A Suboptimal Network Utility Maximization Approach for Scalable Multimedia Applications

Mohammad S. Talebi[*], Ahmad Khonsari[†*], Mohammad H. Hajiesmaili[*†]
and Sina Jafarpour[‡]
[*] School of Computer Science, IPM, Tehran, Iran
[†] ECE Department, University of Tehran, Tehran, Iran
[‡] Department of Computer Science, Princeton University,
Emails: {mstalebi, ak, hajiesamaili}@ipm.ir, sina@cs.princeton.edu

***Abstract*—Wired and wireless data networks have witnessed an explosive growth of inelastic traffics such as real-time or media streaming applications. Recently, applications relying on layered encoding schemes appeared in the context of live-streaming and video and audio delivery applications. This paper addresses the Network Utility Maximization (NUM) for scalable multimedia transmission which is relying on layered encoding schemes. Non-convexity of the NUM problem for such applications makes dual-based approaches incompetent, whereby achieving optimality proves quite challenging. We adopt the staircase utility function and formulate the underlying optimization problem. To tackle the non-convexity of the problem, we use a smooth approximation of the staircase utility function and propose a dual-based distributed algorithm for rate allocation and bandwidth sharing in such scenarios. Numerical results show that the proposed algorithm achieves suboptimal yet efficient solution.**

I. Introduction

Over the past few years, the usage of multimedia applications in computer networks has been growing explosively. Recently, applications with streaming traffic such as live-streaming over peer-to-peer and wireless ad-hoc networks are emerged and expected to continue growing. Thus, in the course of recent years, the research community has witnessed the emergence of new demands for QoS-provisioning in different multimedia applications. In order to tackle this issue, many technical challenges have to be addressed in the two areas of video coding and networking.

In the video coding area, QoS-provisioning is efficiently dealt with in the context of video adaptation paradigms [1]. The problem of video adaptation has been widely addressed through a number of approaches, such as *Scalable Coding* [2], [3], *Transcoding* [5], [6], and *Summarization* [7]. In *Scalable Video Coding*, the objective is to enable the encoding of a high-quality video bitstream that contains one or more valid and decodable subset bitstreams. *Transcoding* is normally referred to as techniques where a compressed media bitstream format is converted into another format. *Video Summarization* schemes through content analysis and optimization select a subset of frames from the video sequence to form a concise representation of the sequence, while incurring as small of a loss as possible.

In the area of networking, rate allocation is at the nexus of a wide variety of paradigms, whose boundaries extents different scenarios ranging from resource-constrained networks to QoS-aware ones. This also has been the issue of primary concern in the research community of multimedia applications. Following the seminal work by Kelly *et al.* [8], the optimization flow control approach was proposed by Low *et al.* [9], in which the optimal rate allocation of a wired network under elastic traffic was modeled and led to a dual-based distributed algorithm for rate allocation both in synchronous and asynchronous environments. Within the previous decade, the work by Low *et al.* was followed by the network research community and led to a more general optimization framework known as Network Utility Maximization (NUM) and its generalized form, GNUM ([10] and references therein). The underlying assumption of these works is that the network traffic is elastic, whose characteristics can be modeled by a strictly concave utility function. Such utility functions, make the problem convex and thereby tractable for optimality analysis.

On the other hand, Internet has witnessed an explosive growth of inelastic traffics such as those arising in real-time or media streaming applications. Such applications are relying on tight performance characteristics, in terms of rate (bandwidth), delay, jitter, etc., which make the utility function non-concave [11]. Non-concave utility functions result in non-convex NUM problems, whose analysis proves quite challenging. So far, only few works have tackled the non-convex NUM problems, e.g. [12]-[14]. In [12], the authors adopted *sigmoidal-like* utility function that is an appropriate choice for the utility of rate-adaptive multimedia applications, and proposed a distributed admission control approach for such utilities, called "self-regulating" heuristic. Hande *et al.* in [13] investigated the optimality conditions for the distributed iterative dual-based algorithm to converge to global optimal despite using non-concave utility functions. In [14], an efficient but centralized method based on sum of square approach is developed to compute the global optimal rate allocation for types of non-concave utility functions that can be transformed into polynomial functions.

