OPTIMAL SERVER BANDWIDTH ALLOCATION AMONG MULTIPLE P2P MULTICAST LIVE VIDEO STREAMING SESSIONS

Aditya Mavlankar, Jeonghun Noh, and Bernd Girod

Information Systems Laboratory, Department of Electrical Engineering
Stanford University, Stanford, CA 94305, USA
Email: {maditya, jhnoh, bgirod}@stanford.edu

ABSTRACT

We consider a server that simultaneously streams multiple video channels. Each video channel is delivered to a set of receivers using peer-to-peer (P2P) live multicast. We propose a framework for allocating server bandwidth to minimize distortion across the peer population, across all channels. The optimization problem considers rate, distortion, the audience size, and peer-churn associated with each channel. Network simulations demonstrate reduction of mean distortion across the peer population due to the proposed server bandwidth allocation. We also highlight the general scope of the proposed framework which makes it applicable to scenarios beyond the one considered in this paper. Other optimization metrics rather than average distortion can also be accommodated in the proposed framework.

Index Terms— video streaming, peer-to-peer, multicast, multiple streaming sessions, multiple multicast sessions

1. INTRODUCTION

Peer-to-peer (P2P) multicasting is appealing as it requires much less server resources compared to a content delivery network (CDN) and is self-scaling as the resources of the network increase with the number of users. Numerous academic and commercial Internet P2P video streaming systems have been reported, for example [1–7]. Protocols proposed in the literature [8–11] often build one or more complementary multicast trees to push a video stream to interested peers.

As an example of a popular Internet P2P TV client, the SopCast [1] "What's on now" menu[1] is sorted into roughly 20 categories; a single popular category like soccer typically offers up to 80 live channels, also called live streams. Hence, tens to hundreds of live channels are likely to be disseminated from the same server, depending on the number of distributed servers deployed for the P2P service. We consider the scenario where a server simultaneously provides multiple video channels. Without loss of generality, we assume that a single tree is built per channel. The server can vary the number of its direct children per channel. For a given channel, the server has to expend more bandwidth to support more direct children, however, supporting more direct children reduces the impact of peer-churn on channel delivery.

We propose an optimization framework to allocate server bandwidth for supporting the optimal number of direct children for each tree. Although we focus on minimizing the total distortion across the peer population, other goals such as min-max fairness can also be accommodated in the proposed framework. Our approach takes into account the following properties that are characteristic of each video channel; the audience size, the churn resulting from peers joining and leaving, and the rate-distortion operating point associated with the compressed video.

The paper is structured as follows. We formulate the optimization problem in Section 2.1. We present the solution in Section 2.2. In Section 2.3, we deal with a special case where audience sizes associated with the channels could be different whereas rate, distortion and peer-churn characteristics are assumed to be identical across all channels. This case specifically explores the influence of audience sizes on rate allocation. The proposed problem is mostly unexplored in prior work on P2P video streaming. Section 3 mentions some related work, although it might not have directly tackled the considered issue. Additionally, Section 3 mentions several extensions and variations of the problem that fall within the scope of the general framework formulated in this paper. Ideas for future work naturally ensue from the discussion on possible extensions. Finally, Section 4 presents experimental results obtained from network simulations.

2. SERVER BANDWIDTH ALLOCATION

2.1. Optimization Problem

Let $\mathbf{P}$ denote the set of all peers. Let $\mathbf{S}$ denote the set of all video channels. The server bandwidth is denoted by R_S.

This work has been supported, in part, by a gift from Deutsche Telekom Laboratories.

[1]The website http://allp2ptv.org/ also offers menus for other popular Internet P2P TV clients like PPLive [3], PPStream [2], TVAnts [6], TVU-Player [12], and PPMate [13].

Assume that we are given the following quantities related to channel j, $\forall j \in \mathbf{S}$:

- m_j: Number of peers watching channel j.
- r_j: Bitrate of channel j.
- d_j: Distortion reduction associated with successful delivery of channel j.
- μ_j: Exponential distribution parameter for channel j. The contiguous period for which a peer joins channel j is an exponentially distributed random variable.

For tree j, the server allocates bandwidth to support up to α_j distinct direct children. Associated with channel j, random variable D_j represents total distortion reduction over all peers that subscribe channel j. The goal of the optimization problem, stated formally in (1) below, is to determine α_j, the number of direct children of the server for channel j, $\forall j \in \mathbf{S}$.

$$\max_{\alpha_j,\,\forall j\in\mathbf{S}} E\sum_{j\in\mathbf{S}} D_j\left(d_j, m_j, \mu_j, \alpha_j\right)$$
$$\text{subject to}$$
$$\sum_{j\in\mathbf{S}} \alpha_j r_j \leq R_S,$$
$$\alpha_j \in \mathbb{W},\, \alpha_j \leq m_j, \tag{1}$$

where, $\mathbb{W}$ is the set of whole numbers; $\mathbb{W} = \{0, 1, 2, \ldots\}$.

