User Experience Awareness Network Optimization for Video Streaming Based Applications

Hengky Susanto, ByungGuk Kim, and Benyuan Liu
Department of computer science
University of Massachusetts at Lowell
{hsusanto, kim, bliu}@cs.uml.edu

Abstract – The growing popularity of video-based applications has put an enormous strain on the network and causes the network to be more prone to congestion. The study of congestion control and bandwidth allocation problems is often formulated into Network Utility Maximization (NUM) framework, and the existing solutions for NUM generally focus on single-layered applications. However, today’s quality of video is often divided into several layers, where each layer provides a different level of quality enhancement. In this paper, we study how multi-layered video-based applications impact network performance and pricing through NUM formulation, particularly traffic from video streaming. In our investigation, we design and implement a new multi-layered user utility model that leverages on studies of human visual perception. Then, using this new utility model to examine network activities, we demonstrate that solving NUM with multi-layered utility is intractable, and that rate allocation and network pricing may oscillate due to user behavior specific to multi-layered applications. To address this, we propose a new approach for admission control to ensure quality of service and experience. Our simulation results show that the proposed admission control mechanism can effectively eliminate the oscillation behavior and mitigate link bottleneck.

Term Index- Congestion control, real time, networks multimedia.

I. INTRODUCTION

The explosive growth and popularity of video streaming based applications has put immense pressure on network requirements and performance. The study of managing network congestion is commonly formulated into Network Utility Maximization (NUM) framework [1,2]. The objective is to allocate bandwidth that maximizes total user utility, subject to network capacity constraints. Existing approaches usually model user satisfaction according to the network traffic elasticity of single-layered applications, as discussed in these literatures [1,2,3,4,5,13]. However, such model is insufficient to capture the characteristics of applications with multiple layers of quality, such as video streaming. This is because today’s video streaming is divided into several layers, with lower layers containing low resolution information and higher layers containing the fine information. The premise is that each layer is delivered separately, such that the vital layers may thus be transferred with guaranteed quality of Service (QoS), while other layers that enhance the quality of the video could be sent as best effort. These characteristics allow users to be adaptive with their demand for quality of experience (QoE) and QoS, which is not reflected in single layered applications. Therefore, to capture the characteristic of multi-layered video based application, in our investigation we implement and design user utility function for multi-layered applications that incorporates studies from computer graphics [9,18,19]. Using this multi-layered user utility function, we study how multi-layered applications may dynamically impact the network traffic and pricing under the NUM framework, particularly for video streaming based applications.

To better address the particularity of user utility function for multimedia applications, we design our utility function to reflect the characteristics of multi-layered encoding schemes, which is often used in video based applications. That is, the level of user utility is not just measured by the ability to meet the minimum required QoS, but also by the varying degrees of qualities associated with each encoding layer. The proposed user utility is guided by studies of human visual perceptions in the fields of computer graphics to ensure the accuracy of modeling the user’s experience aspect. In essence, we derive three important insights: the unique adaptive nature of multimedia applications, users are willing to tolerate some level of disruption for the sake of better image quality, and human ability to detect improvement in image quality is not infinite, i.e. it reaches a certain point where human eyes are no longer able to detect further improvement in image quality. These insights contribute to our staircase-shaped user utility function that follows the law of diminishing returns. This utility function also illustrates that users may have different levels of QoE, which means users can be adaptive with their demand to achieve the desired QoE. Hence, we propose a model to encapsulate this user’s demand adaptability to achieve the desired QoE. Furthermore, the model also considers the impact of user’s willingness to pay (budget) for the desired QoE. We then use these models to investigate network activities by incorporating the multi-layered utility function into NUM framework. Our results show that the algorithm used to resolve NUM may not converge when users are actively seeking to meet their desired QoE, resulting in

The conference version of this paper [39] was accepted by the 24th IEEE International Conference on Computer Communications and Networks (IEEE ICCCN-2015)
oscillation in bandwidth allocation and network price. Furthermore, the oscillation also causes frequent quality adaptation at the video application level and creates a visual 

**flickering effect**, which degrades QoE and users find the effect annoying [27]. Moreover, our results also show that the oscillation ripples to different parts of the network, and cause bandwidth allocation to other users to oscillate too. This makes solving NUM problem with multi-layered user utility to become intractable, therefore, it may not have an optimal solution. To resolve this problem, we propose greedy based network admission control to ensure that acceptable QoE and QoS are achieved. We also provide mathematical proofs to show that the proposed admission control scheme achieves convergence and prevents oscillation. Moreover, we also show that our solution also has characteristics of Nash Equilibrium when users have minimum to gain by changing their own bandwidth demand. Following a discussion of our proposed solution, we will address how the proposed solution can be implemented in practice and how it will benefit network traffic engineering.

![Staircase user utility function for multi layered multimedia applications.](image)

This paper is organized as follows. We begin by discussing previous related work. Following this, we present our major contributions: the proposed multi-layered user utility model and a discussion on how the new model impacts network activities in section III and IV respectively. We introduce admission control in section V. The simulation results and discussion are presented and discussed in section VI and VII respectively, followed by concluding remarks.

### II. RELATED WORK

A number of proposals address NUM problems for multi-layered applications by incorporating rate video distortion minimization into user utility function. In [11], the authors’ objective is to allocate bandwidth that minimizes video rate distortion, where lower rate of distortion increases higher user satisfaction. Likewise, in [16], peak signal noise ratio (PSNR) is incorporated into user utility function in their NUM formulation. In [15], the authors develop network coding based utility maximization model by integrating a taxation-based incentive mechanism to address how many layers each user should receive and how to deliver them. Fundamentally, these proposals follow the convention of single layer utility function. That is, the layer used in the solution is predetermined and not adaptive, even though it is designed to support multi-layered applications.

Others propose to adopt a staircase shape or stepwise utility function. In [11], a staircase shape rate distortion- based utility function is incorporated into NUM frameworks in multicast network environment. Different weight factors are assigned to utility function, corresponding to the quality of the layer. The authors of [12] suggest an approach that connects different points in the staircase to produce a concave shape, which in turn provides some approximate rate change. However, it is still difficult to fine tune the variables to achieve a tight approximation. Similarly, in [28], staircase shape utility function is approximated using the log function. The idea is that every level in the staircase is represented by a concave shape utility function and the authors assume the optimal solution falls within one of the steps of the staircase. In both [12,28], the encoding layer is determined after bandwidth is allocated and user utility is measured according to the allocated bandwidth. In contrast, in our model, user utility is measured according to the quality of experience user has with the image quality, rather than plainly based on the amount of bandwidth allocated. In addition, our model considers a critical characteristic of multi-layered applications – the adaptive nature of such applications which adjusts its quality according to the traffic load in order to deliver the best QoE possible to users.

