Minimizing Rate Distortion in Peer-to-Peer Video Streaming

Tareq Hossain, Yi Cui, and Yuan Xue
Vanderbilt Advanced NETwork Systems Laboratory
Department of Electrical Engineering and Computer Science
Vanderbilt University, Nashville, TN 37212
Email: {tareq.hossain, yi.cui, yuan.xue}@vanderbilt.edu

***Abstract*—This paper addresses the problem of optimal rate allocation for video stream in peer-to-peer networks. We present a distributed rate allocation algorithm with the goal of minimizing the total video distortion among peers receiving the same video. The scheme assumes that video streams can be transcoded/re-quantized at intermediate peers. We deploy a double pricing solution that simultaneously incorporates the network and the relay constraints in the peer-to-peer network and compare it with a single pricing solution where the relay constraint is applied only after all the communicating peers converge to optimality. We combine our solution with two video adaptation techniques: transcoding using x264 and scalable coding using H.264 SVC (scalable video coding) extension. Our experiment shows that the double pricing solution consistently achieves a smaller aggregate distortion for all peers over the single pricing solution.**

I. Introduction

Peer-to-peer (P2P) is a powerful platform to enable a variety of multimedia streaming applications over the Internet, such as video-on-demand, video conferencing, and live broadcasting, etc. P2P system is extremely cost-effective since it utilizes the resources (CPU cycles, storage space, and uplink bandwidth) of the peer machines. Another reason for P2P's success is its instant deployability: it allows almost ubiquitous network coverage in the absence of CDN services and IP multicast.

In this paper, we present an optimal rate allocation solution for P2P video streaming applications that minimizes the aggregate rate distortion for all peers. We propose a distributed algorithm, in which each peer adjusts its own streaming rate to reach the global optimum. We claim the following contributions.

Our optimization problem formulation takes into account peer relaying, a constraint unique in P2P distribution scenario in which a peer is both receiver and sender. Peer relaying constraint ensures that the receiving rate of a peer does not exceed the receiving rate of its parent peer. This is because during the rate adaptation, the video quality, once lost, cannot be recovered. As such, the rate change occurred on one peer not only changes the video quality for itself, but also for all of its descendant peers. Therefore, price based resource allocation that considers peer relaying, ensures that peers with more children receives higher bandwidth compared to peers with fewer children. Our experiment shows that simultaneously incorporating both network and relay constraints significantly reduces the aggregate rate distortion for all peers.

We combine our solution with two video adaptation techniques: video transcoding and scalable coding. In transcoding, the video signal is changed by the relaying peer to meet a lower encoding rate, through either re-encoding or changing of key parameters such as quantization values. In scalable coding, the video signal is encoded into several layers at the source, and receivers only need to receive a subset of the layers to recover the signal with certain quality degradation. Specifically, our experiment performs transcoding with x264 [1] and scalable coding with H.264 SVC extension [2].

The rest of this paper is organized as follows. We discuss the related work in Sec. II. In Sec. III, we introduce the network model. In Sec. IV, we present the formulation of minimizing aggregate rate distortion and propose a distributed solution and its implementation. Sec. V discusses the multicast tree construction procedure, rate adaptation via transcoding or SVC, and presents simulation results and finally, we conclude in Sec. VI.

II. Related Work

Our optimization framework extends from the seminal works by Kelly [3], [4]. In this framework, network resource allocation problem is formulated to maximize user utility under resource constraints. Price based resource allocation strategies have been studied in the context of IP-unicast and multicast. Low et al. [5] extended this pricing strategy to a distributed algorithm. This price based approach has also been applied to multirate multicast by Kar et al. [6] by using sub-gradient projections and proximal approximation techniques [7]. Other works include multi-path unicast [8] and streaming [9], multicast over wireless network [10], etc. A comprehensive review can be found in [11].

Our previous study applies the same framework to overlay multicast [12], which this paper is extended upon. To the best of our knowledge, the this is the first attempt to investigate minimization of video distortion using double pricing solution.