2.2. Solution

During the multicast session, the optimization problem can be solved repeatedly, each time for a short time-horizon T. The probability $\tilde{p_j}$ that a peer leaves channel j in this interval is given by $\tilde{p_j} = \left(1 - e^{-\mu_j T}\right)$.

The distortion reduction for channel j over all peers that subscribe channel j can be written as

$$D_j\left(d_j, m_j, \mu_j, \alpha_j\right) \quad \geq \quad \sum_{i\in\mathbf{P_j}} d_j\left(1 - Y_{ij}\right), \tag{2}$$

where

- $\mathbf{P_j}$ is the set of peers interested in channel j; $|\mathbf{P_j}| = m_j$. A peer is allowed to receive multiple channels.
- Y_{ij}: Indicator random variable which is 1 if peer i suffers disconnection due to an ancestor leaving the tree or if peer i itself leaves the channel in time-interval T; it indicates that peer i fails to contribute to the total distortion reduction.

The inequality sign in (2) accommodates the fact that peers leaving or suffering disconnection might receive the channel

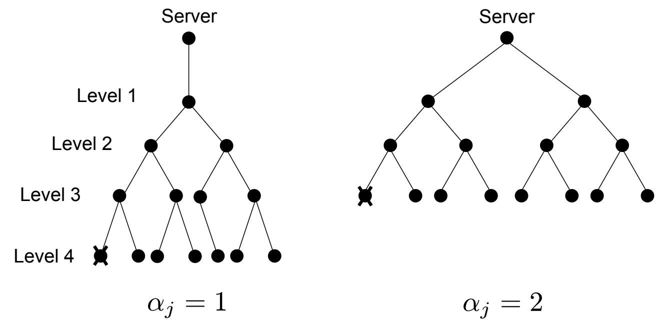


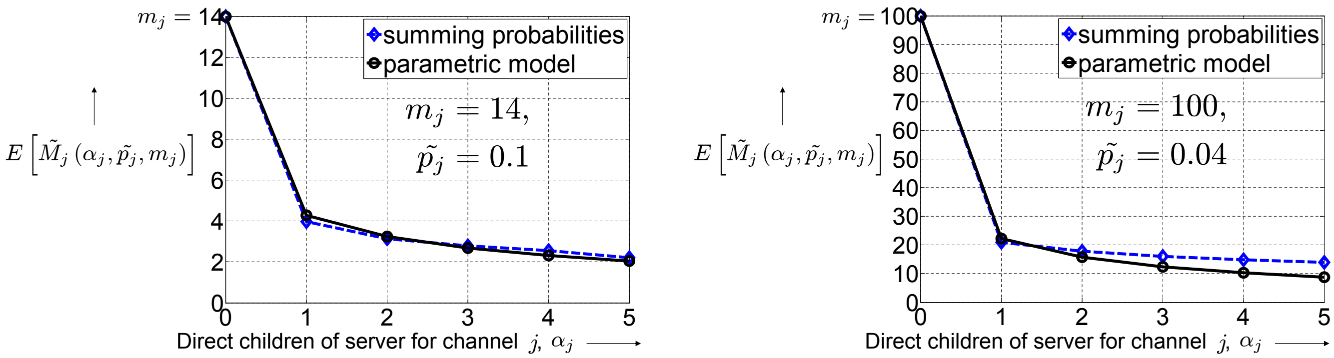
Fig. 1. Example with $m_j = 14$ peers. On the left, $\alpha_j = 1$ direct child is supported and on the right, $\alpha_j = 2$ direct children are supported. More direct children lower the average tree-height, however, the server has to spend more bandwidth. The solution to the optimization problem takes into account the impact of tree-structure on the received video quality per peer.

for part of the time-horizon T. Assuming that T is short[2] enough, we can write

$$\begin{aligned} E\left[D_j\left(d_j, m_j, \mu_j, \alpha_j\right)\right] &\approx m_j d_j - d_j \sum_{i\in\mathbf{P_j}} E\left(Y_{ij}\right) \\ &= m_j d_j - d_j \sum_{i\in\mathbf{P_j}} \Pr\left\{Y_{ij} = 1\right\}. \end{aligned}$$

Let random variable $\tilde{M_j}$ denote the number of peers that fail to contribute to the total distortion reduction associated with channel j; i.e., $\tilde{M_j} = \sum_{i\in\mathbf{P_j}} Y_{ij}$. In general, the lower the average tree-height for channel j the lower the value of $E\left(\tilde{M_j}\right)$. Given the structure of the tree for channel j, $E\left(\tilde{M_j}\right)$ can be obtained by summing the probabilities $\Pr\{Y_{ij} = 1\}$. If peer i is l hops away from the server, then $\Pr\{Y_{ij} = 1\} = 1 - \left(1 - \tilde{p_j}\right)^l$. These probabilities have been analyzed in detail, for example in [14], also for scenarios where forward error correction (FEC) is used for error-resilient P2P multicast video streaming. Consider the example shown in Fig. 1. We can compute $E\left(\tilde{M_j}\right)$ for increasing α_j assuming that all peer-nodes have the same out-degree. We observed that when all peer-nodes have the same out-degree, for a wide range of $(\tilde{p_j}, m_j)$, the expected number of peers that fail to contribute to the distortion reduction can be modeled as $E\left(\tilde{M_j}\right) = m_j e^{-\frac{a(\alpha_j)^c}{(\tilde{p_j})^b (m_j)^c}}$ with a suitably-fit constant triplet of model parameters (a, b, c). Fig. 2 shows an example of the model. In short, $E\left(\tilde{M_j}\right)$