Contemporary to the research studies carried out in the context of NUM, a lot of recent studies have dealt with the inelastic multimedia applications through modeling the traffic characteristics of such applications. These studies appeared in different forms. Huang *et al.* [15], proposed a resource allocation solution for multi-user video streaming over cellular wireless networks. They developed a NUM framework with a resource pricing algorithm via previous well-established dual-based algorithms. The resource price is obtained in turn, is used to derive source content adaptation to each user, using video summarization techniques [7]. In [16], a content-aware distortion-fair networking framework with joint video source adaptation and network resource allocation is developed. A basic difference in this work is that an explicit utility function for sources is not considered. Instead, a content-aware time-varying utility function is chosen that is different per each frame as well as per video content. Based on the idea of dropping less important frames, a distributed iterative algorithm is proposed to achieve min-max distortion fairness. The main superiority of this work is taking into account the special characteristics of video content such as dependency between frames.

In this paper, our focus is on the rate allocation for multimedia applications with scalable encoding based on NUM approaches. Today, a plethora of such applications exist in video and audio delivery systems and are relying on layered encoding schemes [2], [3]. For scalable multimedia applications, rate allocation is limited to distinct levels of utility, i.e. the utility is increased only when a higher layer can be delivered due to increase in the available bandwidth. Thus, the ideal utility for these applications is in the form of a *staircase* function, which is shown in Figure 1 [4]. In order to deal with the nondifferentiable and non-concave behavior of the staircase utility, we introduce *multimodal sigmoid approximation* as a smoothed and well-behaved utility function and to remedy nondifferentiability of the staircase utility. Moreover, we aim at approximating the underlying NUM in order to come up with more amenable formulation, and then propose a dual-based distributed algorithm as the solution to it. To the best of our knowledge, this is the first work that addresses NUM problem for scalable multimedia transmission. Numerical results present a proper validation of our endeavor in achieving a suboptimal yet efficient solution.

The rest of the paper is organized as follows. Section II describes the network model and utility approximation. Section III is devoted to formulate the underlying NUM problem. Section IV investigates the optimality condition and optimal solution to the NUM. The optimal dual-based distributed algorithm is presented in Section V. Numerical results are presented in Section VI and VII concludes the paper and outlines some future directions.

II. System Model

A. Network Model

We consider a network consisting of a set of sources denoted by $\mathcal{S} = \{1, \ldots, S\}$ and a set of unidirectional links, denoted by $\mathcal{L} = \{1, \ldots, L\}$. Let x_s and c_l be the source rate for source s and capacity of link l, both in bps, respectively. Without loss of generality, we assume that source rate of source s is limited so as to certify

$$0 < m_s \leq x_s \leq M_s < \infty \qquad (1)$$

where m_s and M_s denote the minimum and maximum rates for source s, respectively. We assume that source s, when submitting at rate x_s bps, attains a utility function $U_s(x_s)$, which models its benefit.

We associate with source s a *path*, i.e. a set of links $\mathcal{L}(s) \subseteq \mathcal{L}$, that determines the links that source s passes through. Similarly, we define $\mathcal{S}(l) \subseteq \mathcal{S}$, to be the set of sources traversing link l. For the sake of simplicity, we define the routing matrix as $\mathbf{R} = [R_{ls}]_{L\times S}$, where R_{ls} is defined as

$$R_{ls} = \begin{cases} 1 & \text{if source } s \text{ passes through link } l \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

B. Utility Model

As stated in Section I, the utility function of such applications can be ideally characterized using a non-concave and non-differentiable utility function referred to as *staircase* utility function, which is shown in Figure 1 in solid line [4]. Non-concavity of the staircase utility functions implies that the conventional theory of Network Utility Maximization (NUM) cannot be used for such functions. In order to deal with such non-concave and non-differentiable utility functions, we use a smoothed approximation of it. Figure 1 shows the idea behind such an approximation. In this figure, the curve in dashed line represents the smoothed approximation of the staircase function.

In order to construct such a smoothed approximation, we divide its domain into nonoverlapping intervals, so that a step transition occurs within the midpoint of each interval. The step transition i, i.e. the part of the curve in which utility function jumps from level i, (i.e. $U(x) = i$), to level $i+1$, (i.e. $U(x) = i + 1$) is smoothed and approximated by a *sigmoid-like* function, whose point of inflection corresponds to $U(x) = \frac{i+1}{2}$.

