} \left[ \tilde{H}_{ij}(x, b) \right] + \sum_{k=1}^\infty Re(\tilde{H}_{ij}(x, b + ikh)) \cos(kht) \]

The infinite series in (23) can simply be calculated by simple truncating because it converges, but more efficient algorithm can be obtained by applying a summation acceleration method. An acceleration technique that has proven to be effective in our context is Euler summation, after transforming the infinite sum into a nearly alternating series in which successive summands alternate in sign. We convert (23) into a nearly alternating series by letting \( T = \pi / lt \) and \( b = A / 2lt \)

\[ h_T(t) = h_{A,l}(t) = e^{A/2it} + 2e^{A/2lt} \sum_{k=1}^\infty \tilde{h}(A/2lt + ik\pi/lt) e^{ik\pi/lt} \]

\[ = \sum_{k=0}^\infty (-1)^k a_k(t) \]

where

\[ a_k(t) = \frac{e^{A/2lt}}{2lt} \left( \tilde{h}(A/2lt) 1_{\{k=0\}} + 2 \sum_{j=1}^t Re\left[ \tilde{h}(A/2lt + ij\pi/lt + ik\pi/lt) e^{ij\pi/lt} \right] \right) \]

Let \( s_n \) be the approximation \( h_{A,l}(t) \) with the infinite series truncated to \( n \) terms, i.e.,

\[ s_n = \sum_{k=0}^n (-1)^k a_k \]

where \( t \) is suppressed in the notation and \( a_k \equiv a_k(t) \). We apply Euler summation to \( m \) terms after an initial \( n \), so that the Euler sum approximation is

\[ E(m, n) \equiv E(m, n, t) = \sum_{k=0}^m C_m 2^{-m} s_{n+k} \]

Euler summation can be very simply described as the weighted average of the last \( m \) partial sums by a binomial probability with parameter \( m \) and \( p = 1/2 \). Hence, (24) is the binomial average of the terms \( s_{n}, s_{n+1}, ..., s_{n+m} \). The implementation of the algorithm takes into account the values of \( (l, m, n, A) \). As in [12], we use \( (1, M, M, 2ln(10)/M) \) where \( M = 64 \).
After simplification and letting $l$ be 1, we get the Euler approximation $E(t)$ of the inverse $h(t)$ which seems to be a good approximation

$$E(t) = \sum_{k=0}^{m} c_k m^{2-m} \sum_{q=0}^{n+k} \frac{e^{A/2}}{2t} \left[ \tilde{h}(\frac{A}{2t})^{1}(q=0) - 2 Re(\tilde{h}(\frac{A}{2t} + \frac{i\pi(1+q)}{t})) \right]$$

(25)

$$h_{ij}(t) = \sum_{k=0}^{m} C_k m^{2-m} \sum_{q=0}^{n+k} e^{A/2} \left[ \tilde{H}_{ij}(\frac{A}{2t})^{1}(q=0) - 2 Re(\tilde{H}_{ij}(\frac{A}{2t} + \frac{i\pi(1+q)}{t})) \right]$$

(26)

Eq. (26) looks complicated, but it consists only on $((m + 1)(m + 2n + 2))/2$ additions, that is a low computation level. To have the cdf $H$, we just replace $\tilde{H}_{ij}(t)$ by $\tilde{H}_{ij}(t)$. The same formula holds for the start-up delay distribution in replacing $\tilde{H}_{ij}$ by $\tilde{U}_{ij}$.

IV. PERFORMANCES ANALYSIS OF THE QUALITY OF EXPERIENCE

In this section we compute the QoE metrics: the starvation probability and the start-up delay.