[2]Notice that although the optimization problem is solved over short time-horizons, the server bandwidth allocation does not change unless some of the input parameters $\left(m_j, r_j, d_j, \mu_j\right), j \in \mathbf{S}$ change and also induce a change in the optimal solution α_k, $k \in \mathbf{S}$.



(a) Peer-node out-degree is assumed to be two; $(m_j, \tilde{p_j}) = (14, 0.1)$.

(b) Peer-node out-degree is assumed to be two; $(m_j, \tilde{p_j}) = (100, 0.04)$.

Fig. 2. The expected number of peers that fail to contribute to the total distortion reduction associated with channel j can be modeled as $E\left(\tilde{M}_j\right) = m_j e^{-\frac{a(\alpha_j)^c}{(\tilde{p_j})^b (m_j)^c}}$. The model parameters (a, b, c) used for both plots (a) and (b) are $(a = 0.33, b = 0.9, c = 0.3)$. We observed that for a wide range of $(m_j, \tilde{p_j})$, the approximation of $E\left(\tilde{M}_j\right)$ by the parametric model is close to $E\left(\tilde{M}_j\right) = \sum_{i \in \mathbf{P_j}} \Pr\{Y_{ij} = 1\}$, obtained by assuming a tree structure for the channel determined by α_j and a common peer-node out-degree.

can be computed either as $E\left(\tilde{M}_j\right) = \sum_{i \in \mathbf{P_j}} \Pr\{Y_{ij} = 1\}$ by assuming a tree structure for the channel determined by α_j or simply by approximating with the proposed parametric model. Notice that the fraction of peers failing to contribute to the total distortion reduction associated with channel j remains unchanged if $\tilde{p_j}$ and the ratio $\frac{\alpha_j}{m_j}$ are constant. Hence, the parametric model captures the fact that if the population of the sub-tree rooted at each direct child is kept constant, then the fraction $\frac{E(\tilde{M}_j)}{m_j}$ remains unchanged.

Using the parametric approximation, the expected distortion reduction by supporting α_j direct children (i.e., allocating rate $\alpha_j r_j$) for channel j is

$$E\left[D_j\left(d_j, m_j, \mu_j, \alpha_j\right)\right] = m_j d_j - m_j d_j e^{-\frac{a(\alpha_j)^c}{(\tilde{p_j})^b (m_j)^c}}. \quad (3)$$

The expected distortion reduction per unit rate by adding the α_j^{th} direct child, given that $\alpha_j - 1$ direct children have already been added, is given by

$$\Theta_j\left(\alpha_j\right) = \frac{E\left[D_j\left(d_j, m_j, \mu_j, \alpha_j\right)\right] - E\left[D_j\left(d_j, m_j, \mu_j, \alpha_j - 1\right)\right]}{r_j}. \quad (4)$$

Note that $\Theta_j\left(\alpha_j\right) > \Theta_j\left(\alpha_j + 1\right)$. Hence our optimization problem (1) can be cast as a classic knapsack problem. A greedy solution[3] can be obtained by sorting all $\Theta_j(1) \cdots \Theta_j(m_j)$, $\forall j \in \mathbf{S}$ and allocating rate till the bit-budget is exhausted. The parametric model introduced above dramatically reduces the computational burden for calculating the quantities Θ_j. This allows the algorithm to scale with the number of channels hosted by the server.

[3] not necessarily optimal

2.3. Special Case

Consider two video channels with the same rate, distortion and peer-churn characteristics; i.e., $r_1 = r_2 = r$, $d_1 = d_2 = d$, and $\tilde{p_1} = \tilde{p_2} = \tilde{p}$. The channel audience sizes, (m_1, m_2), need not be equal. The server bandwidth is given by $R_S = \alpha_S \times r$. The optimization problem then becomes,

$$\min_{\alpha_1,\, \alpha_2} E\left(\tilde{M}_1\right) + E\left(\tilde{M}_2\right) \quad \text{subject to} \quad \alpha_1 + \alpha_2 \leq \alpha_S. \quad (5)$$

For simplicity, we relax the constraint that (α_1, α_2) need to be whole numbers. The Lagrangian cost function can be written as

$$J = E\left(\tilde{M}_1\right) + E\left(\tilde{M}_2\right) + \lambda\left(\alpha_1 + \alpha_2 - \alpha_S\right).$$

With $\frac{\partial J}{\partial \alpha_1} = \frac{\partial J}{\partial \alpha_2} = 0$, we obtain

$$m_1 \times e^{-\frac{a(\alpha_1)^c}{(\tilde{p})^b (m_1)^c}} \times \frac{(\alpha_1)^{(c-1)}}{(m_1)^c} = m_2 \times e^{-\frac{a(\alpha_2)^c}{(\tilde{p})^b (m_2)^c}} \times \frac{(\alpha_2)^{(c-1)}}{(m_2)^c}. \quad (6)$$

The solution $(\alpha_1, \alpha_2) = \left(\alpha_S \frac{m_1}{m_1 + m_2}, \alpha_S \frac{m_2}{m_1 + m_2}\right)$ satisfies (6) and other Karush-Kuhn-Tucker (KKT) [15] conditions.