III. Network Model

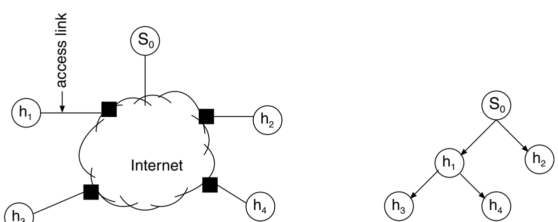
We consider a P2P network consisting of H end hosts, denoted as $\mathcal{H} = \{0, 1, 2, \cdots, H\}$. Host 0 acts as the server. Other end hosts are peers. A flow can direct from any peer to any other one, except from any peer to the server 0. We collect all these flows into the set $\mathcal{F} = \{1, 2, \cdots, F\}$. Each flow $f \in \mathcal{F}$ has a rate x_f. The rate vector then is defined as $\boldsymbol{x} = (\, x_f,\, f \in \mathcal{F}\,)$. For two peers h_k and h_i, if a flow f directs from h_k to h_i, then h_k is a parent of h_i and h_i is the child of h_k. This relationship is then denoted as $h_k \rightarrow h_i$. Furthermore, we denote $h(f)$ as the host that f directs to, which is h_i in the above example. Based on this definition, we further introduce $f(h)$ as the flow destined to peer h.

Each flow f takes place on the unicast path connecting the two peers, which encompasses a set of physical links on Internet. We collect all physical links encompassed by all flows in $\mathcal{F}$ into a vector as $\mathcal{L} = \{1, 2, \cdots, L\}$. The bandwidth of each link is c_l, collected into a capacity vector defined as $\boldsymbol{c} = (\, c_l,\, l \in \mathcal{L}\,)$. We further define $\mathcal{L}(f) \subseteq \mathcal{L}$ as the set of links encompassed by flow f. Based on this definition, for each link l, we define its flow set $\mathcal{F}(l) = \{f \in \mathcal{F} \mid l \in \mathcal{L}(f)\}$ as the set of flows that pass through it.

We now introduce the first set of constraints, *capacity constraint*. It states that for each link l, the total volume of its flow set $\mathcal{F}(l)$ cannot exceed its capacity c_l. Formally, let $\boldsymbol{A}$ be an $L \times F$ matrix, such that $A_{lf} = 1$ if flow f goes through the link l, i.e., $f \in \mathcal{F}(l)$. Otherwise, $A_{lf} = 0$.

$$\boldsymbol{A} \cdot \boldsymbol{x} \leq \boldsymbol{c} \tag{1}$$

Although this formulation fits into any physical topology,



(a) Star Topology (b) Overlay Tree

Fig. 1. P2P Streaming Illustration

in this paper, we base our solution on a widely-accepted assumption [13] that the bottleneck link of an end-to-end connection only happens at the uplink of the sending peer. As such, since the capacity constraint of other links never gets violated, they can be removed from the above inequality. This effectively reduces the link set $\mathcal{L}$ to only contain the uplinks of all peers and the server. As illustrated in Fig. 1(a), we term such a special topology as the *star topology*. Under this topology, the size of the link set equals to the size of the end host set $\mathcal{H}$.

The *relay constraint* states that the receiving rate of a peer cannot exceed the receiving rate of its parent. We illustrate this idea with the overlay tree shown in Fig. 1(b). In this picture, the relay constraint states the simple fact that the video quality received by h_3 and h_4 cannot be higher than the quality received by their parent h_1. Therefore, the corresponding rates of flows f_3 and f_4 cannot exceed the rate of flow f_1.

Since any peer can be the parent of any other peer, the total number of such parent-child pairs[1] is H^2. We formulate the relay constraint in a $H^2 \times F$ matrix $\boldsymbol{B}$ as follows:

$$B_{((h_k-1)H+h_i)\cdot f} = \begin{cases} -1 & \text{if } h_k = h(f) \text{ and } h_k \rightarrow h_i \\ 1 & \text{if } h_i = h(f) \text{ and } h_k \rightarrow h_i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$\boldsymbol{B}$ is a sparse matrix, where the $((h_k - 1)H + h_i)$th row will only be active if there is a flow from h_k to h_i. The relay constraint can be formalized as follows.

$$\boldsymbol{B} \cdot \boldsymbol{x} \leq \boldsymbol{0} \tag{3}$$

IV. Optimal Rate Allocation

In this section, we present our rate allocation solution whose goal is to minimize the aggregate rate distortion of all peers.