A sigmoid-like function has been well studied in the field of neural networks. The most commonly used form of sigmoid-like function is the *logistic function* defined as

$$F(x, \alpha, \beta) = \frac{1}{1 + e^{-\alpha(x-\beta)}} \qquad (3)$$

It is easy to show that β is the inflection point of $F(x)$, i.e. for $x < \beta$, $F(x)$ is convex, and for $x > \beta$ it is concave. Moreover, $\alpha > 0$ is a parameter that determines the sharpness of its curve. It's worth mentioning that α must be chosen sufficiently large so as to effectively capture the sharp transition of an increase in the utility level.

Using the notation for the sigmoid-like function introduced above, we then represent the approximation shown in Figure 1 in dashed line. Recall the interval division of the domain introduced above. Then, for the step transition i, i.e. jump from $U(x) = i$ to $U(x) = i + 1$, we have,

$$\tilde{U}(x) = F(x, \alpha, ki) + i; \qquad x \in \left[ki - \frac{k}{2}, ki + \frac{k}{2}\right] \qquad (4)$$

where $\tilde{U}(.)$ denotes the approximation to the original utility function and $[ki-\frac{k}{2}, ki+\frac{k}{2}]$ is the interval in which transition i occurs. It's worth mentioning that k is the required rate increase to advance the utility U by 1. Hence, $1/k$ can be thought of as the slope of the straight line passing through the midpoint of step transitions.

Combining all of the intervals, we get

$$\tilde{U} = \begin{cases} F(x,\alpha,k)+1 & x \in [k-\frac{k}{2}, k+\frac{k}{2}] \\ . & \\ . & \\ F(x,\alpha,ki)+i & x \in [ki-\frac{k}{2}, ki+\frac{k}{2}] \\ . & \\ . & \\ F(x,\alpha,kN)+N & x \in [kN-\frac{k}{2}, kN+\frac{k}{2}] \end{cases} \tag{5}$$

where it is assumed that the domain is divided into N equal intervals, corresponding to N encoding layer.

In statistics, a sigmoid-like function, which is in possession of a single point of inflection, is usually referred to as *unimodal function*. Our approximated staircase utility function is comprised of several sigmoid-like functions, and thereby has several points of inflection. Thus, it is a *multimodal function* as opposed to the unimodal case. In this respect, we refer to this approximation as the *multimodal sigmoid*. The multimodal sigmoid approximation presented above is non-differentiable in general; however, if α is chosen sufficiently large, discontinuity gap between contiguous steps vanishes and thereby makes it continuous.

Sources in the network may demand for different QoS requirements; hence, it makes sense that each source s, would advance its utility according to its own k_s factor, which may differ from the others. Moreover, each source s is assigned a positive weight w_s which can be used to address its priority in rate allocation. Such weights are normalized so as to satisfy $\sum_s w_s = 1$. Therefore, the (approximated) utility function of source s is

$$\tilde{U}_s(x_s) = w_s \tilde{U}(x_s, \alpha, k_s) \tag{6}$$

where $\tilde{U}$ is defined by (5) and k_s and α were omitted from the notation.

III. Problem Formulation

In this paper, we mainly focus on modeling a convex optimization as an approximation to the non-convex NUM arising in scalable multimedia applications. Thus, for the sake of simplicity, we consider the simplest form of the NUM, i.e. the optimization flow control problem, introduced in the seminal work of Low *et al.* [9]. The objective of such a simple NUM is to choose source rates so as to maximize the aggregate utility of all sources while satisfying capacity constraints, as follows

$$\max_{\mathbf{x}\in\mathcal{X}} \sum_{s=1}^{S} \tilde{U}_s(x_s) \tag{7}$$

subject to:

$$\sum_s R_{ls} x_s \le c_l; \quad l \in \mathcal{L} \tag{8}$$

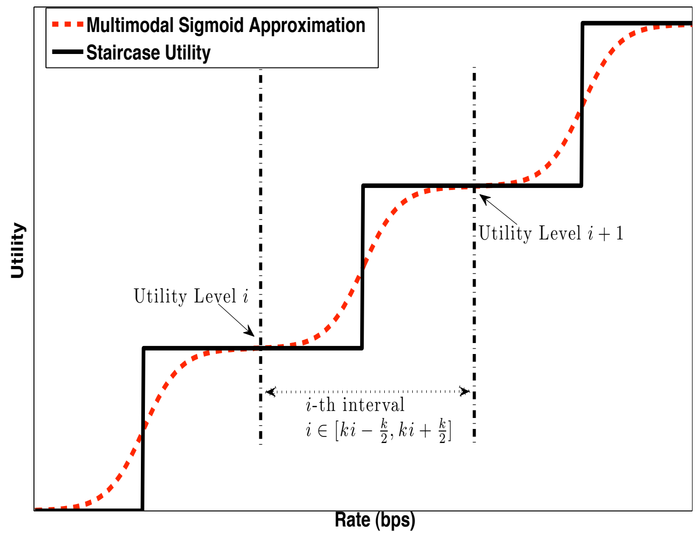

Fig. 1. Staircase Utility Function (solid line) and Its Multimodal Sigmoid Approximation (dashed line)

where $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_S$ denotes the Cartesian product of all rate domains $\mathcal{X}_s = [m_s, M_s]$. In order to come up with a more amenable formulation, we consider the following optimization problem

$$\max_{\mathbf{x}\in\mathcal{X}} \sum_{s=1}^{S} \log \tilde{U}_s(x_s) \tag{9}$$

subject to:

$$\sum_s R_{ls} x_s \le c_l; \quad l \in \mathcal{L} \tag{10}$$

The following theorem shows that the problem (9)-(10) approximates the problem (7)-(8).

Theorem 1: The optimization problem (9)-(10) approximates the problem (7)-(8).

Proof: Taking the logarithm of the objective of (7) yields

$$\max_{\mathbf{x}\in\mathcal{X}} \log \left(\sum_s \tilde{U}_s(x_s) \right) \tag{11}$$

Since $\log(.)$ function is monotone increasing, maximizing (11) is equivalent to maximizing (7), and thereby problem (11) is equivalent to problem (7)-(8). On the other hand, $\log(.)$ is a concave function, for $z_s > 0$, we have

$$\log(\sum_s w_s z_s) \ge \sum_s w_s \log(z_s) \tag{12}$$

provided $w_s \ge 0$ and $\sum_s w_s = 1$. Substituting utility functions, $\tilde{U}_s = w_s \tilde{U}$ into (12), we get

$$\log(\sum_s w_s \tilde{U}(x_s)) \ge \sum_s w_s \log \tilde{U}(x_s) \tag{13}$$

Therefore, the transformed objective of (7) is lower bounded by the R.H.S of (13). Thus, to obtain an approximate to problem (7), we choose its lower bound as the objective function. ■

We defer solving the optimization problem until the next section.

IV. Optimal Solution

A. Deriving Primal Optimal

We can solve the optimization problem (9)-(10) using dual-based approach[1]. However, compared to the conventional methods for the NUM proposed so far [10], it demands for partly more elaboration. We obtain the dual problem using the definition of Lagrangian. The Lagrangian of (9) is given by

$$L(\mathbf{x}, \mu) = \sum_s \log \tilde{U}_s(x_s) - \sum_l \mu_l \left(\sum_s R_{ls} x_s - c_l \right)$$

where μ_l is the positive Lagrange multiplier associated to capacity constraint (10) for link l and $\mu = (\mu_l, l \in \mathcal{L})$ is the vector of Lagrange multipliers.

The approximated optimization problem introduced above is non-convex. In order to come up with a convex formulation, we use the following transformation

$$\tilde{x}_s = e^{\alpha x_s} \tag{14}$$

Since this transformation is monotonic increasing, maximizing $L(\tilde{\mathbf{x}}, \lambda)$ is equivalent to maximizing $L(\mathbf{x}, \lambda)$. Rewriting the Lagrangian with this transformation, we get

$$L(\tilde{\mathbf{x}}, \mu) = \sum_s \log \tilde{U}_s(\tilde{x}_s) - \sum_l \mu_l \left(\sum_s \frac{R_{ls}}{\alpha} \log x_s - c_l \right) \tag{15}$$

We denote the primal-optimal point of the approximated problem by $\mathbf{x}^* = (x_s^*, s \in \mathcal{S})$. Based on KKT Theorem [17], at optimal point the following conditions must be satisfied:

$$\nabla_{\tilde{\mathbf{x}}} L(\tilde{\mathbf{x}}, \mu)|_{(\tilde{\mathbf{x}}^*, \mu^*)} = \mathbf{0} \tag{16}$$

$$\sum_s \frac{R_{ls}}{\alpha} \log \tilde{x}_s^* \leq c_l; \qquad l \in \mathcal{L} \tag{17}$$

$$\mu_l^* \geq 0; \qquad l \in \mathcal{L} \tag{18}$$

$$\mu_l^* \left(\sum_s \frac{R_{ls}}{\alpha} \log \tilde{x}_s^* - \mathbf{c}_l \right) = 0; \tag{19}$$

where $\mathbf{0}$ is a vector, all of whose element is zero.