Table 1. Simulation scenarios and result of rate allocation algorithm from Section 2.2.

Scenario number	Number of channels	Video sequences	Bitrates [kbps]	Distortion reduction [reduction in MSE]	Audience sizes	Average peer lifetimes [sec]	Rate allocation as per Section 2.2	Server capacity, R_S [Mbps]
1.	2	1: *Foreman*	$r_1 = 1280$	$d_1 = 1490$	$m_1 = 95$	$\frac{1}{\mu_1} = 90$	$\alpha_1 = 9$	12.8
		2: *Foreman*	$r_2 = 1280$	$d_2 = 1490$	$m_2 = 5$	$\frac{1}{\mu_2} = 90$	$\alpha_2 = 1$	
2.	2	1: *Foreman*	$r_1 = 1280$	$d_1 = 1490$	$m_1 = 50$	$\frac{1}{\mu_1} = 80$	$\alpha_1 = 6$	10.3
		2: *Foreman*	$r_2 = 1280$	$d_2 = 1490$	$m_2 = 50$	$\frac{1}{\mu_2} = 200$	$\alpha_2 = 2$	
3.	2	1: *Foreman*	$r_1 = 1280$	$d_1 = 1490$	$m_1 = 20$	$\frac{1}{\mu_1} = 90$	$\alpha_1 = 3$	6.5
		2: *Mother & Daughter*	$r_2 = 425$	$d_2 = 198$	$m_2 = 80$	$\frac{1}{\mu_2} = 90$	$\alpha_2 = 6$	
4.	3	1: *Foreman*	$r_1 = 1280$	$d_1 = 1490$	$m_1 = 30$	$\frac{1}{\mu_1} = 90$	$\alpha_1 = 4$	7.7
		2: *News*	$r_2 = 628$	$d_2 = 200$	$m_2 = 30$	$\frac{1}{\mu_2} = 90$	$\alpha_2 = 2$	
		3: *Container*	$r_3 = 640$	$d_3 = 400$	$m_3 = 30$	$\frac{1}{\mu_3} = 90$	$\alpha_3 = 2$	

Since the objective function is convex in (α_1, α_2), this is the optimal rate allocation. This special case can be easily extended to multiple channels and provides the following insight; for video channels with similar rate, distortion, and peer-churn characteristics, the amount of server bandwidth allocated to them should be proportional to their audience sizes. In practice, however, each α_j has to be a whole number. As proposed in Section 2.2, the greedy solution can be employed. Unlike the special case considered in this section, the greedy solution can also account for differences in rate, distortion and peer-churn characteristics.

3. EXTENSIONS AND RELATED WORK

Various modifications fall within the scope of the general framework formulated above: min-max fairness, distortion-oblivious maximization of expected number of peers that receive a channel[4], adaptation of model of $E\left(\tilde{M}_j\right)$ to better suit the distributed P2P protocol, e.g., speed of rejoin procedure compared to the time-horizon T. It is a common observation that rejoins are faster on average with bigger population sizes, for example [16]. The loss model should account for this.

The proposed framework can also incorporate multiple complementary trees per channel with an optional constraint of forcing equal number of direct children for each tree of the *same* channel. The additional constraint can be dropped if the data associated with the trees possess unequal rate-distortion importance. Although we focus on multicast trees, the approach is also applicable to mesh-based topologies.

[4]This can be easily accomplished by letting $d_j = d, \forall j \in \mathbf{S}$ (d is an arbitrary constant) in the solution presented in Section 2.2 and corresponds to minimizing the number of frame freezes overall.

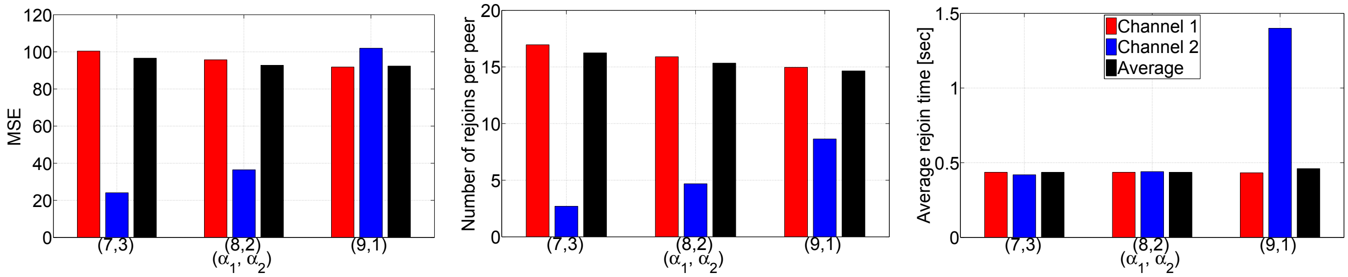
A recent work dealing with issues of co-existence of multiple streaming overlays is [17]. In their work, a peer that is part of multiple overlays allocates its uplink capacity among the overlays according to the result of an auction game. Their framework can incorporate priorities for different overlays, however, they do not propose any server-side algorithm for assigning priorities. Two other pieces of work [18, 19] propose that peers in different overlays help each other by relaying media belonging to other overlays.