A. Performance Evaluation

For the evaluation of the video transmission system, the Mean-Squared-Error (MSE) metric is commonly used. However, representing simulation results in terms of Peak Singal-to-Noise Ratio (PSNR) is widely accepted in the multimedia coding and networking community. The PSNR used in this paper is the average PSNR of all the frames. It is defined as

$$PSNR = 10 \log_{10}(255^2/D) \tag{4}$$

where D represents the overall MSE of the entire encoded video sequence. In order to measure distortion as a function of rate x_f of a flow, we adopt a parametric rate distortion function proposed in [14]

$$D_f(x_f) = \frac{\theta^s}{x_f - x_0^s} + D_0 \tag{5}$$

where the variables $(\theta, x_0$ and $D_0)$ depend on the encoded sequence as well as on the percentage of INTRA coded macroblocks β. In our experiment, for each video sequence, these values are fitted with empirical data based on the trial encoding method described in [14].

B. Problem Formulation

Our goal is to minimize the aggregated distortion of the streaming video received by all peers. Given the rate distortion definition in (5), we present the following non-linear rate allocation optimization problem[2].

$$\begin{array}{rl} \textbf{min.} & \displaystyle\sum_{f \in \mathcal{F}} D_f(x_f) \\ \textbf{subject to} & (1) \textbf{ and } (3) \\ \textbf{over} & \boldsymbol{x} \in I_f \end{array} \tag{6}$$

[1]There are in fact several special cases which forbid parent-child pairs. For example, the server h_0 cannot be the child of any peer, also a peer cannot be the parent of itself. We do not specialize on these cases in our formulation for simplicity purposes. Nevertheless, the actual number of parent-child pair number remains in the order of H^2.

[2]We note that the objective function should exclude the rate distortion function for server h_0. For simplicity purpose, we ignore this detail in the rest of the paper. This can be easily achieved by assigning value 0 to the rate distortion function of h_0.

where (6) is a convex function of the allocated rate. The rate of each flow x_f is adapted within the range $I_f = [m_f, M_f]$.

By non-linear optimization theory, there exists a minimizing value of rate vector $\boldsymbol{x}$ for the above optimization problem. We consider the Lagrangian form of this problem:

$$\begin{aligned} & L(\boldsymbol{x},\, \boldsymbol{\mu}^{\alpha},\, \boldsymbol{\mu}^{\beta}) \\ = & \sum_{f \in \mathcal{F}(h)} D_f(x_f) + \boldsymbol{\mu}^{\alpha}(\boldsymbol{A} \cdot \boldsymbol{x} - \boldsymbol{c}) + \boldsymbol{\mu}^{\beta}(\boldsymbol{B} \cdot \boldsymbol{x}) \end{aligned} \tag{7}$$

where $\boldsymbol{\mu}^{\alpha} = (\mu_l^{\alpha},\, l \in \mathcal{L})$ and $\boldsymbol{\mu}^{\beta} = (\mu_k^{\beta},\, k = 1, \ldots, H^2)$ are vectors of Lagrangian multipliers. We define two new vectors $\boldsymbol{\lambda}^{\alpha} = (\lambda_f^{\alpha},\, f \in \mathcal{F})$ and $\boldsymbol{\lambda}^{\beta} = (\lambda_f^{\beta},\, f \in \mathcal{F})$ as follows

$$\lambda_f^{\alpha} = \sum_{l \in \mathcal{L}} \mu_l^{\alpha} A_{lf} = \sum_{l \in \mathcal{L}(f)} \mu_l^{\alpha} \tag{8}$$

$$\begin{aligned} \lambda_f^{\beta} &= \sum_{k=1}^{H^2} \mu_k^{\beta} B_{kf} \\ &= \sum_{h \to h(f)} \mu^{\beta}_{(h-1)H+h(f)} - \sum_{h(f) \to h} \mu^{\beta}_{(h(f)-1)H+h} \end{aligned} \tag{9}$$