A. The Probability of Starvation

We consider a single user receiving a media file with size $Z$. The necessary time to play the whole video if there is no starvation is $\frac{Z}{\mu}$. Hence, using the first passage time distribution $H_{ij}(x, t)$, the probability of starvation happened at state $j$ with the initial state $i$, is given by

$$P_{s} = \sum_{k=0}^{m} C_k m^{2-m} \sum_{q=0}^{n+k} e^{A/2} \left[ \tilde{H}_{ij}(\frac{A}{2Z})^{1}(q=0) - 2 Re(\tilde{H}_{ij}(\frac{A}{2Z} + \frac{\mu(1+q)}{2Z})) \right]$$

(27)

The starvation probability before time $t$ gives an idea of the severity of the starvation during all the video duration. Let $D_{ij}(x) := E[\tau, I(\tau) = j|\tau < \infty, I(0) = i, X(0) = x]$ be the mean continuous playback time if the initial state is $i$, the prefetching threshold is $x$ and the starvation happens in state $j$. $D_{ij}(x)$ is an important measure for the severity of starvations. A small $D_{ij}(x)$ means that the starvation events happen frequently. We find $D_{ij}(x)$ by taking derivatives of $H_{ij}(x, \omega)$ in $\omega = 0$.

$$D_{ij}(x) = -\frac{\partial H_{ij}(x, \omega)}{\partial \omega} |_{\omega=0} \quad i, j = 1, ..., L$$

When the user starts the video session, the initial state is unknown to the system. The video starts playing when the prefetching process is finished. Conditioning on the distribution of the entry states $\pi$, the distribution of the states that the playback process begins (or prefetching process ends) is computed by $\pi.V(0, x)$. Recalling that $V_{ij}(0, x)$ is the probability that the prefetching phase ends at state $j$ knowing that the video session starts at state $i$. Then the starvation probability with the prefetching threshold $x$ is obtained by

$$P_{s}(x) = \pi.V(0, x).H(x, \frac{Z}{\mu})$$

(28)

where $H$ is a column vector, $H = (H_1, H_2, ..., H_L)^T$ and $H_i = \sum_{j=1}^{L} H_{ij}$. The last formula is called the overall starvation probability. The probability of no starvation is computed as $1 - P_{s}(x)$.

B. The distribution of the Start-up delay

The start-up delay is proportional to the start-up threshold. But in the QoE literature, it is more practical to consider the delay rather than the threshold because the delay has a direct impact on the streaming user behaviour. Using the results of the sections III-C and III-B, we derive the cumulative distribution function of the start-up delay

$$U_{ij}(x, t) = \sum_{k=0}^{m} C_k m^{2-m} \sum_{q=0}^{n+k} e^{A/2} \left[ \tilde{U}_{ij}(\frac{\mu A}{2Z})^{1}(q=0) - 2 Re(\tilde{U}_{ij}(\frac{\mu A + 2\pi \mu(1+q)}{2Z})) \right]$$

(29)

where $x$ is the start-up threshold, $Z$ is the file size and $m, n, A$ are the Euler Summation Algorithm parameters.

C. The generating function of the starvation events

When a starvation event happens, the media player pauses until $x$ contents are re-buffered. We are interested in the probability distribution of the starvations, given the file size $Z$. We define a path as a complete sequence of frames arrivals and departures. We illustrate a typical path with $j$ starvations in Fig. 1. The path can be decomposed into three types of mutually exclusive events as follows:

1. Event $E(t_1)$: the buffer becoming empty for the first time in the entire path.
2. Event $S(t_l, t_{l+1})$: the empty buffer after the instant $t_{l+1}$ given that the previous empty buffer happens at $t_l$.
3. Event $U(t_j)$: the last empty buffer observed after the instant $t_j$.

Obviously, a path with $j$ starvations is composed of a succession of events

$$E(t_1), S_1(t_1, t_2), S_2(t_2, t_3), ..., S_{j-1}(t_{j-1}, t_j), U(t_j)$$

We let $P_{E(t_1)}$, $P_{S_1(t_1, t_{l+1})}$ and $P_{U(t_j)}$ be the probabilities of events $E(t_1)$, $S_1(t_1, t_{l+1})$ and $U(t_j)$ respectively. The probability distribution of event $E(t_1)$ is expressed as

$$P_{E(t_1)} = \begin{cases} 0, & \mu t_1 < x \text{ or } \mu t_1 \geq Z; \\ \pi.V(0, x).h(x, t_1), \text{ otherwise.} \end{cases}$$