If video transcoding is possible on-the-fly, then it might provide more degrees of freedom for optimal server bandwidth allocation among multiple streams. Video transcoding and online rate-distortion tradeoff estimation has been studied in [20], however, in a different context; delivery of multiple video streams over a common wireless channel.

In the context of P2P video streaming with interactive region-of-interest (IROI) [21, 22], we have applied the proposed framework for rate allocation among picture slices. In this novel application, a peer can watch an arbitrary region of the scene with an arbitrary zoom factor. Video streams corresponding to slices of the scene are delivered over multicast trees. Thus, each slice can be considered as a separate video channel and a peer subscribes multiple channels according to its region-of-interest (ROI). The server hosts hundreds of slices in a typical P2P IROI video session and the low computational complexity of the proposed algorithm is desirable for computing the rate allocation among hundreds of channels.

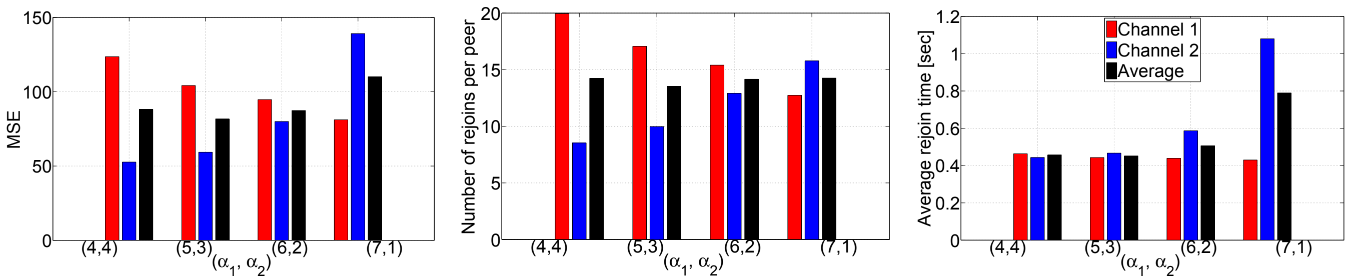
4. EXPERIMENTAL RESULTS

We implemented the Stanford Peer-to-Peer Multicast (SPPM) [23–27] protocol within the NS-2 network simulator [28]. We use the GT-ITM topology generator [29] to create a 4-ary tree physical topology for the backbone network. Peers are placed



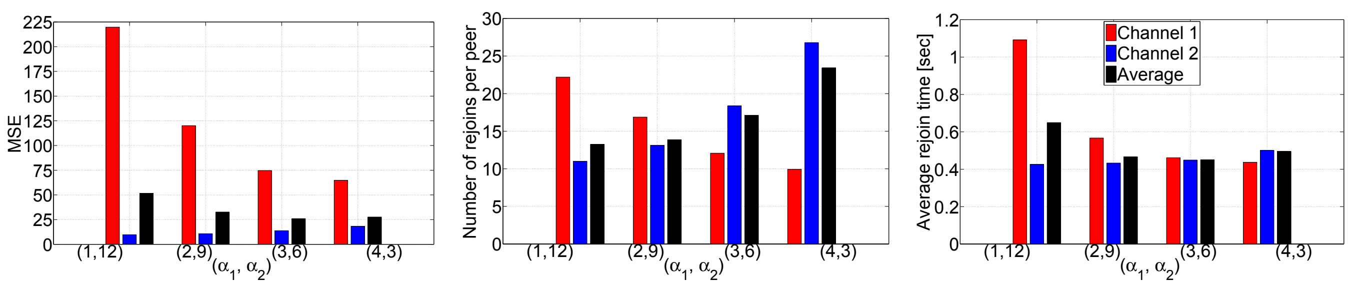
(a) Mean squared error (MSE). (b) Number of rejoins per peer. (c) Average rejoin time.

Fig. 3. Scenario 1: $(m_1, m_2) = (95, 5)$ peers, $(r_1, r_2) = (1280, 1280)$ kbps, $(d_1, d_2) = (1490, 1490)$, $\left(\frac{1}{\mu_1}, \frac{1}{\mu_2}\right) = (90, 90)$ seconds. Optimal allocation suggested by proposed algorithm is $(\alpha_1, \alpha_2) = (9, 1)$. It results in the smallest overall MSE experimentally. Rejoin is triggered by a peer if it detects that its immediate parent has left the multicast session or if video data are not received for more than one second.