Here μ_l^{α} is the link price. Consequently, λ_f^{α} is the sum of prices of all links in f's path, i.e., the *network price* mentioned in (1) that f has to pay. In the star topology, f only has to pay the price for the uplink of its sending peer. $\mu^{\beta}_{(h_i-1)H+h_k}$ is the *relay price* that peer h_k has to pay to its parent h_i for relaying the data. λ_f^{β} can be interpreted as *relay price* for f, which is the difference between the aggregated relay price of parent of $h(f)$ ($\sum_{h \to h(f)} \mu^{\beta}_{(h-1)H+h(f)}$) and the relay benefit h_f receives from all its children ($\sum_{h(f) \to h} \mu^{\beta}_{(h(f)-1)H+h}$).

C. Distributed Algorithm

Eq. (7) can be solved in a distributed fashion, following the gradient projection method adopted by many existing works. Thus, the optimal rate x_f for a flow f can be calculated as

$$x_f(\boldsymbol{\mu}^{\alpha},\, \boldsymbol{\mu}^{\beta}) = \left[x_0^s + \sqrt{\frac{\theta^s}{\lambda_f^{\alpha} + \lambda_f^{\beta}}} \right]_{m_f}^{M_f} \tag{10}$$

To this end, we present our distributed algorithm, which proceeds in rounds denoted as $t = 1, 2, \ldots$. Each round involves two steps. The first step is the price update, where price vectors are adjusted in opposition direction to the gradient $\nabla D(\boldsymbol{\mu}^{\alpha}, \boldsymbol{\mu}^{\beta})$.

$$\mu_l^{\alpha}(t+1) \;=\; [\mu_l^{\alpha}(t) + \gamma(\sum_{f \in \mathcal{F}(l)} x_f(t) - c_l)]^{+} \tag{11}$$

$$\begin{aligned} \mu^{\beta}_{(h-1)H+h(f)}(t+1) \;=& [\mu^{\beta}_{(h-1)H+h(f)}(t) + \gamma(x_f(t) \\ &- \sum_{f \in \mathcal{F}(h)} x_f(t))]^{+} \end{aligned} \tag{12}$$

The second step is the rate update, where the rate of flow f is adjusted corresponding to the price change. The rate x_f is calculated based on (10). This step requires the knowledge of network price λ_f^{α} and relay price λ_f^{β}, whose definitions can be found in (8) and (9) respectively.

D. Implementation

We discuss how to implement our algorithm in a distributed fashion, starting with the price update. As seen in (11), to update the price of link l, one needs to know its old price, and the rate of all flows going through it. Since our solution builds upon the star topology assumption, l must be a uplink of either the server or a peer, and all flows on this link must be generated from this peer as well. Obviously, the peer owning l is the best candidate as the bookkeeper of its price μ_l^{α}.

The relay price given in (12) applies to the parent-child pair $h \to h(f)$, where flow f directs from peer h to $h(f)$. To update it, one needs to know the old price, as well as peer $h(f)$'s receiving rate (x_f) and the receiving rate of its parent h ($\sum_{f \in \mathcal{F}(h)} x_f(t)$). Therefore, the best candidate to calculate and maintain the relay price is h. It can easily measure the rates of both flows since one of them enters h and the other exits from it.

To understand how to implement the rate update step for flow f, we again look at the parent-child pair $h \to h(f)$. As outlined in (10), calculating x_f requires the knowledge of network price λ_f^{α} and relay price λ_f^{β}. Given the definition of λ_f^{α} in (8) and the star topology assumption, we can easily see that λ_f^{α} is the price of the uplink of h, the sender of flow f. Since h is also the bookkeeper of its own uplink price, λ_f^{α} will involve no messaging overhead if we use the sender-based approach, i.e., we let h, the sender of flow f, to control its rate.