### III. MULTI-LAYERED USER UTILITY FUNCTION

#### A. Foundations

Multi-layered user utility function is modeled according to layered encoding scheme to reflect how multimedia traffic is managed at the network and at the application level. This means video information is divided into several encoding layers in order to minimize the amount of bandwidth used, with lower layers containing low resolution information and higher layers finer quality ones [10]. This strategy allows the video provider to provide a range in video qualities. This strategy is also known as hierarchical coding. Network forwards only the number of layers that the physical link can support and drops layers selectively at the bottleneck link. Thus, the hierarchical video encoding provides the foundation of staircase-like user utility function illustrated in figure 1.

Then we incorporate knowledge of how human eyes perceive and evaluate image qualities in our utility model because QoE is significantly affected by our visual ability. A study in [21] shows as video is encoded with more layers, discerning differences in quality between images at similar levels of quality becomes progressively difficult as the quality improves. This is because human visual capacity is less sensitive to high frequency image details and more sensitive to lower frequency [10,18,19]. Thus, we assert that human visual perception actually follows the law of diminishing returns, i.e. the benefits of offering higher quality diminishes as quality improves. Our user utility function incorporates these factors in the QoE, demonstrated by the progressively decreasing heights of the steps in the staircase-shaped function. In addition, supporting studies show that degradation in video is even less noticeable when there is a high degree of motion in the imagery [20], such as in action movies or sports videos, and especially when images are combined with good audio quality [21].

#### B. User Utility Function

By incorporating layered encoding scheme and limitations of human’s visual perception, our user utility function for multi-layered applications is modeled as follows. Since the staircase in the user utility model obeys the law of diminishing returns, the height of the steps progressively flattens toward the maximum quality, similar to an upward moving escalator. Let $Y$ be a set of
video encoding layers, $Y = \{1, 2, 3, ..., |Y|\}$, and $y \in Y$ when $y = 1$, $y$ is the lowest encoding layer; and when $y = |Y|$, $y$ is the highest encoding layer with the maximum quality. Let $x_s$ be the amount of bandwidth allocated to user $s$ and $x_s^y$ denotes the amount of bandwidth needed by user $s$ to support the application quality at layer $y$, such that $x_s \geq x_s^y$, which means network allocates at least $x_s^y$ to support the quality at layer $y$. Furthermore, let $U_{bw}(x_s^y)$ denote the user utility function of user $s$ for the amount of bandwidth $x_s$ at layer $y$, the relationship between user utility of a given layer can be illustrated as

$$\lim_{y \to \infty} U_{bw}(x_s^y) - U_{bw}(x_s^{y-1}) = 0.$$ 

Subsequently, by incorporating multi layers of encoding into the user utility function, function $U_{bw}(x_s)$ is defined as follows:

$$U_{bw}(x_s^y) = \sum_{i=0}^{|y|} \left( U_s^y(x_s^y) \psi_i(x_s^y) \right), \quad (1)$$

where $U_s^y(x_s^y)$ denotes user utility function given the required amount of bandwidth $x_s^y$ to support the quality at layer $y$ and $\psi_i(x_s^y)$ denotes the decay factor function at each layer and a common candidate function is $\psi_i(x_s^y) = e^{-\omega_i x_s^y}$, where $\omega_i$ is a positive variable used for normalization. Moreover, to ensure the minimum bandwidth required for the lowest encoding layer is met, condition $x_s \geq x_s^1$ must be met. Thus, $U_s^1(x_s) = \frac{1}{1-e^{-\omega_1}}$.

Next, we introduce $U_{cost}(x_s)$, the user utility function that measures user satisfaction over the cost of the service relative to user’s willingness to spend $m_s$. Thus, $U_{cost}(x_s) = 1 - \frac{\text{cost}(x_s)}{m_s}$. Therefore, user utility function for multi-layered applications has the following properties:

1. $U_{bw}(x_s^y), U_{cost}(x_s) \geq 0, \forall x_s, 0 \leq x_s^y \leq x_s \leq c_l$, and $U_{bw}(0) = 0, U_{cost}(0) = 1, \forall x_s, s$, where $c_l$ is the capacity of link $l$.

2. $U_{bw}(x_s^y)$ is twice differentiable and follows the law of diminishing returns.

3. $U_{bw}(x_s^y)$ is staircase-shape like $\forall x_s, 0 \leq x_s^y \leq x_s, \quad y = \text{max}(Y \mid x_s^y \leq x_s)$, and $U_{bw}(x_s^y) \leq U_{bw}(x_s)$.

4. $U_{cost}(x_s)$ is linear.

5. $\frac{\partial U_{bw}(x_s^y)}{\partial x_s^y} < \infty$, for all $0 \leq x_s^y \leq x_s \leq c$.

6. $\lim_{x_s^y \to 0} \frac{\partial U_{bw}(x_s^y)}{\partial x_s^y} < \infty, \forall s$, $s \in S$ and for $x_s^y \leq x_s$.

7. For $\forall y, s, y < y + 1, x_s^y < x_s^{y+1}$, for $s \in S$ and $y \in Y$.

Then, the user utility function for multi-layered application is defined as follows:

$$U_s(x_s, x_s^y) = U_{bw}(x_s) + U_{cost}(x_s), \quad (2)$$

such that each user maximizes his/her own utility function by solving

$$\text{maximize } U_s(x_s, x_s^y) \quad (3.a)$$
$$\text{over } x_s \geq x_s^y \geq 0. \quad (3.b)$$

How user solves the problem (3) above will be discussed in the following section.

C. System Setup

Consider a network with a set of links $L$, and a set of link capacities $C$ over the links. Given a utility function $U_s(x_s, x_s^y)$ of user $s$ with the allocated bandwidth of $x_s$, the NUM formulation becomes

$$\max \sum_{s \in S} U_s(x_s, x_s^y), \quad (4.a)$$
$$\text{s.t. } \sum_{s \in S} x_s \leq C_l, \forall l \in L \quad (4.b)$$

layer $y < layer y + 1, \forall y \in Y, \quad (4.c)$
$$\text{over } 0 \leq x_s^{min} \leq x_s, \forall s \in S \quad (4.d)$$

where $S$ denotes a set of users, $x_s^{min}$ denotes the minimum bandwidth requirement of user $s$, and $s \in l$ means user who is transmitting data through link $l$. The constraint (4.c) is necessary to ensure that the higher layer depends on lower layer. In other words, a higher layer is only considered when lower layers are delivered.