(30)

where $V$ and $h$ are $LxL$ and $Lx1$ matrices respectively. The first starvation cannot happen at the departure of first $(x-1)$ contents because of the prefetching of $x$ contents. It cannot happen after all $Z$ contents have been served because this
To compute the probability of having more than one starvation, the starvation happens at time $t_1$ conditioned on the states that the playback process begins. The probability distribution of event $U_j(t_j)$ is given by

$$P_{U_j(t_j)} = \begin{cases} 0, & \text{if } \mu_{t_j} < jx \text{ or } \mu_{t_j} \geq Z; \\ 1, & \text{if } Z - x \leq \mu_{t_j} < Z; \\ V(0,x),(1 - H(x, Z - t_j)), & \text{otherwise.} \end{cases}$$

(31)

where $H$ is a column vector. $t_j$ is the time of the $j$-th starvation. The extreme case is that these $j$ starvations take place consecutively. Then $\mu_{t_j}$ should be greater than $jx$. Otherwise there cannot have $j$ starvations. If $\mu_{t_j}$ is no less than $Z - x$, the media player resumes until all the remaining content $Z - \mu_{t_j}$ is stored in the buffer. Then, starvation will not appear afterwards. In the remaining case, it is the probability of having no starvation after time $t_j$. We denote by $P_s(j)$ the probability of having $j$ starvations. The case with one starvation is given by

$$P_s(1) = \int_{t=0}^{Z} P_{E(t)} P_{U_1(t)} \, dt$$

(32)

To compute the probability of having more than one starvation, we need to find the probability of event $S_l(t_1, t_1+1)$. $\mu_{t_1}$ should not be less than $lx$ in order to have $l$ starvations. Given that the buffer is empty just after time $t_1$, the $(l + 1)^{th}$ starvation cannot happen at $\mu_{t_1+1} \in [\mu_{t_1} + 1, \mu_{t_1} + x - 1]$ because of the prefetching process. Since there are $j$ starvations in total, the $(l + 1)^{th}$ starvation must satisfy $\mu_{t_1+1} < Z - (j - l - 1)x$. We next compute the remaining case that the $l^{th}$ and the $(l + 1)^{th}$ starvations happen at time $t_l$ and $t_{l+1}$ respectively. We compute this probability using the first passage time density when the starvation happens at time $t_{l+1}$ and the initial time was $t_l$ with a prefetching process. $P_{S_l(t_l, t_{l+1})}$ is expressed as

$$V(0,x),h(x, t_{l+1} - t_l),$$

if $\mu_{t_l} \geq lx, \mu_{t_l} + x \leq \mu_{t_{l+1}} < Z - (j - l - 1)x; 0, \text{otherwise.}$

(33)

We use this method a trick that concerns the time scale. Every time the player resumes for the prefetching process we resume also the time scale, that means if the starvation happens at time $t$, the player will start playing at the same time $t$ with $x$ initial contents in the buffer. The probability of having $j(j \geq 2)$ starvations is given by

$$P_s(j) = \int_{t_1=0}^{Z} \int_{t_2=0}^{Z} \cdots \int_{t_{j-1}=0}^{Z} \int_{t_j=0}^{Z} P_{E(t_j)} P_{S_1(t_1, t_2)} \cdots P_{S_{j-1}(t_{j-1}, t_j)} P_{U_1(t_1)} dt_1 dt_2 \cdots dt_{j-1} \, dt_j$$

(34)

In the next section, we provide explicit expressions of QoE metrics where the CTMC has two states.