Substituting (15) into (16) yields

$$\begin{aligned} \frac{\partial L}{\partial \tilde{x}_s} &= \frac{d}{d\tilde{x}_s} \log \tilde{U}_s(\tilde{x}_s) - \frac{1}{\alpha \tilde{x}_s} \sum_l R_{ls} \mu_l \qquad (20) \\ &= \frac{\tilde{U}_s'(\tilde{x}_s)}{\tilde{U}_s(\tilde{x}_s)} - \frac{1}{\alpha \tilde{x}_s} \sum_l R_{ls} \mu_l \\ &= \frac{F'(\tilde{x}_s, \alpha, k_s i_s)}{F(\tilde{x}_s, \alpha, k_s i_s) + i_s} - \frac{1}{\alpha \tilde{x}_s} \sum_l R_{ls} \mu_l = 0 \end{aligned}$$

[1]Due to space limit, we relegate the proof of the convexity to our future works.

where it is assumed that x_s^* falls within the i_sth interval. Substituting $F(\tilde{x}_s, \alpha, k_s i_s)$ into the above result and doing some algebraic manipulations, we get

$$\frac{\frac{A_{si}}{(\tilde{x}_s + A_{si})^2}}{\frac{\tilde{x}_s}{\tilde{x}_s + A_{si}} + i_s} - \frac{\mu^s}{\alpha \tilde{x}_s} = 0 \tag{21}$$

where $\mu^s = \sum_l R_{ls} \mu_l$ and $A_{si} = e^{\alpha k_s i_s}$. Further simplification of (21) yields

$$\frac{A_{si}}{(1 + i_s)(\tilde{x}_s + B_{si})(\tilde{x}_s + A_{si})} = \frac{\mu^s}{\alpha \tilde{x}_s} \tag{22}$$

where $B_{si} = \frac{i_s}{i_s + 1} A_{si}$. The above result leads to the following quadratic equation Obtaining x_s^* demands for solving (21).

$$\tilde{x}_s^2 + \left(A_{si} + B_{si} - \frac{\alpha A_{si}}{(1 + i_s)\mu^s} \right) \tilde{x}_s + A_{si} B_{si} = 0 \tag{23}$$

Solving (23) yields $\tilde{x}_s^*$ as follows

$$\tilde{x}_s^* = A_{si} \frac{\frac{\alpha}{\mu^s} - 2i_s - 1 + \sqrt{(1 - \frac{\alpha}{\mu^s})^2 - \frac{4 i_s \alpha}{\mu^s}}}{2(1 + i_s)} \tag{24}$$

Since the above equation must have a real solution, we deduce that

$$i_s^* = \lfloor \frac{\mu^s}{4\alpha} (1 - \frac{\alpha}{\mu^s})^2 \rfloor \tag{25}$$

Optimal source rate can be obtained simply by taking the inverse transformation as follows

$$x_s^* = [\frac{1}{\alpha} \log \tilde{x}_s^*]_{\mathcal{X}_s} \tag{26}$$

where $[.]_{\mathcal{X}_s}$ is the projection operator on the $\mathcal{X}_s$.

We then proceed to solve the problem through its dual. The dual problem is defined as

$$\max_{\mu \geq 0} D(\mu) \tag{27}$$

where $D(\mu)$ is dual function and is defined as the maximum of the Lagrangian over $\mathbf{x}$, i.e.

$$D(\mu) = \min_{\tilde{\mathbf{x}} \in \tilde{\mathcal{X}}} L(\tilde{\mathbf{x}}, \mu) \tag{28}$$