Typically, the common solution to NUM is subgradient based method [3], and the dual problem $D$ to the primal problem of (4) is constructed as follows. $L(x, \lambda) = \sum_{s \in S} U_s(x_s, x_s^y) - \sum_{s \in S} \lambda_s x_s + \sum_{l \in L} \lambda_l C_l$, where $L(x, \lambda)$ is the Lagrangian form and $\lambda$ is known as a set of Lagrangian multipliers $\lambda_l$, which is often interpreted as the link cost and $\lambda_s = \sum_{l \in S} \lambda_l$. The dual problem $D$ is then defined as

$$\min D(\lambda), \quad \text{s.t. } \lambda \geq 0,$$

where the dual function $D(\lambda) = \max_{0 \leq x_s \leq x_s^{max}} L(x_s, \lambda)$. User decides the transmission rate $x_s(\lambda_s)$ at price $\lambda_s$ by solving

$$x_s(\lambda_s) = \arg \max_{0 \leq x_s \leq x_s^{max}} \left( U_s(x_s, x_s^y) \right), \quad (5)$$

where $x_s(\lambda_s)$ denotes bandwidth allocation at price $\lambda_s$. A subgradient projection method is used in [3]. Thus, the network on each link $l$ updates $\lambda_l$ on that link:

$$\lambda_l^{t+1} = \left[ \lambda_l^t - \sigma_l^t \left( C_l - \sum_{s \in S} x_s \right) \right]^+, \quad (6)$$

where $C_l - \sum_{s \in S} x_s$ is a subgradient of problem $D$ for $\lambda_l^t \geq \lambda_{min} \geq 0$. Here, $\lambda_{min}$ denotes the minimum price determined by the network. The time $t$ denotes the iteration index, for $0 \leq t$, and $\sigma_l^t$ denotes the step size to control the tradeoff between a convergence guarantee and the convergence speed, such that $\sigma_l^0 \to 0$ as $t \to \infty$ and $\sum_{t=1}^{\infty} \sigma_l^t = \infty$. Price paid by users is $\lambda_s = \sum_{l \in S} \lambda_l$, where $l \in S$ means the link that is used by $s$ to transmit data.

IV. ADAPTIVE USER DEMAND

The authors of [26] claim that, regardless of the algorithm, deciding the proper quality in dynamic network is essentially difficult because of a lack of information and transparency between OSI layers. Often, user experience at the application layer relies on users to decide and adapt to reach their desired quality. For that reason, we investigate how multi-layered user utility function mirrors user QoE and decision for bandwidth demand, and how these two factors impact network activities.
A. Adaptive Demand

The staircase like function allows user to be adaptive with their requirement to achieve the desired QoE, as long as the minimum demand is met. However, network does not distinguish between user utility of single or multi-layered applications. The implication is, the bandwidth allocation which is sufficient for users of single-layer applications may not be sufficient for users of multi-layered applications. This is because users using multi-layered applications may continue to demand more bandwidths to improve their experience. This is noted in HCI studies in [27] where viewers prefer jerking (or less smooth) video with better image quality over smooth but poor visual quality where objects in the video are not recognizable. Low tolerance for poor image quality may motivate users to demand for more bandwidths for better image quality. In other words, due to the dynamic nature of user demand, providing the minimum bandwidth requirement does not automatically maximize the aggregated user satisfaction level because there are varying levels of user satisfaction in a multi-layered environment. Conversely, a user may stop demanding for additional bandwidth as they consider cost of service, especially when a user is already experiencing a sufficiently high quality of service, when further improvement in image quality cannot be appreciated because our eyes are not able to detect the quality difference.

B. User’s Desire for Better Quality

In order to investigate the impact of the adaptive nature of multi-layered utility on network activities, we first provide a model that encapsulates user’s motivation to stay put with or scale up from the current quality. The rationale behind user’s desire for better quality function $B(x_s, x^k_s)$ is described as follows. Intuitively, users are generally assumed to desire the best possible “value” for the money they pay. By this, it means a user may prefer to lower his/her requirement to achieve better perceived value. On the other hand, a user may demand more bandwidth for higher quality when

$$\beta_s \frac{x_s^{y+k}}{m_s} \leq B(x_s, x^k_s), \quad (7)$$

where $\beta_s$ is a positive constant variable that indicates user’s desire to save or spend money. Lower $\beta_s$ means higher willingness to spend more money for additional bandwidth to achieve better quality at $y + k$. Otherwise, the user may stay put with the current quality at layer $y$. Additionally, positive variable $k$ denotes the number of layers to be increased, $m_s$ denotes user’s budget or willingness to pay for the service, and $\lambda_s$ is the network price. $(x_s^{y+k}, \lambda_s)$ can be interpreted as the cost a user incurs for quality at layer $y + k$. Hence, the ratio $\frac{x_s^{y+k}}{m_s} \lambda_s$ shows that user’s desire to for higher layer for better quality can be expressed through increasing $m_s$. Next, $B(x_s)$ is defined as follows.

$$B(x_s, x^k_s) = \frac{u_{j_y}^y(x_s^{y+k}) - u_{j_y}^y(x^k_s)}{x_s^{y+k} \frac{\lambda_s + \lambda_s^{\text{Inc}}}{\beta_s}}, \quad (8)$$

where $\lambda_s^{\text{Inc}}$ denotes the range of price a user is willing to increase for the desired quality. Eq. (8) can be interpreted as user satisfaction gained over the cost users incurs from obtaining from additional $k$ layers. By re-arranging eq. (7), the lower bound for the additional price range $\lambda_s^{\text{Inc}}$ is

$$\lambda_s^{\text{Inc}} \geq \left( m_s \left( u_{j_y}^{y+k}(x_s^{y+k}) - u_{j_y}^y(x^k_s) \right) \right) - \lambda_s. \quad (9)$$

Observe in (9), the relationship between $\beta_s$ and $\lambda_s^{\text{Inc}}$ is that lower $\beta_s$ (higher desire to spend) implies a greater range in the price a user is willing to pay. In other words, eq. (9) provides the lower bound for the additional cost a user must spend to achieve quality at layer $y + k$ that at that given moment. Subsequently, given $x_s^{y+k}$, user may demand additional bandwidth to achieve higher quality if the new demand bandwidth $x_s^{y+k}$ is.

$$x_s = \begin{cases} x_s^{y+k}, & \text{If condition (7) is satisfied} \\ x_s, & \text{Otherwise} \end{cases} \quad (10)$$

That implies user may attempt to transmit data at $x_s^{y+k}$ instead of $x_s$ for $x_s < x_s^{y+k}$, when the condition allows.

C. The Impact of Adaptive User Demand

According to the condition in (10), when (8) is not satisfied, then user must transmit data at $x_s$ and stop demanding additional bandwidth $x_s^{y+k}$.