Given the definition of λ_f^{β} in (9), it is the difference between the relay price of parent-child pair $h \to h(f)$ and the summation of relay prices of all parent-child pairs originating from $h(f)$. Since the bookkeeper of a parent-child pair's relay price is the parent, we can see in order to calculate λ_f^{β}, h needs to receive a message from $h(f)$ reporting all relay prices managed by $h(f)$. Finally, other than λ_f^{α} and λ_f^{β}, h also needs to know several parameters (θ^s and x_0^s) specific to the video stream itself. The server can embed them into video packets at the beginning of the P2P streaming.

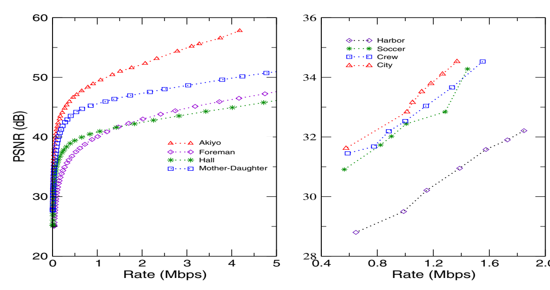


Fig. 2. (a)Video transcoded using x264 [15] (b) Approximated SVC based PSNR-Rate data [10]

In summary, the price update of our algorithm requires no messaging overhead since the bookkeeper can collect all necessary information locally to complete the update. The flow rate update process needs one message from the receiver of the flow to its sender, i.e., from child peer to parent peer. Since such a message can be blended into existing traffic between parent and children peers, such as heartbeat message or acknowledgement message in transmission protocol (e.g., TCP or RTCP), the messaging overhead can be further reduced by a great extent.

V. Simulations

In this section, we present our simulation results. We use our own simulation setup to perform the experiment. During the simulation, we the uplink bandwidth of each peer is randomly assigned between $0.6\,\mathrm{Mbps}$ and $2\,\mathrm{Mbps}$.

A. Video Adaptation

In our experiment, each peer performs transcoding by adjusting the quantization value of the video. Attempting to maximize the PSNR for all peers, the transcoding chooses the highest quantized rate that is less than the receiving rate achieved by our rate allocation solution. Formally, let x_{lf} be the optimal receiving rate for flow f calculated by our distributed algorithm, we denote $x_q = \{x_q \,|\, x_q < x_{q+1}, 1 \leq q \leq 51\}$ as the video encoding rate with quantization value q. The actual relay rate will then be x_f, where $x_f = \{x_q \,|\, x_q \leq x_{lf} < x_{q+1}\}$. We use the open-source software x264 [1] to encode videos with different quantization values. The test sequences used for our transcoding experiment are the ITU-T test sequences [15] *foreman*, *akiyo*, *hall* and *mother-daughter*, each having 300 frames with CIF resolution. Their PSNR-rates are given in Fig. 2(a).

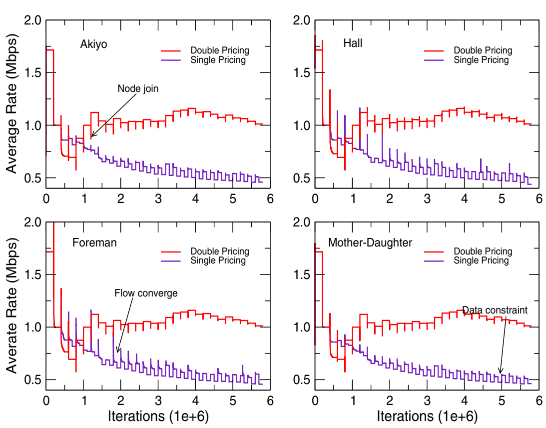


Fig. 3. Rate convergence as peers join the multicast tree

For scalable coding, we choose the H.264 scalable video coding extension. We acquire the approximated SVC based PSNR-rate data from [10] for video sequences *harbor*, *soccer*, *crew* and *city*. The PSNR-rates of these videos are given in Fig. 2(b).