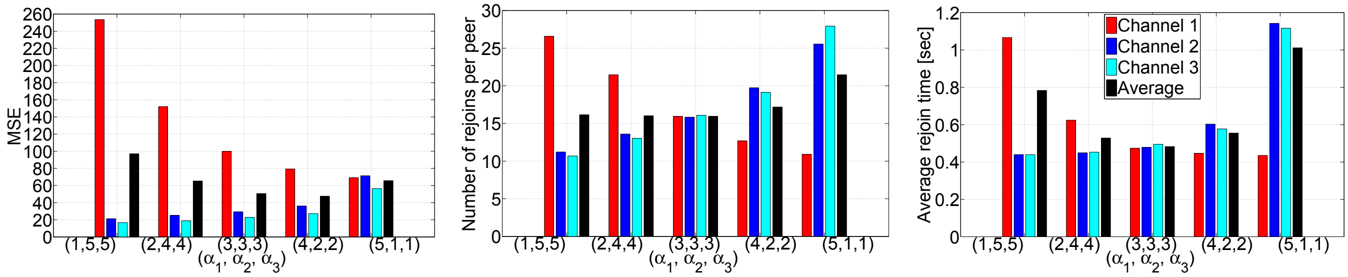
(a) Mean squared error (MSE). (b) Number of rejoins per peer. (c) Average rejoin time.

Fig. 4. Scenario 2: $(m_1, m_2) = (50, 50)$ peers, $(r_1, r_2) = (1280, 1280)$ kbps, $(d_1, d_2) = (1490, 1490)$, $\left(\frac{1}{\mu_1}, \frac{1}{\mu_2}\right) = (80, 200)$ seconds. Optimal allocation suggested by proposed algorithm is $(\alpha_1, \alpha_2) = (6, 2)$. Among the tested allocations, the proposed allocation results in the smallest disparity of MSE as well as the number of rejoins per peer between the two channels. Rejoin is triggered by a peer if it detects that its immediate parent has left the multicast session or if video data are not received for more than one second.



(a) Mean squared error (MSE). (b) Number of rejoins per peer. (c) Average rejoin time.

Fig. 5. Scenario 3: $(m_1, m_2) = (20, 80)$ peers, $(r_1, r_2) = (1280, 425)$ kbps, $(d_1, d_2) = (1490, 198)$, $\left(\frac{1}{\mu_1}, \frac{1}{\mu_2}\right) = (90, 90)$ seconds. Optimal allocation suggested by proposed algorithm is $(\alpha_1, \alpha_2) = (3, 6)$. It results in the smallest overall MSE experimentally. Although the allocation $(\alpha_1, \alpha_2) = (2, 9)$ allocates number of direct children roughly proportional to the audience sizes, it results in higher MSE than the suggested allocation. We conjecture this is because the two channels have different rate-distortion operating points. Rejoin is triggered by a peer if it detects that its immediate parent has left the multicast session or if video data are not received for more than one second.



(a) Mean squared error (MSE). (b) Number of rejoins per peer. (c) Average rejoin time.

Fig. 6. Scenario 4: $(m_1, m_2, m_3) = (30, 30, 30)$ peers, $(r_1, r_2, r_3) = (1280, 628, 640)$ kbps, $(d_1, d_2, d_3) = (1490, 200, 400)$, $\left(\frac{1}{\mu_1}, \frac{1}{\mu_2}, \frac{1}{\mu_3}\right) = (90, 90, 90)$ seconds. Optimal allocation suggested by proposed algorithm is $(\alpha_1, \alpha_2, \alpha_3) = (4, 2, 2)$. It results in the smallest overall MSE experimentally. Although the allocation $(\alpha_1, \alpha_2, \alpha_3) = (3, 3, 3)$ allocates number of direct children proportional to the audience sizes, it results in higher MSE than the suggested allocation. We conjecture this is because the three channels have different rate-distortion operating points. Rejoin is triggered by a peer if it detects that its immediate parent has left the multicast session or if video data are not received for more than one second.

on the randomly chosen edge nodes of the backbone network. The backbone links are sufficiently provisioned with high capacity. The propagation delay of the network links is set to 5 ms. A single multicast tree is built per channel and each peer can support up to two direct children. Peer-churn is simulated by drawing peer lifetimes randomly from the channel-specific exponential distribution. A peer watches one channel at a time.

We use four video sequences for our experiments. All sequences are 30 frames/sec, have CIF resolution, and are encoded using H.264/AVC with an intraframe period of 16 frames and three consecutive B frames between anchor frames. The PSNR @ bitrate for the *Foreman*, *Mother & Daughter*, *News*, and *Container* sequences are 40.5 dB @ 1280 kbps, 42.9 dB @ 425 kbps, 42.6 dB @ 628 kbps, and 40.6 dB @ 640 kbps, respectively. Each video sequence is 10 seconds long and is looped for longer network simulations. In order to simplify the video quality assessment, we assume that a packet loss associated with a frame causes the loss of the whole frame. A video frame is not decodable, and hence considered to be lost, if either this frame or any other frame that this frame depends on is not received by its display deadline. In case of a frame loss, the previous correctly decoded frame is displayed. We assume a pre-roll delay of 3 seconds before video playback starts.