V. THE 2-STATE MMFF SOURCE

The environmental state process $\{I(t), t \geq 0\}$ is a CTMC on $\{1, 2\}$ (1 for $S_1$ and 2 for $S_2$) (Fig. 2) with infinitesimal generator $Q$ and rate matrix $R$

$$Q = \begin{pmatrix} -\beta & \beta \\ \alpha & -\alpha \end{pmatrix}$$

$$R = \begin{pmatrix} \lambda_2 - \mu & 0 \\ 0 & \lambda_1 - \mu \end{pmatrix}$$

Using the results of the section III-A irrespective of the condition of stability of the queue $\pi_1 \lambda_1 + \pi_2 \lambda_2 < \mu$, we get:

$$\Delta(s, \omega) = det(Q + sR - \omega I)$$

$$= (\lambda_1 - \mu)(\lambda_2 - \mu)s^2 - [(\lambda_1 - \mu)(\omega + \beta) + (\lambda_2 - \mu)(\omega + \alpha)]s + \omega(\omega + \alpha + \beta)$$

(35)

It is a polynomial of degree 2 in $s$ where the two zeros are given by:

$$s_1(\omega) = \frac{b + \sqrt{b^2 - 4(\omega(\omega + \alpha + \beta)(\lambda_1 - \mu)(\lambda_2 - \mu)}}{2(\lambda_1 - \mu)(\lambda_2 - \mu)}$$

(36)

$$s_2(\omega) = \frac{b - \sqrt{b^2 - 4(\omega(\omega + \alpha + \beta)(\lambda_1 - \mu)(\lambda_2 - \mu)}}{2(\lambda_1 - \mu)(\lambda_2 - \mu)}$$

(37)

where $b = (\lambda_1 - \mu)(\omega + \beta) + (\lambda_2 - \mu)(\omega + \alpha)$. Equation [6] contains terms with only $Re(s_k(\omega)) < 0$. So we have to determine the signs of $Re(s_1(\omega))$ and $Re(s_2(\omega))$. The next propositions give the placement of these two zeros in the complex plane.
Proposition 1. Let $\lambda_1 > \lambda_2$.
1. $\lambda_2 > \mu \Rightarrow \lambda_1 > \mu$, so $\lambda_1 - \mu > 0$ and $\lambda_2 - \mu > 0$ then no starvation.
2. $\lambda_2 < \mu$ and $\lambda_1 > \mu$, so $\lambda_2 - \mu < 0$ and $\lambda_1 - \mu > 0$ then $\text{Re}(s_2(\omega)) > 0$ and $\text{Re}(s_1(\omega)) < 0$.
3. $\lambda_2 < \mu$ and $\lambda_1 < \mu$, so $\lambda_2 - \mu < 0$ and $\lambda_1 - \mu < 0$ then $\text{Re}(s_2(\omega)) < 0$ and $\text{Re}(s_1(\omega)) < 0$.
4. $\lambda_2 = \mu \Rightarrow \lambda_1 > \mu$ then no starvation.
5. $\lambda_1 = \mu \Rightarrow \lambda_2 < \mu$, $s_2(\omega) = s_1(\omega) = s(\omega) = \frac{1}{(\lambda_2 - \mu)(\omega + \beta)}$ and $\text{Re}(s(\omega)) < 0$.

Proposition 2. Let $\lambda_2 > \lambda_1$.
1. $\lambda_1 > \mu \Rightarrow \lambda_2 > \mu$, so $\lambda_1 - \mu > 0$ and $\lambda_2 - \mu > 0$ then no starvation.
2. $\lambda_1 < \mu$ and $\lambda_2 > \mu$, so $\lambda_1 - \mu < 0$ and $\lambda_2 - \mu > 0$ then $\text{Re}(s_2(\omega)) > 0$ and $\text{Re}(s_1(\omega)) < 0$.
3. $\lambda_1 < \mu$ and $\lambda_2 < \mu$, so $\lambda_1 - \mu < 0$ and $\lambda_2 - \mu < 0$ then $\text{Re}(s_2(\omega)) < 0$ and $\text{Re}(s_1(\omega)) < 0$.
4. $\lambda_1 = \mu \Rightarrow \lambda_2 > \mu$ then no starvation.
5. $\lambda_2 = \mu \Rightarrow \lambda_1 < \mu$, $s_2(\omega) = s_1(\omega) = s(\omega) = \frac{1}{(\lambda_1 - \mu)(\omega + \beta)}$ and $\text{Re}(s(\omega)) < 0$.