**Proposition 1:** When $x_s \geq x_s^y$, then $\frac{u_{j_y}(x_s^y)}{u_{j_y}(x_s^y)} = 1$, as $\lambda_s \to \infty$.

**Proof:** First, observe that

$$B(x_s, x^k_s) = \lim_{\lambda_s \to \infty} \frac{u_{j_y}^y(x_s^{y+k}) - u_{j_y}^y(x^k_s)}{x_s^{y+k} \frac{\lambda_s + \lambda_s^{\text{Inc}}}{\beta_s}}$$

and $\lim_{\lambda_s \to \infty} \beta_s \frac{x_s^{y+k}}{m_s} \lambda_s = \infty$ for $\beta_s > 0$. Since

$$\limsup_{\lambda_s \to \infty} \beta_s \frac{x_s^{y+k}}{m_s} \lambda_s = \infty,$$

notice that $B(x_s) < \infty$ for any $\lambda_s$. Hence, as $\lambda_s \to \infty$, $\beta_s \frac{x_s^{y+k}}{m_s} \lambda_s > B(x_s)$. Consequently, when the condition in (8) is no longer satisfied, user $s$ stops demanding additional bandwidth beyond $x_s$. As a result, user $s$ settles with layer $y = \max(y \mid x_s^y \leq x_s)$, such that $\limsup x_s^y = x_s$. Hence, $\frac{u_{j_y}(x_s^y)}{u_{j_y}^y(x_s^y)} = 1$, as $\lambda_s \to \infty$.

Proposition 1 shows that users eventually stop demanding more bandwidth when additional quality is not worth the additional cost. Next, we investigate whether the algorithm with multi-layered utility function also converges. First, we prove the following statement.

**Lemma 1:** Suppose that $\lambda_i^*$ is an optimal solution for the dual problem $D(\lambda)$, where $\lambda_i^* > 0$ for link $\forall l, l \in L$, such that there exists a subgradient of $D(\lambda)$, $g(\lambda_i^*)$, at $\lambda_i^*$, where $g(\lambda_i^*) = 0$.

**Proof:** we have $\lambda_i^*$ as the minimizer of $D(\lambda)$, as $\forall l, l \in L$, there exists $g(\lambda_i^*)$ that satisfies

$$g(\lambda'^T) \mid \lambda - \lambda'^* \mid \geq 0, \quad \forall \lambda, \lambda' \geq 0. \quad (11)$$

If we take $\lambda = \lambda'$, where $\mid \lambda_i - \lambda_i^* \mid = \epsilon$, $\epsilon > 0$. By (11), we have $g_1(\lambda_i^*) \epsilon_i \geq 0$. Hence, when $\lambda_i^*$ is the optimal solution, $g_1(\lambda_i^*) = 0$. \[\blacksquare\]
We have shown that there exists subgradient \( g_l(\lambda^*_t) = 0, \forall t, l \in L \), when \( \lambda^* \) is an optimal solution for the dual problem \( D(\lambda) \).

Next, we investigate whether the algorithm converges with multi-layer user utility function.

**Proposition 2:** Suppose that \( \lambda^*_t \) is an optimal solution for the dual problem \( D(\lambda) \), where \( \lambda^*_t > 0 \) for link \( \forall t, l \in L \). If \( D(\lambda) \) is differentiable at \( \lambda^* \), then \( x(\lambda^*_t) \) converges \( x(\lambda^*) \) as \( \lambda^*_t \) converges to \( \lambda^* \), for \( t \to \infty \). Otherwise, \( x(\lambda^*_t) \) and \( \lambda^*_t \) may not converge.

**Proof:** Certainly, when dual problem \( D(\lambda) \) is differentiable at \( \lambda^* \), then \( D(\lambda) \) has a unique subgradient at \( \lambda^* \). Thus, \( x(\lambda^*_t) \) is also unique. This means \( x(\lambda^*_t) \) continues at \( \lambda^* \), which implies that \( x(\lambda^*_t) \) converges to \( x(\lambda^*) \) and \( \lambda^*_t \) converges to \( \lambda^* \), for \( t \to \infty \). By lemma 1, this includes when subgradient \( g_l(\lambda^*_t) = 0 \).

However, when \( D(\lambda) \) is not differentiable at \( \lambda^* \), then the subgradient at \( \lambda^* \) is not unique. Thus, there exists a user with \( x^y_{s}^{\gamma_{l}} \), such that \( x_s < x^y_{s}^{\gamma_{l}} \). Given price at \( \lambda^*_t \), by the condition in (10), \( x_s(\lambda^*_t) \) is discontinuous when condition (8) is satisfied. This implies the subgradient of \( D(\lambda^*_t) \) at \( \lambda^* \) is not unique. Thus, by eq. (10), \( x_s(\lambda^*_t) \) may not converge. Furthermore, since \( \lambda^*_t \) is a reflection of \( \sum_{s \in S(l)} x_s(\lambda^*_t) \) in (6), \( \lambda_t \) may not converge either. ■

**Lemma 2:** When \( D(\lambda) \) is not differentiable at \( \lambda^* \), there exists a link \( l \) that satisfies this following condition:

\[
\sum_{s \in S(l)} x_s(\lambda^*_t) < C_l \quad \text{and eq. (7) is satisfied.} \quad (12a)
\]

\[
\sum_{s \in S(l)} x_s(\lambda^*_t) \geq C_l \quad \text{and eq. (7) is not satisfied.} \quad (12b)
\]

**Proof:** When \( D(\lambda) \) is not differentiable at \( \lambda^*_t \), proposition 2 shows that there exists user \( s \) with \( \lambda^*_s \) that satisfies condition (8). Thus, according to the condition in (10), \( x_s(\lambda^*_t) \) discontinues at \( \lambda^*_t \), then network ends up in condition (12.a). Furthermore, user \( s \) may transmit data at rate \( x^y_{s}^{\gamma_{l}} \) at time \( t \), for \( x_s < x^y_{s}^{\gamma_{l}} \). Since \( \sum_{s \in S(l)} x_s(\lambda^*_t) \geq C_l \), user \( s \) must transmit at \( x_s(\lambda^*_t) \). Thus, \( x_s(\lambda^*_t) \) also discontinues at \( \lambda^*_t \), then network ends up in (12.b). ■

Proposition 2 and lemma 2 imply that the algorithm may not converge and the rate allocation may oscillate between the two cases in (10) as a result from users attempting to obtain additional bandwidth for better QoS. The oscillation is also an indication there is a gap between the primal problem (4) and its dual problem \( D \). The gap is driven by users’ responses to the new prices advertised. That is, in one situation, users may feel the price is too high and decide not to demand additional bandwidth. In a different situation, the same users may demand additional bandwidth to achieve higher QoS when the price is acceptable to them. This makes solving the optimization problem with multi-layered user utility becomes intractable. Thus, there may be no optimal solution for the primal problem.

For this reason, we further investigate how this phenomenon may affect the network. Here, we divide users of multi-layered application into two categories: Passive users and Active users.

**Definition 1.** Passive Users: Users who accept the amount of bandwidth \( x_s \) allocated by the network and adjust the quality according to \( x_s \), and achieve \( x^*_s \) by solving \( y = \max \{Y | x^*_s \leq x_s \} \).

**Definition 2.** Active Users: Users who continue to try demanding additional bandwidth above the amount of bandwidth allocated to them.