We simulate four scenarios outlined in Table 1. The rate allocation according to the algorithm from Section 2.2 is shown in Table 1. The time-horizon assumed for computing the rate allocation was $T = 3$ seconds. The average distortion reduction[5] associated with successful delivery of each sequence is computed as the average distortion increase when the display is frozen for the entire time-horizon instead of normal playback. The frozen frame is assumed to be the last frame from the previous time-horizon. In general, the higher the motion in the video sequence the higher the distortion reduction associated with successful delivery of the channel. We vary the rate allocation for our experiments in each scenario and perform multiple simulation trials for each tested rate allocation. Each simulation trial is 900 seconds long and the results presented next are obtained by averaging over 10 trials. We present the mean squared error (MSE) per channel as well as over the entire peer population. We also present rejoin statistics. Rejoin is triggered by a peer if it detects that its immediate parent has left the multicast session or if video data are not received for more than one second.

Scenario 1 corresponds to the special case from Section 2.3 in which the rate, distortion and peer-churn characteristics of the two channels are identical. If the whole number constraint on (α_1, α_2) is relaxed then the optimal rate allocation is $(\alpha_1, \alpha_2) = (9.5, 0.5)$. The greedy solution is $(\alpha_1, \alpha_2) = (9, 1)$. Fig. 3 shows experimental results for this scenario. The allocation suggested by the proposed algorithm results in the smallest overall MSE among all tested rate allocations.

In Scenario 2, we vary the peer-churn characteristics of the two channels and keep the other characteristics of the two channels identical. Fig. 4 shows experimental results for Scenario 2. The allocation suggested by the proposed algorithm is $(\alpha_1, \alpha_2) = (6, 2)$. In our experiments we found that $(\alpha_1, \alpha_2) = (5, 3)$ results in slightly lower MSE than $(\alpha_1, \alpha_2) = (6, 2)$. Nevertheless, $(\alpha_1, \alpha_2) = (6, 2)$ results in the smallest disparity of MSE as well as the number of rejoins per peer between the two channels.

Scenario 3 represents the case where the two channels are

[5]For our experiments, we compute the average over the entire video sequence such that d_j does not vary with time. Alternatively, a moving average and hence changing d_j could be incorporated while solving the optimization problem over successive time-horizons.

showing different video sequences. However, the peer-churn characteristics are identical for the two channels. Since the two channels have different rate-distortion operating points, the optimal solution need not allocate number of direct children proportional to the audience sizes. Fig. 5 shows experimental results for Scenario 3. The bitrate of Channel 1 is roughly three times the bitrate of Channel 2 and hence we did not test for allocations like $(\alpha_1, \alpha_2) = (2, 8)$ that leave part of the server bandwidth unutilized. Among the tested allocations, $(\alpha_1, \alpha_2) = (2, 9)$ roughly allocates number of direct children proportional to the audience sizes. However, according to both the proposed algorithm and the experiments, $(\alpha_1, \alpha_2) = (3, 6)$ results in the smallest MSE.

Finally, Fig. 6 shows experimental results for Scenario 4. In this Scenario, the three channels are showing three different video sequences. Peer-churn characteristics are identical for all three channels. Again, the proposed allocation need not allocate number of direct children proportional to the audience sizes. This is because the three channels have different rate-distortion operating points. According to both the proposed algorithm and the experiments, $(\alpha_1, \alpha_2, \alpha_3) = (4, 2, 2)$ results in the smallest MSE.

5. CONCLUSIONS

We propose an optimization framework to allocate the server bandwidth among multiple P2P multicast video channels. The generality of the framework makes it applicable to scenarios beyond the one considered in this paper; examples include P2P multicast of slices for video streaming with IROI, mesh-based instead of tree-based topology, multiple complementary multicast trees instead of single tree per channel, multiple complementary multicast trees carrying unequally important data, e.g., FEC and/or layered coding, etc. The framework is amenable to a change of metric as well as replacement of the model of number of peers suffering disconnection. The use of a parametric model for approximating the number of peers suffering disconnection lowers the computational complexity. It allows the algorithm to scale with the number of channels. This is desirable since popular P2P systems may host hundreds of channels simultaneously.

The proposed allocation most importantly considers the audience sizes and rate-distortion operating points associated with different channels. Intuitively, the gain due to churn-aware rate allocation is lower when the P2P protocol is more robust against churn. However, this robustness often comes at the cost of longer playout delay and redundancy in received data [30]. In our experiments, we focus on the push approach which enables low playout delay and low redundancy. We observe reduction of the mean distortion across the peer population due to the proposed rate allocation.