Problem (28) is an unconstrained optimization problem and is already solved by $\tilde{\mathbf{x}}^*$. Therefore, for the dual function we get

$$D(\mu) = L(\tilde{\mathbf{x}^*}, \mu) \tag{29}$$

B. Solving Dual Problem

Now we are ready to solve the dual problem (27). In order to obtain a distributed solution, we will solve the dual problem using *gradient projection method* [19]. The gradient projection method, iteratively steps toward the opposite direction of the gradient of the objective of the optimization (minimization) problem. Therefore, for the dual problem (27), we get

$$\mu^{(t+1)} = [\mu^{(t)} - \gamma \nabla D(\mu^{(t)})]^+ \tag{30}$$

or equivalently,

$$\mu_l^{(t+1)} = \left[\mu_l^{(t)} - \gamma \frac{\partial D(\mu^{(t)})}{\partial \mu_l} \right]^+ \tag{31}$$

where $\mu^{(t)} = (\mu_l^{(t)}, l \in \mathcal{L})$ is the value of μ at tth iteration step, γ is a constant step size and $[z]^+ = \max(z, 0)$. For the derivatives of $D(\mu)$, we get

$$\frac{\partial D(\mu^{(t)})}{\partial \mu_l} = c_l - \sum_s R_{ls} x_s^{(t)} \tag{32}$$

Substituting (32) into (31) yields

$$\mu_l^{(t+1)} = \left[\mu_l^{(t)} - \gamma \left(c_l - \sum_s R_{ls} x_s^{(t)}\right)\right]^+ \tag{33}$$

where $x_s^{(t)}$ is given by (26) The two update equations obtained above form an iterative algorithm as the solution to the (9), which will be discussed in the next section.

V. Optimal Algorithm

In this section, we propose a distributed algorithm based on the iterative solution obtained in Section IV.

In this subsection, we propose a distributed algorithm based on the iterative solution obtained above.

Optimal source rate equations, i.e. (26) and (24), and Lagrange multiplier update (33), derived in the previous section, can be used in conjunction with each other to form an iterative algorithm as the solution to the optimization problem (9)-(10). Lagrange multiplier is usually referred to as *shadow price* owing to the economic interpretation of its role to adjust the source rate [8], and hence thereafter we use this term instead.

For each time slot t (or iteration step t), the following key steps exist in the algorithm:

1) Each link l calculates its corresponding Lagrange multiplier (shadow price) for the next time slot, i.e. $\mu_l^{(t+1)}$, based on its previous shadow price and its aggregate traffic in the current time slot.
2) Each source s calculates its rate based on the aggregate shadow price in its path.
3) Each source s transmits the packets based on the allocated rate.

The rate control algorithm can be described as follows. For each link l, shadow price μ_l is updated according to (33) and the new shadow price result is communicated to sources traversing this link. Each source s receives from the network the shadow prices for links on its path and calculates μ^s using (33), chooses a new source rate using (26) and (24), and communicates this new rate to all the links in its path. The procedures at the links and the video sources are repeated until the algorithm converges to the optimal video rates and optimal shadow prices. The iterative algorithm for solving (27) is listed as Algorithm 1.

Algorithm 1. Dual-based Rate Control Algorithm for Scalable Multimedia

Do until $\max_s |x_s^{(t+1)} - x_s^{(t)}| < \epsilon$

At each link l,
1. Update the shadow price as following:

$$\mu_l^{(t+1)} = [\mu_l^{(t)} - \gamma \left(c_l - \sum_s R_{ls} x_s^{(t)}\right)]^+$$

At each source s,
1. Obtain the path price $\mu^{s(t)} = \sum_l R_{ls} \mu_l^{(t)}$
2. Update $x_s^{(t)}$ according to the following equation:

a. $i_s^{(t+1)} = \lfloor \frac{\mu^{s(t)}}{4\alpha}(1 - \frac{\alpha}{\mu^{s(t)}})^2 \rfloor$

b. $\tilde{x}_s^{(t+1)} =$

$$A_{si}^{\frac{-2i_s^{(t+1)} - 1 + \frac{\alpha}{\mu^{s(t)}} + \sqrt{(1 - \frac{\alpha}{\mu^{s(t)}})^2 - \frac{4 i_s^{(t+1)} \alpha}{\mu^{s(t)}}}}{2(1 + i_s^{(t+1)})}}$$

where $A_{si}^{(t+1)} = e^{\alpha k_s i_s^{(t+1)}}$

c. $x_s^{(t+1)} = \left[\frac{1}{\alpha} \log \tilde{x}_s^{(t+1)}\right]_{\mathcal{X}_s}$

where $[.]_{\mathcal{X}_s}$ is the projection operator on the $\mathcal{X}_s$.