B. Multicast Tree Construction

In our experiment, peers join the P2P streaming network one by one. The multicast tree is gradually constructed. When a new peer wishes to join the network, we use the spare capacity of each existing peer to determine a suitable parent. For each peer $h_l \in \mathcal{H}$ with link $l \in \mathcal{L}$, we define the spare coefficient as

$$s_h = [c_l - \sum_{f\in\mathcal{F}(l)} x_f]_0^{x_{f(h)}} \tag{13}$$

where $x_{f(h)}$ is the incoming flow rate of peer h. At the end of each rate update cycle, leaf peers send their spare coefficient to their parents. A parent then decides the best candidate with the highest coefficient value and sends this information to its own parent. The ID of the best candidate eventually propagates to the server. The server then provides a new peer with the parent information, so it can *join* the network as a child of that parent.

However, assigning the parent for a new peer based on the spare coefficient may lead to different overlay tree configuration for the single and double pricing solutions. In order ensure fairness when comparing these two solutions, we use the same tree configuration. We apply the above mentioned multicast tree construction criteria to construct the multicast tree for the single pricing solution. The double pricing solution then replicates this tree configuration.

When a peer leaves the multicast, all of its descendants simply perform the *join* operation independently to rejoin the multicast tree. During the simulation, we only perform join operations, since peer leaving a multicast implies independent join operation by all of its descendant peers in the tree.

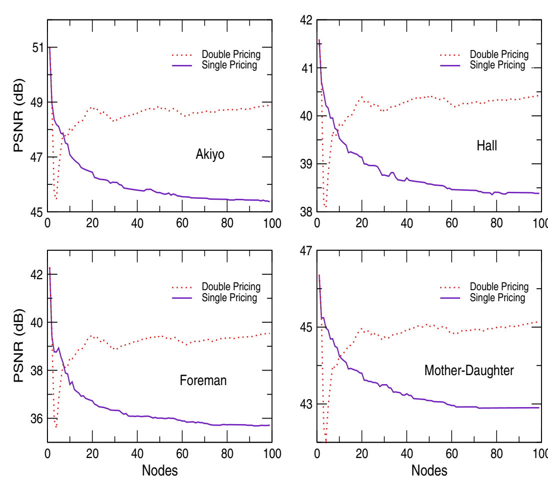


Fig. 4. Double pricing solution vs. single pricing solution for transcoded video

C. Simulation Results

We first test our solution's performance on rate convergence. We set up a P2P network of 30 peers with the first one as the server having the maximum uplink capacity of $2\,\mathrm{Mbps}$. After the first peer joins the network, in every $2\mathrm{e}+5$ iterations, a new peer joins the network. For the single pricing solution, every $1\mathrm{e}+5$ iterations after a peer joining event, we apply the data constraint. Following trial and error, we adjust the initial value of μ^{α} and μ^{β} to 0.5 and the initial rate to $1\,\mathrm{Mbps}$. Fig. 3 shows the rate convergence of the transcoded videos used in this paper. At the beginning, it takes more iterations for the rates to converge. This is due to the fact that initially the price value of λ^{α} and λ^{β} are assigned to 0. However, as more peers join the network, the price value stabilizes to the optimum point and it takes less number of iterations for the price to move from old optimum point to a new optimum point. Initially, the single pricing solution performs better when the network has only few peers. However, the double pricing solution consistently performs better when the network is large.

The step size also plays an important role in the number of iterations it takes for the rates to converge. An increase in step size from 0.0003 to 0.03 dramatically improves the number of iterations it takes for the rate to converge. With the step size of 0.03, the simulation converges to an optimal point in less than 2000 iteration. Detailed simulation results can be found in our technical report [16].