These active users may cause oscillation as they change their transmission rate, which in turn affects the network pricing. They will stop demanding more bandwidth when they feel the additional quality is not worth the additional cost, or when the maximum quality is obtained. The question is therefore, how the behavior of active users affects network activities.

**D. The Ripple Effects of Active Users on Network**

The following discussion addresses how the behavior of active users may affect the bandwidth allocation to passive users.

**Lemma 3:** Suppose \( \sum_{s \in S(l)} x_s(\lambda^*_t) \) on link \( l \) oscillates, for \( t \to \infty \), then \( \lambda^*_t \) also oscillates.

**Proof:** Assume that \( x_s(\lambda^*_t) \) oscillates as \( t \to \infty \) and let \( z^*_l = \sum_{s \in S(l)} x_s(\lambda^*_t) \) on link \( l \). Since \( \lambda^*_t \) is updated by eq. (6), for \( t \geq t \), \( z^*_l \) increases and \( \lambda^*_t \) increases; and when \( z^*_l \) decreases, then \( \lambda^*_t \) also decreases. ■

Obviously, since eq. (6) is designed to respond to the traffic load in the network, the network price \( \lambda^*_t \) oscillates when the traffic load oscillates. In fact, eq. (6) is a feedback loop and \( \lambda^*_t \) continues to evolve as long as \( \lambda^*_t \geq 0 \). Hence, lemma 3 shows that active users can cause oscillation in pricing. Subsequently, we explore the effect of this pricing oscillation on bandwidth allocation for passive users.

**Proposition 3:** Bandwidth allocation for passive users is affected by the changes in network price caused by active users.

**Proof:** Let set \( S'(l) \) denote a set of active users and \( S(l) \) be a set of passive users sharing link \( l \), where \( S(l) = S'(l) \cup S \), \( \forall x \in S(l) \), and \( S \in S'(l) \), for \( l \in L \). Assume at time \( t - 1 \), \( x_s(\lambda^*_t) \) and \( x_s(\lambda^*_t) \) have converged at \( \lambda^*_t \), for \( \forall s \in S \). Suppose at \( t \), users in \( S'(l) \) demand more bandwidth and transmit data at \( x^y_{s}^{\gamma_{l}} \), where \( x^y_{s}^{\gamma_{l}} > x_s(\lambda^*_t) \), for \( \forall s \in S'(l) \), \( \forall y \geq 0 \). In the new price, \( \lambda^*_t \) is updated by solving (6). Then, user \( s \in S(l) \) computes \( x_s(\lambda^*_t) \) by solving (5). By rearranging (2), we have

\[
\lambda^*_t = \frac{m_s(U_{BW}(x_s) - U_{s}(x_s, x^*_s) - 1)}{x_s(\lambda^*_t)}
\]

\( \lambda^*_t \) is the network price that must be paid by user \( s \) from \( U_{cost}({\lambda}_s)(x_s) \) in (2). Hence, the relationship between \( \lambda^*_s \) and \( x_s(\lambda^*_t) \) can be illustrated as follows.

\[
\lim_{x_s \to \infty} \lambda^*_s(x_s) = \lim_{x_s \to \infty} \left( \frac{m_s(U_{BW}(x_s) - U_{s}(x_s, x^*_s) - 1)}{x_s} \right) = 0.
\]

However, we have
\[
\lim_{x_i \to -\infty} \lambda_i(x_i) = \lim_{x_i \to -\infty} \frac{m_i(U_{\text{net}}(x_i) - U_2(x_i, \lambda'_2) - 1)}{x_i} = \infty.
\]
Thus, when \( \lambda_{S(t)}^{i+1} < \lambda_{S(t)}^i \), then \( x_S(\lambda_{S(t)}^{i+1}) \geq x_S(\lambda_{S(t)}^i) \). However, when \( \lambda_{S(t)}^{i+1} \geq \lambda_{S(t)}^i \), then \( x_S(\lambda_{S(t)}^{i+1}) < x_S(\lambda_{S(t)}^i) \).

Proposition 3 implies that during excessive network congestion, oscillatory behavior exhibited by active users also impacts bandwidth allocation for passive users, which is consistent with the assumption that a heavier congestion leads to higher network price. As a result, passive users may end up with less bandwidth at a higher price. The worst case scenario is when the oscillation of network price \( \lambda_i \) causes the bandwidth allocation for passive user \( s \) to oscillate between two cases: \( x_S(\lambda_i^l) \leq x_{S}^{\min} \) and \( x_S(\lambda_i^l) > x_{S}^{\min} \), where \( x_{S}^{\min} \) is the minimum required bandwidth for minimum QoS. For these reasons, the worst case scenario also applies to single layer user utility function.

**Corollary 1.** Bandwidth allocation for users with single-layer utility function is also affected by the changes in network price caused by users with multi-layers utility function.

Therefore, the actions of active users may have a negative impact on the bandwidth allocation for passive users, such that passive users may not receive sufficient bandwidth even to meet the minimum requirement. Additionally, the oscillation also causes the quality of video to oscillate creating the visual flickering effect at the user level, that most people find annoying [27] and degrade user QoE. Therefore, we propose an admission control to assure QoE of users with multi-layered applications.

V. ADMISSION CONTROL

A. Admission Control Designed

In order to design an effective admission control (adm ctrl), network must decide the selection criteria to accept or reject users’ requests for admission. In multi-layered user utility environment, user demand is adaptive and the long term consequence of poor QoS is potential loss of future revenue. We assume that admission control is invoked at the occurrence of excessive network congestion and each candidate for admission is evaluated with function \( \theta_s(x_s(\lambda_s)) \) defined as follows.

\[
\theta_s(x_s(\lambda_s)) = x_s^y \left( \frac{\delta_x}{\lambda_s} + \frac{U_2'(x_s^y)}{\lambda_s x_s^y} \right),
\]

where \( \delta_x \) and \( \delta_u \) are non-negative parameters that function as a weight: the increase in \( \delta_x \) implies that network puts more emphasis in revenue. Similarly, network places more weight in user utility when \( \delta_u \) is increased.

Let \( \lambda_s \) be the network price decided by the network such that \( \lambda_{S}^{\min} \leq \lambda_s \) and \( \lambda_s \) denote the price user is willing to pay for the desired service quality. Here, a static minimum price \( \lambda_{S}^{\min} \) is determined by the network (for example, ISP). That is \( \lambda_s = \lambda_s^{inc} + \lambda_s \), for \( \lambda_s^{inc} \geq 0 \). Observe that \( \lim_{\delta_x \to \infty} \left( \frac{\lambda_s}{\lambda_s} \right) = 0 \) when \( \frac{\lambda_s}{\lambda_s} \to 1 \).