VI. Numerical Results

In this section, we present the numerical results of the proposed iterative algorithm. Numerous validation experiments have been established, however, for the sake of specific illustration, validation results are presented.

In our scenario we consider a network with a single bottleneck link. There are 5 sources in the network, all passing through a shared link with capacity $c = 15$ Mbps. Different sources use different values of k as a result of different QoS metrics and rate requirements. Recall that k_s is the required rate increase for source s to advance the utility U_s by one. In this scenario we chose: $(k_1, \ldots, k_5) = (2.8, 2, 1.3, 0.8, 1)$. For the sake of illustration, utility functions corresponding to different values of k_s are displayed in Figure 2. Different α for all sources is set to 5 and step size is chosen to be $\gamma = 0.001$. Also, weight factors of all users are assumed to be equal. Staircase utility functions for sources are depicted in Figure 2.

The evolution of source rates is depicted in Figure 3. The evolution of the shadow price (μ) is depicted in Figure 4. Both figures reveal that the convergence is relatively fast. In order to get more insight, the rate allocation is summarized in Table 1.

VII. Conclusion

In this paper we addressed the Network Utility Maximization (NUM) for applications relying on scalable multimedia transmission using layered encoding schemes. The utility function for such applications was shown to be non-concave, which makes the NUM non-convex. Thereby dual-based NUM approaches fail to reach the optimal solution. In order to tackle this issue, we smooth the non-concave utility function using