6. REFERENCES

[1] SopCast. [Online]. Available: http://www.sopcast.org/

[2] PPStream. [Online]. Available: http://www.ppstream.com/

[3] PPLive. [Online]. Available: http://www.pplive.com/

[4] Coolstreaming. [Online]. Available: http://www.coolstreaming.us/

[5] GridMedia. [Online]. Available: http://www.gridmedia.com.cn/

[6] TVAnts. [Online]. Available: http://www.tvants.com/

[7] VVSky. [Online]. Available: http://www.vvsky.com.cn/

[8] Y.-H. Chu, S. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 8, pp. 1456–1471, Oct. 2002.

[9] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," in *Proc. ACM SIGCOMM*, Pittsburgh, PA, USA, Aug. 2002.

[10] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking," in *Proc. ACM NOSSDAV*, Miami, FL, USA, May 2002, pp. 177–186.

[11] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: High-bandwidth content distribution in a cooperative environment," in *Proc. of IPTPS*, Berkeley, CA, USA, Feb. 2003.

[12] TVUPlayer. [Online]. Available: http://www.tvunetworks.com/

[13] PPMate. [Online]. Available: http://ppmate-nettv.en.softonic.com/

[14] G. Dan, I. Chatzidrossos, V. Fodor, and G. Karlsson, "On the performance of error-resilient end-point-based multicast streaming," in *Proc. IEEE 14th Intl. Workshop on Quality of Service (IWQoS)*, New Haven, CT, USA, Jun. 2006, pp. 160–168.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[16] R. Kumar, Y. Liu, and K. Ross, "Stochastic fluid theory for P2P streaming systems," in *Proc. 26th IEEE Intl. Conf. on Computer Communications (INFOCOM'07)*, Anchorage, Alaska, May 2007, pp. 919–927.

[17] C. Wu and B. Li, "Strategies of conflict in coexisting streaming overlays," in *Proc. 26th IEEE Intl. Conf. on Computer Communications (INFOCOM'07)*, Anchorage, Alaska, May 2007, pp. 481–489.

[18] X. Liao, H. Jin, Y. Liu, L. M. Ni, and D. Deng, "Anysee: Peer-to-peer live streaming," in *Proc. 25th IEEE Intl. Conf. on Computer Communications (INFOCOM'06)*, Barcelona, Spain, Apr. 2006, pp. 1–10.

[19] G. Tan and S. A. Jarvis, "Inter-overlay cooperation in high-bandwidth overlay multicast," in *Proc. Intl. Conf. on Parallel Processing (ICPP'06)*, Columbus, OH, USA, Aug. 2006, pp. 417–424.

[20] M. Kalman, "Low-latency delivery of multiple video

streams over a wireless channel," Ph.D. dissertation, Stanford University, Stanford, CA, USA, Sep. 2006.

[21] A. Mavlankar, J. Noh, P. Baccichet, and B. Girod, "Peer-to-peer multicast live video streaming with interactive virtual pan/tilt/zoom functionality," in *Proc. IEEE Intl. Conf. on Image Processing (ICIP'08)*, San Diego, CA, USA, Oct. 2008, pp. 2296–2299.

[22] ——, "Optimal server bandwidth allocation for streaming multiple streams via P2P multicast," in *Proc. IEEE 10th Workshop on Multimedia Signal Processing (MMSP'08)*, Cairns, Australia, Oct. 2008.

[23] E. Setton, J. Noh, and B. Girod, "Rate-distortion optimized video peer-to-peer multicast streaming," in *Proc. Workshop on Advances in Peer-to-Peer Multimedia Streaming at ACM Multimedia*, Singapore, Nov. 2005, pp. 39–48, invited paper.

[24] ——, "Low latency video streaming over peer-to-peer networks," in *Proc. IEEE Intl. Conf. on Multimedia and Expo (ICME'06)*, Toronto, Canada, Jul. 2006, pp. 569–572.

[25] ——, "Congestion-distortion optimized peer-to-peer video streaming," in *Proc. IEEE Intl. Conf. on Image Processing (ICIP'06)*, Atlanta, GA, USA, Oct. 2006, pp. 721–724.

[26] P. Baccichet, J. Noh, E. Setton, and B. Girod, "Content-aware P2P video streaming with low latency," in *Proc. IEEE Intl. Conf. on Multimedia and Expo (ICME'07)*, Beijing, China, Jul. 2007, pp. 400–403.

[27] E. Setton, P. Baccichet, and B. Girod, "Peer-to-peer live multicast: A video perspective," *Proc. IEEE*, vol. 96, no. 1, pp. 25–38, Jan. 2008.

[28] The Network Simulator - ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns

[29] E. Zegura, K. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. 15th IEEE Intl. Conf. on Computer Communications (INFOCOM'96)*, vol. 2, San Francisco, CA, USA, Mar. 1996, pp. 594–602.

[30] A. Mavlankar, P. Baccichet, B. Girod, S. Agarwal, and J. Singh, "Video quality assessment and comparative evaluation of peer-to-peer video streaming systems," in *Proc. IEEE Intl. Conf. on Multimedia and Expo (ICME'08)*, Hanover, Germany, Jun. 2008, pp. 645–648.