However, \( \lim_{\delta_x \to \infty} \left( \frac{\lambda_s}{\lambda_s} \right) = \lambda_s \), for \( \lambda_s > \lambda_s \) as \( \delta_x \to \infty \). This means users with \( \lambda_s > \lambda_s \) receive higher preference for admission when network places more emphasis in revenue. Additionally, since user utility function with multi layers of quality follows the law of diminishing returns, network may consider \( \frac{U_2'(x_s^y)}{\lambda_s x_s^y} \) from eq. (13), which can be interpreted as user satisfaction over the cost for desired quality at layer \( y \). Now we can formulate the admission control problem as the following optimization problem:

\[
\max \sum_{s \in S} \theta_s(x_s^y) x_s^y,
\]

subject to

\[
\begin{align*}
&s.t. \sum_{s \in S} x_s^y \leq C_l, \quad \text{for } \forall l, l \in L, \\
&x_s \in [0,1], \quad \text{for } \forall s, s \in S, \\
&\quad \text{over } x_s^y \geq 0, \quad \text{for } \forall s, s \in S \text{ and } y \in Y,
\end{align*}
\]

where \( z_s = 1 \) if user \( s \) is selected, otherwise zero. The difficulty of solving this problem lies in the search for all possible combinations of \( \theta_s(x_s^y) \), for \( s \in S \). For this reason, problem (14) is reduced to the 0-1 Knapsack problem [8], where each user must either be admitted or rejected. The network cannot admit a fraction of the amount of user’s traffic flow or admit users above the available capacity. To ensure real-time performance and quick completion of the admission process, we propose a three-stage heuristic greedy based algorithm to solve (14):

**Step one:** Network determines the price \( \lambda_i \) of each link \( l \). If \( \lambda_i < \lambda_{S}^{\min} \), then network sets the price \( \lambda_i = \lambda_{S}^{\min} \). This assures \( \lambda_{S}^{\min} \leq \lambda_i \). Next, network sends \( \lambda_s \) to user \( s \), where \( \lambda_s = \sum_{l \in S} \lambda_i \).

**Step two:** users submit a tuple of \( (x_s^y, \lambda_s) \), where \( \lambda_s \) is the price user is willing to pay. Users respond to network after evaluating

\[
x_s^y = \arg \max_{0 \leq x_s^y \leq \lambda_s} \left( \frac{U_2'(x_s^y)}{x_s^y \lambda_s} \right).
\]

**Step three:** Once network receive the necessary information, tuple \( (x_s^y, \lambda_s) \), from the entire users, network computes \( \theta_s(x_s(\lambda_s)), \forall s \in S \), and invoke “User Selection” algorithm.

**Algorithm 1:** User Selection.

1. \( \theta_{\text{max}} = \max \{ \theta_s(x_s(\lambda_s)) \} \)
2. \( x_s^y = \text{get_bandwidth} ( \theta_{\text{max}} ) \)
3. If \( (x_s^y + \sum_{s \in S} x_s^y) \leq C_l, \forall l, l \in \text{path } r \) and \( (s \in l) \) then
4. Reserve \( l \) for user \( s \), for \( \forall l, l \in r \)
5. \( C_l = C_l - x_s^y \) for \( \forall l, l \in r \)
6. \( \hat{S} = \hat{S} + s \)
7. \( \theta_{\text{SET}} = \theta_{\text{SET}} - \theta_{\text{max}} \)
8. Repeat from line 1 until \( \theta_{\text{SET}} = 0 \) until \( \theta_{\text{SET}} \) is empty

Let \( \theta_{\text{SET}} \) denote a set of \( \theta_s \) that is associated with user and \( \hat{S} \) denote a set of users admitted into the network. In line 1 and 2 of user selection algorithm, given \( \theta_{\text{max}}, x_s^y \) is retrieved. In line 3, the algorithm verifies whether the link has sufficient capacity to provide at least \( x_s^y \) and that \( x_s^y \) has not been included from the previous run, and then, execute line 4, 5, and 6. Next, \( \theta_{\text{max}} \) is removed from the set in line 6. We assume that the network begins to provide service as soon as the user is admitted into
network. The performance of this algorithm is determined by the number of links $|L|$, the number of users $|S|$, and the number of links in the path of each admitted user $s$ that need to be updated. Thus, the total running time is at most $O(|L|^2.|S|)$.

### B. Convergence

In this section, we show Algorithm 1 achieves convergence. Let $U_s^{*}(x_s, x_2^y)$ be maximum user utility of user $s$, given the price $\lambda_s$ that he/she is willing to pay and bandwidth allocation $x_s$ allocated to him/her, such that $x_s^y \leq x_s$, and $\lambda_s \leq \lambda$. Moreover, every user $s \in \bar{S}$ that is selected through admission control of Algorithm 1, is guaranteed that he/she can achieve $U_s^{*}(x_s, x_2^y)$. In order to demonstrate that the algorithm converges, we first show that every link in the network is feasible [1, 2], which also means that the total demand for bandwidth in every link in the network does not exceed its capacity. Let $\bar{S}(l)$ be a set of selected users by algorithm 1 whose traffic traverse through link $l$.

**Theorem 1.** $\forall l, l \in L$, Algorithm 1 guarantees $\sum_{s \in \bar{S}(l)} x_s \leq C_l$.

**Proof.** While considering a new user $s'$ in set $\bar{S}$ for inclusion, line 3 in the algorithm 1 verifies whether additional bandwidth demand $x_s'$ of $s'$ will cause the total aggregated demand to exceed any link capacity on path $r_s$, for $\forall r \in r_s$. When any of the link $l \in r_s$ exceeds the link capacity $C_l$, user $s'$ will not be included in $\bar{S}$, which is described in the following expression.

$$ s' \in \bar{S} \mid \forall r \in r_s', \ x_s' + \sum_{s \in \bar{S}(l)} x_s \leq C_l $$

Otherwise, $s' \notin \bar{S}$. This completes the proof. ■

Theorem 1 demonstrates that admission control guarantees that every link is feasible. Next, we demonstrate that algorithm 1 also achieves convergence.

**Theorem 2.** Algorithm 1 achieves convergence.

**Proof.** By theorem 1, we have $\forall l \in L, \sum_{s \in \bar{S}(l)} x_s \leq C_l$. That is every link in the network is feasible. As a result, price $\lambda^{t+1}$ determined by e.q. (6) satisfies $\forall l \in L, \lambda^{t+1} \leq \lambda^t$. Moreover, since $\lambda_{min} \leq \lambda$, the price $\lambda^t$ converges at $\lambda_{min}$, $t \rightarrow \infty$. As a result the price converges, which also implies bandwidth allocated to selected users also converges. Therefore, the algorithm 1 converges. ■

Theorem 2 implies Algorithm 1 converges when every link in the network is feasible. This is possible because theorem 1 shows that algorithm 1 can guarantee that links in the network are feasible. So far, we have demonstrated how the algorithm achieves convergence. In the next discussion, we will endeavor to provide an insight why convergence is possible through Nash Equilibrium [29]. To address this question, we investigate the likelihood of selected users changing their bandwidth demand after they are selected by Algorithm 1. In the following discussion, we assume that price decided by network and the price ceiling that a user is willing to pay do not change. Additionally, we also assume that each user does not know the maximum price that other users are willing to pay.