In Fig. 4, we show the average PSNR value for the transcoded videos as the number of peer grows from 0 to 100. The better performance of the double pricing solution is also confirmed when we run our simulation with the approximated H.264 SVC based video data as shown in Fig. 5. In this picture, the single pricing solution does not change after approximately 20 peers are added. This is due to the lack of data points at lower rates for the H.264 SVC based inputs [10]. To put this into prospective, for the transcoded video sequences, there are 51 data points available, where as, there are 7 data points available for the H.264 SVC based videos and only 4 of them are for data rates below $1.2\,\mathrm{Mbps}$. The average PSNR gain for the double pricing solution over all the transcoded videos in Fig. 4 is $2.03\,\mathrm{dB}$. For the SVC based data in Fig. 5, this gain is $1.81\,\mathrm{dB}$.

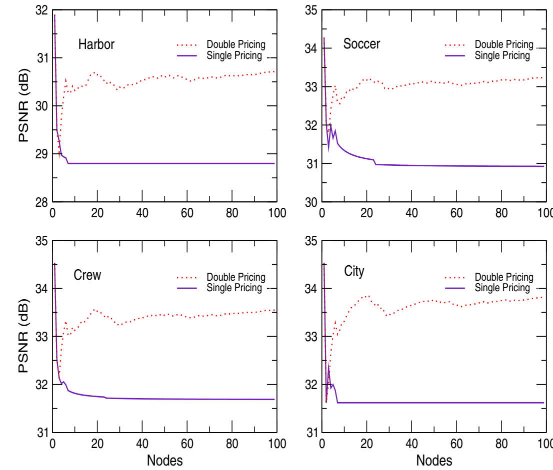


Fig. 5. Double pricing solution vs. single pricing solution for approximated video data based on SVC

VI. Conclusion

In this paper, we present an optimal rate allocation solution for P2P applications. We use non-linear optimization framework to minimize the aggregated distortion and thus maximize the overall PSNR among all peers in a P2P network. We consider the peer relaying price, unique in a P2P distribution scenario, along with the network price. Simulation shows that this double pricing solution improves the aggregate rate distortion for all peers in the network and provides a better video experience compared to the single pricing solution.

References

[1] "Videolan." [Online]. Available: http://www.videolan.org

[2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the h.264/avc standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 9, pp. 1204–1217, 2007.

[3] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, no. 1, 1997.

[4] F. Kelly, A. Maullo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *Journal of Operations Research Society*, vol. 49, no. 3, 1998.

[5] S. Low and D. Lapsley, "Optimization flow control, i: Basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, 1999.

[6] K. Kar, S. Sarkar, and L. Tassiulas, "Optimization based rate control for multirate multicast sessions," *In IEEE INFOCOM*, 2001.

[7] D. Bertsekas and J. Tsittsiklis, *Parallel and Distributed Computation*. Prentice-Hall, 1989.

[8] H. Han, S. Shakkottai, C. Hollot, R. Srikant, and D. Towsley, "Multi-path tcp: A joint congestion control and routing scheme to exploit path diversity in the internet," *IEEE/ACM Trans. Networking*, 2006.

[9] D. Jurca and P. Frossard, "Media flow rate allocation in multipath networks," *IEEE Transactions on Multimedia*, vol. 9, no. 6, 2007.

[10] X. Zhu, T. Schierl, T. Wiegand, and B. Girod, "Video multicast over wireless mesh networks with scalable video coding(svc)," *Visual Communications and Image Processing*, January 2008.

[11] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition:a mathematical theory of network architectures," in *Proc. IEEE*, no. 1, 2007.

[12] Y. Cui, Y. Xue, and K. Nahrstedt, "Optimal resource allocation in overlay multicast," *IEEE International Conference on Network Protocols*, 2003.

[13] M. Chen, M. Ponec, S. Sengupta, J. Li, and P. A. Chou, "Utility maximization in peer-to-peer systems," in *SIGMETRICS '08: Proceedings of the 2008 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. ACM, 2008, pp. 169–180.

[14] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 6, pp. 1012–1032, 2000.

[15] "Xiph.org." [Online]. Available: http://media.xiph.org/video/derf/

[16] T. Hossain, Y. Cui, and Y. Xue, "Minimizing rate distortion in peer-to-peer video streaming," 2008. [Online]. Available: http://vanets.vuse.vanderbilt.edu/publications/min-rate-report.pdf