Proposition 3. Let $\lambda_1 = \lambda_2 = \lambda$.
1. $\lambda > \mu$, no starvation.
2. $\lambda < \mu$, $\text{Re}(s_2(\omega)) < 0$ and $\text{Re}(s_1(\omega)) < 0$.
3. $\lambda = \mu$, no starvation since we start with $x$ contents in the buffer.

The LST $\tilde{H}_2(x, \omega)$ of the distribution is given in the next theorem.

Theorem 1. 1. When $\lambda_1 < \mu$, $\lambda_2 \geq \mu$, $\tilde{H}_2(x, \omega) = 0$ and
\[
\tilde{H}_1(x, \omega) = \frac{\tilde{H}_{11}(x, \omega)}{\tilde{H}_{21}(x, \omega)} = e^{s_1(x)x} \left[ \frac{1}{\beta + \omega - (\lambda_2 - \mu)s_1(\omega)} \right]
\]
2. When $\lambda_2 < \mu$, $\lambda_1 \geq \mu$, $\tilde{H}_1(x, \omega) = 0$ and
\[
\tilde{H}_2(x, \omega) = \frac{\tilde{H}_{22}(x, \omega)}{\tilde{H}_{12}(x, \omega)} = e^{s_2(x)x} \left[ \frac{1}{\beta + \omega - (\lambda_2 - \mu)s_2(\omega)} \right]
\]
3. When $\lambda_1 < \mu$, $\lambda_2 < \mu$,
\[
\tilde{H}_1(x, \omega) = a_{21}e^{s_2(x)x} \left[ \frac{1}{\beta + \omega - (\lambda_2 - \mu)s_2(\omega)} \right] + a_{11}e^{s_1(x)x} \left[ \frac{1}{\beta + \omega - (\lambda_2 - \mu)s_1(\omega)} \right]
\]
\[
\tilde{H}_2(x, \omega) = a_{22}e^{s_2(x)x} \left[ \frac{1}{\beta + \omega - (\lambda_2 - \mu)s_2(\omega)} \right] + a_{12}e^{s_1(x)x} \left[ \frac{1}{\beta + \omega - (\lambda_2 - \mu)s_1(\omega)} \right]
\]
where
\[
a_{11} = \frac{\beta + \omega - (\lambda_2 - \mu)s_1(\omega)}{(\lambda_2 - \mu)(s_2(\omega) - s_1(\omega))}, a_{21} = \frac{\beta + \omega - (\lambda_2 - \mu)s_2(\omega)}{(\lambda_2 - \mu)(s_1(\omega) - s_2(\omega))},
a_{22} = \frac{1}{(\lambda_2 - \mu)(s_1(\omega) - s_2(\omega))}, a_{12} = \frac{1}{(\lambda_2 - \mu)(s_2(\omega) - s_1(\omega))}
\]
The proof of this theorem can be find in the appendix. Taking $\lambda_2 = 0$ gives the first passage time distribution for the ON-OFF source.

VI. NUMERICAL ANALYSIS

A. Simulation

We use ns-3 simulator in order to compare the dynamics of the process with our model. We run 10 simulations of each scenario and we plot the average in the figures, the 95% of confidence interval is very small, we decide not to plot it for clarity. We first show the accuracy of the method that we use to invert the Laplace Transform. In Fig. 3, we plot the known inverse Laplace Transform of the function $f(t) = e^{-\beta t} \sin(\pi t)$ that is $f(s) = \pi/((s + \pi)^2 + \pi^2)$ and the inverse using formula (25) of section III-C. Fig. 4 and 5 show the starvation probability for an on-off source for two scenarios. In the first one, $\lambda = 30$ and $\mu = 25$, so the effective rate $\lambda - \mu$ is positive, that means the buffer size increases during packets arrival. In the second one, $\lambda = 9$ and $\mu = 10$. This is done for states transitions $\alpha = 2$ and $\beta = 6$. Fig. 6 illustrates the impact of the start-up threshold $x$ on the probability of no starvation. Fig. 7 and 8 show the probability of having one and two starvations respectively.
B. Performances Evaluation