**Theorem 3.** Algorithm 1 achieves Nash Equilibrium.

**Proof.** First, we demonstrate that user has minimal to gain by changing their demand for bandwidth. By definition described in eq. (1), users satisfaction $U_s(x_s, x_2^y)$ has diminishing return characteristic because

1. $\lim_{y \rightarrow \infty} U_{bw}(x_s^y) - U_{bw}(x_s^{y-1}) = 0$, as described in sec.2.

2. $\lim_{\lambda_s \rightarrow 0} U_{bw}(x_s) = 1$.

Therefore, when user $s$ utility reaches $U_s(x_s, x_2^y) = U_s^*(x_s, x_2^y)$, demanding more bandwidth may not significantly improve his/her satisfaction of enjoying video streaming application. Following this argument, lowering the network price therefore may not encourage users to demand more bandwidth. Thus, no users benefit from demanding more bandwidth when $U(x_s, x_2^y) = U^*_s(x_s, x_2^y)$. Additionally, it is unlikely for users to reduce their bandwidth demand to achieve lower utility, such that $U_s(x_s, x_2^y) < U_s^*(x_s, x_2^y)$. For these reasons, there is a set of unique user utilities $U^*_s(x_1^1, x_2^y), U^*_s(x_2^1, x_2^y), ..., U^*_s(x_s^y)$, that are associated with users that are selected by Algorithm 1. Hence, $U^*_s(x_s, x_2^y), \forall s \in \bar{S}$ are the unique user utilities in Nash Equilibrium. This completes the proof. ■

Theorem 3 implies that bandwidth convergence is possible because users selected by Algorithm 1 has characteristics of Nash Equilibrium when users have minimum to gain by changing their own bandwidth demand [29].

### VI. SIMULATION AND DISCUSSION

In this section, we present a demonstration of multi-layered utility function with admission control using a network shared by eight users, shown in figure 2. The initial setup is listed in table 1 and table 2. Through this simulation written in C++, we demonstrate how user 3’s switching between two layers may impact other users in the network and the pricing. We also show how implementing admission control improves network activities.

![Fig. 2. Network Topology.](image)
1, and 2 also oscillates, as illustrated in figure 5.a. At iteration 300, network implements admission control with user selection algorithm, resulting in the dismissal of user 0, and this in turn leads to the convergence of network price and bandwidth allocation for user 1, 2, and 3. In addition, without user 0, the network has additional bandwidth to meet the higher demand of user 3.

<table>
<thead>
<tr>
<th>User</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m )</td>
<td>40</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>30</td>
<td>10</td>
<td>100</td>
<td>70</td>
</tr>
<tr>
<td>( x^{\text{min}} )</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>New ( x^{\text{min}} )</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>5</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 1. Simulation setup.

<table>
<thead>
<tr>
<th>User route or path setup.</th>
</tr>
</thead>
<tbody>
<tr>
<td>User route</td>
</tr>
<tr>
<td>Uc: ABCD</td>
</tr>
<tr>
<td>Uc: AB</td>
</tr>
<tr>
<td>Uc: CDG</td>
</tr>
<tr>
<td>Uc: EF</td>
</tr>
</tbody>
</table>

Table 2. User route or path setup.

Next, we examine the ripple effects from the oscillation caused by user 3. The rate allocation in link BC shows a pattern similar to the allocation in link AB because the same flows (user 0, 1, and 2) traversing through link BC also traverse in link AB. This is illustrated in figure 6.a, the rate allocation of user 5 and 6 in link DG converge and is not affected by the oscillation in CD. However, the oscillation from link AB is affecting user 4, such that his/her bandwidth allocation is also oscillating. This is because user 4 is sharing link DE with an oscillating flow that belongs to user 1, as illustrated in figure 7.a and 7.b. User 3’s oscillation has impacted link DE because the aggregated flow exceeds the maximum capacity. Thus, any spike in the transmission rate causes congestion in DE. That is, the aggregated flow in DE exceeds the capacity limit and it forces the network to hike the price at link DE. User 1 stops oscillating after the admission control is invoked at link AB, and this causes user 4 to also stabilize. Lastly, since user 7 is not sharing link EF with anyone, user 7 is not affected at all.

The simulation shows that active users striving for better QoE may cause many ripple effects, causing rate allocation assigned to other users to oscillate. Furthermore, the ripple effects of bandwidth oscillation may spread from one specific link to other parts of the network. The oscillation and its ripple effects confirm that solving optimization involving adaptive QoE makes the optimization problem become intractable. This is because there is a circular event of users continuously adjusting their transmission rate according to the fall and rise of the price. At the same time, price fluctuation follows and depends on the rise and fall in bandwidth demand. Additionally, the simulation also shows that higher throughput may result in higher risk of ripple effects, which lead to higher network instability. Such the ripple effects may cause more users experiencing the visual flickering.
effect and user QoE degradation. That is price and bandwidth allocation oscillation at different part of the network. Admission control by the network is a viable approach to attend to and stop oscillation. This is because reducing the population in the network provides sufficient bandwidth for the admitted users to increase their demand until the desired QoE is achieved. The lessons learned in overcoming the oscillation are: firstly, network may pick the higher value in the price oscillation as the network price, hence users may have to settle with the QoE they can afford. Secondly, network must assure that it has sufficient bandwidth for admitted users to be able to achieve the desired QoE.

VII. IN PRACTICE

In this paper we explicate the nature of the problem concerning optimizing multi-layered utility, specifically on how user desire for better quality may impact the network traffic and proposes a design of multi-layered utility model with the following benefits. Our multi-layered utility function may provide an upper bound for bandwidth allocation and a lower bound of how much video traffic can be reduced without significantly impacting user QoE. Secondly, our model provides useful guidance for the design of optimization algorithm for video streaming based applications such that user QoE is maximized. Thirdly, the proposed model also provides the first step to understand the relationship between user QoE and network traffic management.

The following paragraphs describe how our model can be implemented in various scenarios. Applying admission control and selecting millions of incoming traffic flows in Internet may not be practical and effective because network topology in the Internet is too large to be managed by a single provider. However, we can scale down the implementation for smaller network topologies, such as private networks where video traffic is managed internally. Some examples of such networks are CDN providers that provide live video streaming [30], Google’s Youtube datacenter [36], local Internet providers that directly provide service to users, etc. In such smaller private networks, computation of the selection of appropriate video traffic flows can be done quicker, and they also provide a more controlled environment compared to the Internet. Our proposed admission control is particularly useful for such types of environments. Therefore, although we cannot control the traffic in the Internet, implementation of our model improve traffic management and guarantees user QoE in internal networks. We also provide a general discussion on the implementation of our solution in secondary data market [33,35]. Finally, we conclude this section with a discussion on how providers should decide per link cost.