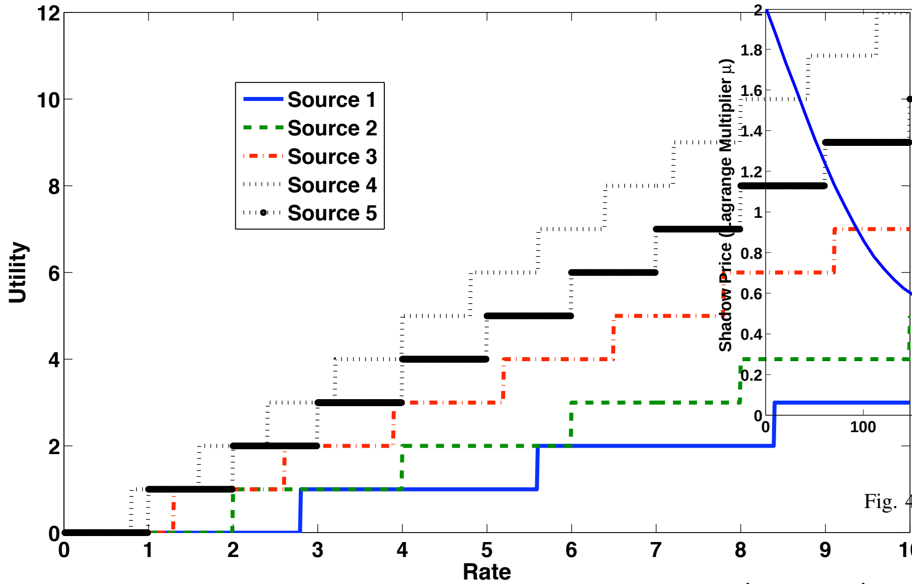

Fig. 2. Staircase Utility Functions

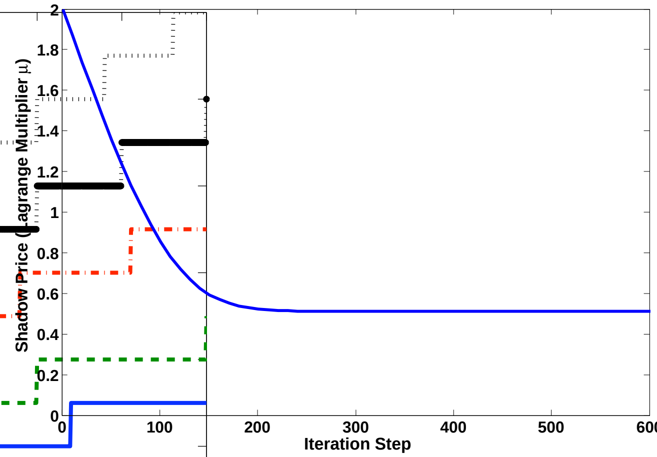

Fig. 4. Evolution of The Shadow Price

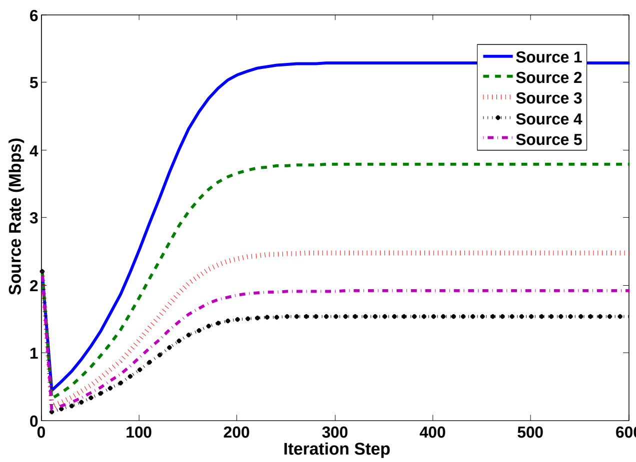

Fig. 3. Evolution of The Source Rates

sigmoidal-like approximation, which was led to formulate a convex NUM. This allowed us to propose a dual-based distributed algorithm as the solution to the approximated NUM and thereby as a suboptimal solution to the non-convex NUM. Numerical results showed that the proposed algorithm achieves suboptimal yet efficient solution. Convergence analysis of the proposed algorithm in asynchronous and time-varying environments is our primary concern for the future studies.

TABLE I
RATE ALLOCATION FOR MULTIPLE LINK CASE

Source	k_s	x_s (Mbps)
1	2.8	5.29
2	2	3.79
3	1.3	2.48
4	0.8	1.54
5	1	1.91

REFERENCES

[1] S. F. Chang and A. Vetro, "Video adaptation: concepts, technologies, and open issues," *Proceeding of IEEE*, pp. 148-158, 2005.

[2] J. Ohm, "Advances in Scalable Video Coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 42-56, 2005.

[3] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of H.264/AVC," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 17, no. 9, pp. 1103-1120, Sep. 2007.

[4] Z. Cao and E. W. Zegura,"Utility max-min: an application-oriented bandwidth allocation scheme," *Proc. IEEE INFOCOM*, pp. 793801, 1999.

[5] I. Ahmad, X. Wei, Y. Sun, and Y. Zhang, "Video Transcoding: An Overview of Various Techniques and Research Issues," *IEEE Trans. on Multimedia*, vol. 7, no. 5, pp. 793-804, Oct 2005.

[6] J. Xin, C. Lin, and M. Sun, "Digital video transcoding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 84-97, 2005.

[7] Z. Li, G. Schuster, A. Katsaggelos, and B. Gandhi, "Rate-distortion optimal video summary generation," *IEEE Trans. Image Processing*, vol. 14, no. 10, pp. 1550-1560, Oct. 2005.

[8] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, pp. 237-252, Mar. 1998.

[9] S. H. Low, and D. E. Lapsley, "Optimal flow control, I: basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861-874, Dec. 1999.

[10] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle. "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255-312, 2007.

[11] S. Shenker, "Fundamental design issues for the future Internet," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 7, pp. 1176-1188, Sep. 1995.

[12] J. W. Lee, R. R. Mazumdar, and N. Shroff, "Non-convex optimization and rate control for multi-class services in the Internet," *IEEE/ACM Trans. Netw.*, vol. 13, no. 4, pp. 827-840, Aug. 2005.

[13] P. Hande, S. Zhang, and M. Chiang, "Distributed rate allocation for inelastic flows," *IEEE/ACM Trans. Netw.*, vol. 15, no. 6, pp. 1240-1253, Dec. 2007.

[14] M. Fazel and M. Chiang, "Nonconcave network utility maximization through sum of squares method," *Proc. IEEE CDC*, Seville, Spain, Dec. 2005.

[15] J. Huang, Z. Li, M. Chiang, and A. K. Katsaggelos, "Joint source adaptation and resource pricing for multi-user wireless video streaming," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol 18, no 5, pp. 582-595, May 2008.

[16] Y. Li, Z. Li, M. Chiang, and A. R. Calderbank, "Content Aware Distortion-Fair Video Streaming in Networks," *Proc. IEEE GLOBECOM*, New Orleans, USA, 2008.

[17] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods,* Prentice-Hall, 1989.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization,* Cambridge University Press, 2003.
[19] D. Bertsekas, *Nonlinear Programming,* Athena Scientific, 1999.