Fig. 9 gives the CDF of the start-up delay for different values of start-up threshold $x$. We can see that the start-up delay increases with $x$. Indeed, streaming users are impatient about large start-up delay. They could leave the video session before the playback starts. That is the reason why we must set a small start-up delay to achieve a good QoE. In the other hand, too small start-up delay could increase the starvation probability, so a tradeoff exist between the start-up delay and the probability of starvation. The model gives this tradeoff because the probability of starvation depends on the start-up threshold $x$, so on the start-up delay. Figures 6 and 10 illustrate respectively the impact of the start-up threshold $x$.
and the file size $Z$ on the probability of no starvation. When $x$ is large enough (near 300 pkts in the figure) no starvation will happen until the end of the video. Because the curve grows sharply, it is clear that a slight increase in $x$ can greatly improve the starvation probability. On the other side, when the file size is large enough (near 400 pkts in the figure), starvation will happen for sure. This curve shows that starvation issue is more severe for long video duration. In figure 11 we plot the probability of having no more than two starvations with $\lambda = 30, \mu = 25$ irrespective to $x$. When $x$ is great enough, no starvation will happen until the end of the video session. On the other side, figure 12 shows that the starvation is for sure when the file size approaches infinity. The curves of the probability of having one or two starvations increase first, and then decrease to zero. This means that the starvation can be avoid when $x$ is great enough. These two curves have mainly a maximum value at a given start-up threshold or a given file size. The model provides us to choose the threshold $x$, to have (almost) exactly one or two starvations. This is an important measure because it allows the content provider to achieve its QoE requirements in setting up the right values.

C. Optimization of the QoE

The model can be used to optimize the QoE in the following way: Given the QoE requirements based on the number of starvation events that can annoy the users, the starvation probability which depends on the file size $Z$ is limited by a threshold that we call $\epsilon$. One can compute the optimum start-up threshold that satisfy the criteria $P_s(Z) < \epsilon$ by

$$x^* = \min\{x | P_s(Z) < \epsilon\}.$$  

Fig. 13 gives the optimum start-up threshold for different file sizes, for different QoE requirements. It first shows that the optimum value of $x$ increases with the file size, but we can also see that this value increases for systems which require small number of starvation. This is the reason why authors in [14] defined a cost to model the tradeoff between the probability of starvation and the start-up delay which lead to the same results. This figure also show that a network which states switch more is more efficient than another one which states does not change enough.

VII. Conclusion

In this paper, we propose a new analytical framework to compute the QoE of streaming in network modeled by the Markov Modulated Fluid Model. We find the first passage time distribution in fluid models, that gives the probability of starvation and the start-up delay which lead to the same results. These allow us to compute the number of starvation during the video session that is an important metric of the quality of experience. Then we do some simulations using ns3 to show the accuracy of our model. We propose our method to optimize the quality of experience using our model. This method is useful enough because it allows the content provider to set up the right QoE metrics given a set of video with different durations and the network parameters.

REFERENCES

[1] https://www.sandvine.com/trends/global-internet-phenomenal
[2] Y. Xu, E. Altman, R. El-Azouzi and S. Elayoubi, QoE Analysis of Media Streaming in Wireless Data Networks, Networking Springer Berlin Heidelberg, 2012.
yields the result of the theorem. The same proof holds in the

\[ a \]

without packets in the buffer in state 1, we’ll have starvation

\[ QoE-Driven \]

[5] S. Thakolsri, S. Khan, E. Steinbach, and W. Kellerer, \textit{QoE-Driven Cross-Layer Optimization for High Speed Downlink Packet Access}, Journal of Communications, vol. 4, no. 9, pp. 669-680, Oct. 2009.

[6] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica and H. Zhang, \textit{Developing a Predictive Model of Quality of Experience for Internet Video}, Sigcomm 2013, Hong Kong.

[7] Y.D. Xu, S.E. Elayoubi, E. Altman, R. Elazouzi and Y. Yu, \textit{Impact of Flow-level Dynamics on QoE of Video Streaming in Wireless Networks}, IEEE Infocom, 2013.