Content Distribution Network (CDN). Figure 8 illustrates the high level structure of CDN’s network video delivery system [30]. A CDN’s internal network consists of three logical parts: video sources, reflectors, and edge clusters. (i) Video sources: Servers that retrieve videos from the originating sites (for example YouTube.com, Hulu.com, etc.) into the CDN’s system, (ii) reflector: servers that forward the video content internally, they act as intermediaries between video sources and the edge clusters, where each reflector can receive one or more streams from the video sources and can send those streams to one or more edge clusters. Note that a reflector is capable of making multiple copies of each video stream received, and each copy can be sent to a different edge cluster. This feature enables the rapid replication of a stream to a large number of edge clusters to serve a highly-popular video. (iii) edge clusters directly serve end-users. When a client's request for a particular channel arrives at an edge cluster, the edge cluster forwards it to a reflector, which in turn forwards it to a source; the content is returned via the reverse path. When multiple requests for the same content arrive at the same reflector server, only one request is forwarded upward. The end result is a distribution tree for each video from sources to end users. Video data stream is delivered between logical structures through CDN’s internal network.

To meet users’ minimum QoE requirements, the proposed admission control can be implemented in CDN’s internal network, such that bandwidth oscillation can be avoided internally. Network between edge servers and reflectors is particularly important in assuring user’s QoE requirement because the edge servers directly serve the end-users and any oscillation in this part of network will be experienced by users. Thus, although traffic behavior in the Internet is unpredictable, guaranteeing good performance in CDN’s internal network may improve the overall performance of video stream delivery to end users and in meeting users’ requirements for QoE.

Fig. 8. CDN’s network structure, which consists of Video Sources, reflectors, and edge clusters.

Fig. 9. Examples of datacenter topology. (a) Fat-Tree topology [31]. (b) Google’s Clos topology [32].

Datacenter Network. Today’s datacenter architecture is typically a three to four-layer multi-rooted tree topology [31,32]. The leaves are the servers and the upper layers are populated by
commodity switches, as illustrated in figure 9. Such design allows multi paths between two end points (severs) and at the same time provides load balancing and fault tolerance. Two examples of a typical datacenter network topologies are Fat-Tree [31] and Clos based topology, the latter is adopted by content providers like Google [32] (including YouTube datacenter [37]) and Facebook [36]. In a datacenter that serves video applications, video data is streamed from one of the servers where the video is stored, through an internal datacenter network, to a gateway server before being finally delivered to the end users. One of challenges in managing an internal traffic in a datacenter is that bottleneck may occur when there are multiple video streams competing for the same resources [40]. Moreover, as we have shown previously, link bottleneck may lead to bandwidth oscillation, which may degrade user QoE. Our admission control can be considered to be implemented in a datacenter to mitigate bandwidth oscillation. This also assures that datacenter network provides supports to meet users QoE requirements.

**Fig. 10.** Video data is streamed from content provider (e.g. YouTube), to Transit ISP, to Eyeball ISP, and to end users.

**Eyeball internet service provider** is an ISP that provides Internet access to individual users and is responsible for the last-mile connectivity [38]. As illustrated in Figure 10, video data is streamed from content providers (e.g. YouTube, etc.) to end users through transit ISP, which is responsible for the backbone network, and then finally through eyeball ISP before reaching end users. Since eyeball ISPs are directly serving end users, last-mile connection quality have a direct impact on user QoE. Thus, to avoid oscillation, our multi-layer utility model provides an estimation of the number of users that can be supported while meeting user QoE requirements. Additionally, our proposed admission control can be utilized to select which users eyeball ISP can support.

**Secondary Data Market** [34] is a concept of local Wi-Fi providers (for e.g., coffee shops, airports, hotels, etc.) providing Internet access to end users by reselling bandwidth subscribed from ISP. As video streaming applications become more popular, these Wi-Fi providers should consider providing supports for video streaming based applications to their customers. Therefore, similar to previous case studies, multi-layered utility model may be utilized to estimate the amount of bandwidth required to provide proper QoS for video streaming application to meet user QoE requirements. Moreover, Wi-Fi providers may leverage the proposed admission control to ensure that there is sufficient bandwidth to meet user QoE requirements. Similar approach can be applied to secondary market for mobile data [33,35,41] environment illustrated in figure 11. This is a framework which allows an individual mobile Internet subscriber to resell their unused bandwidth to other users. Here, the subscriber may utilize our model to determine whether the subscriber has sufficient amount of bandwidth to provide acceptable QoE to their buyers that are buying bandwidth for video streaming. Moreover, the subscriber may also employ our admission control scheme to select a group of buyers that he/she can support, such that his/her customers’ QoE requirements are met.

**Discussion on link cost.** Determining link cost or how much providers should charge per link generally depends on two factors: operational cost and market pricing. Operational cost is typically determined by the cost required to manage the infrastructure and perform administration tasks [7]. Examples of such costs include the cost of employing network administrators, the cost of electricity, etc. As for market price, it is influenced by the dynamic and complex relationship between supply and user demand for bandwidth [33,38]. The complexity of deciding a market price increases when there are dependencies between multiple parties involved in providing a service to users. For example, in secondary market, mobile Internet subscribers (users) are also the providers. This may create two layers of supply and demand which may lead to market price saturation [33]. Another example, amount of data an Eyeball ISP can deliver to its users are often determined by bandwidth availability at the transit ISP’s network, as illustrated in figure 10. Therefore, more bandwidth in Eyeball ISP’s network does not translate to more supply when the network that belongs to transit ISP is experiencing heavy congestion. These concerns are however outside of the scope of this paper. For more detailed discussions on operational cost and market pricing, please refer to [7,33,38].

**VIII. CONCLUSION**

General solutions for NUM problems in single-layered environment are not sufficient for traffic problems in multi-layered multimedia applications where user utility is adaptive. Our multi-layered user utility function incorporates insights from the fields of computer graphic and HCI, resulting in a user-utility function that considers human’s natural visual ability to perceive changes in image quality, influencing their demand for desired QoE. This translates to a user utility demand that follows the law of diminishing returns. We demonstrate that the adaptive demand of some users cause oscillation that may ripple through the network, leading to lower aggregate QoE. Our study shows that optimization problem with users who constantly pursue better QoE makes the optimization problem intractable. This desire for better quality also causes the visual flickering effect at the video application level, which degrades user QoE. Thus, we
propose a greedy based solution for admission control, such that the balance between revenue and user satisfaction can be achieved. This allows the rate allocation algorithm to converge. However, even when the network seeks to maximize its revenue during the process of admission control, the algorithm may not yield the maximum revenue because of the nature of greedy algorithm. Furthermore, the efficiency of multi-layer utility function must be investigated, which will be addressed as part of our future work.

Acknowledgement. This research was supported in part by the NSF under Awards CNS-0953620 and CNS-1527303.

REFERENCE