[8] A. Narayanan, and V.G. Kulkarni, \textit{First Passage Times in Fluid Models and an Application to Two Priority Fluid Systems}, Computer Performance and Dependability Symposium, Proceedings IEEE International, 1996.

[9] H. Kobayashi, \textit{Modeling and Analysis: An introduction to System Performance Evaluation Methodology}, Addison-Wesley Reading, NY 1978.

[10] P. Ameigeiras, J. I. Ramos-Munoz, I. Navarro-Ortiz, P. Mogensen, and J. M. Lopez-Soler, \textit{QoE oriented cross-layer design of a resource allocation algorithm in beyond 3G systems}, Computer Communications, vol. 33, no. 5, pp. 571-582, 2010.

[11] F. Dobrian, A. Awan, I. Stoica, V. Sekar, A. Ganjam, D. Joseph, J. Zhan, and H. Zhang, \textit{Understanding the Impact of Video Quality on User Engagement}, Proc. of ACM Sigcomm 2011, Vol.41, No.4, pp. 362-373.

[12] J. Abate, G.L. Choudhury and W. Whitt, \textit{Computational Probability}, W. Grassman (ed.) Kluwer, Boston 1999.

[13] D.J. Bertsimas, D. Nakazato, \textit{Transient and busy period analysis of the G/GI queue: The method of stages}, J.C. Baltzer A.G. Scientific Publishing Company, June 1990.

[14] Y.D. Xu, E. Altman, R. El-Azouzi, M. Haddad, S. Elayoubi and T. Jimenez, \textit{Analysis of Buffer Starvation with Application to Objective QoE Optimization of Streaming Services}, IEEE Infocom, 2012.

[15] S. Shah-Heydari, T. Le-Ngoc, \textit{MMPP models for multimedia traffic}, Telecommunication Systems 15, March 1999.

[16] T.H. Luan, L.X. Cai and X. Shen, \textit{Impact of Network Dynamics on User’s Video Quality: Analytical Framework and QoS Provision}, IEEE Transactions on Multimedia, Vol. 12, January 2010.

[17] \url{http://www.nsnam.org/}

\section*{APPENDIX}

\textbf{Proof.} When \( \lambda_1 \neq \mu, \lambda_2 \neq \mu, \)

\[ Q+s(\omega)R-\omega I = \begin{bmatrix} -\beta + (\lambda_2-\mu)s(\omega) - \omega \\ \alpha \end{bmatrix} + \begin{bmatrix} \beta \\ -\alpha + (\lambda_1-\mu)s(\omega) - \omega \end{bmatrix} \]

\((Q + s(\omega)R - \omega I)\phi(\omega) = 0 \) and \( \phi^k \) is the eigenvector correspondidng to \( s_k(\omega) \) according to section \textbf{III-A}, then

\[ \phi^k = \begin{bmatrix} 1 \\ \frac{\beta + \omega - (\lambda_2-\mu)s(\omega)}{\beta} \end{bmatrix}^T, k = 0, 1. \]

When \( \lambda_1 < \mu, \lambda_2 \geq \mu, \) we use only \( \phi^1(\omega) \) in computing the distribution, since \( \text{Re}(s_0(\omega)) > 0. \) Thus the distribution \( \tilde{H}_1(x,\omega) \) becomes

\[ \tilde{H}_1(x,\omega) = a_{11} e^{s_1(x)\omega} + a_{21} e^{s_1(x)\omega} \begin{bmatrix} 1 \\ \frac{\beta + \omega - (\lambda_2-\mu)s(\omega)}{\beta} \end{bmatrix} \]

\( a_{11} = \tilde{H}_1(0,\omega) \) and \( \tilde{H}_2(0,\omega) = 1 \) because if we start without packets in the buffer in state 1, we’ll have starvation with probability 1 within the same state. So \( a_{11} = 1, \) that yields the result of the theorem. The same proof holds in the case \( \lambda_2 < \mu, \lambda_1 \geq \mu \) by interchanging \( \lambda_1 \) and \( \lambda_2. \)