Mesh-based Peer-to-Peer Layered Video Streaming With Taxation

Hao Hu
ECE, Polytechnic Institute of NYU
Brooklyn, NY 11201
hhu01@students.poly.edu

Yang Guo
Research and Innovation, Technicolor
Princeton, NJ 08540
Yang.Guo@technicolor.com

Yong Liu
ECE, Polytechnic Institute of NYU
Brooklyn, NY 11201
yongliu@poly.edu

ABSTRACT

Recent advance in scalable video coding (SVC) makes it possible for users to receive the same video with different qualities. To adopt SVC in P2P streaming, two key design questions need to be answered: 1) layer subscription: how many layers each peer should receive? 2) layer scheduling: how to deliver to peers the layers they subscribed? From the system point of view, the most efficient solution is to maximize the aggregate video quality on all peers, i.e., the social welfare. From individual peer point of view, the solution should be fair. Fairness in P2P streaming should additionally take into account peer contributions to make the solution incentive-compatible. In this paper, we show that taxation mechanisms can be devised to strike the right balance between social welfare and individual peers' welfare. We develop practical taxation-based P2P layered streaming designs, including layer subscription strategy, chunk scheduling policy, and mesh topology adaptation. Extensive trace-driven simulations show that the proposed designs can effectively drive layered P2P streaming systems to converge to the desired operating points in a distributed fashion.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Applications; C.2.4 [**Distributed Systems**]: Distributed Applications

General Terms

Algorithm, Design, Performance

Keywords

Peer-to-Peer, Layered Video, Fairness, Incentive, Taxation

1. INTRODUCTION

P2P live video streaming has achieved tremendous success and emerged as a costefficient IPTV solution on the Internet. The continuous success of P2P computing hinges on the underlying principle that participants shall contribute their resources (in terms of bandwidth, storage space, or computational power) while enjoying the service. Most existing P2P streaming systems assume the cooperation of peers and deliver the same video quality to all peers.

NOSSDAV'10, June 2–4, 2010, Amsterdam, The Netherlands.

With scalable video coding (SVC), it is possible for users to receive the same video with different qualities. SVC encodes video into correlated layers. The base layer can be independently decoded, while higher layers are decodable only if layers beneath have been decoded. The video quality perceived by a user increases as the number of decoded layers increases. While multiple-layer coded video incurs coding overhead, recent advance in SVC coding has brought down the overhead to 10% [16]. It is now practical to adopt SVC into P2P video streaming to extend its design space.

The adoption of SVC into P2P streaming faces two key design questions: 1) *layer subscription:* how many layers each peer should receive; and 2) *layer scheduling:* how to deliver to peers the layers they subscribed. From the system point of view, the most *efficient* solution is to maximize the aggregate video quality perceived by all peers, i.e, to optimize the *social welfare*. From individual peer point of view, the solution should be *fair*. However, in P2P streaming, due to the dual server-consumer role of individual peers, the notion of fairness is much more subtle than that in traditional server-client systems, where clients are only considered as resource consumers. A solution allocating the same video quality to all peers regardless of their contributions would not be considered as fair, and therefore would not provide *incentives* for peers to contribute.

Unbalanced setting of efficiency, fairness and incentive could incur serious problem in reality. For example, if an Ethernet user with uplink capacity of 2,000 Kbps and a DSL user with uplink capacity of 200 Kbps both receive video at rate of 500 Kbps, why would the Ethernet user contribute more than 200 Kbps? If we assume all peers are strategic, then the bandwidth contributed by peers will decrease and everyone will get poor video quality. On the other hand, if the DSL user uploads video at its full capacity, he may deserve some "help" from Ethernet users to download video at a rate higher than 200 Kbps. A good layered P2P streaming solution has to strike the right balance between efficiency, fairness and incentive.

Taxation based incentive mechanism [10, 14] offers a flexible framework that allows such tradeoffs. Let u_d be the upload bandwidth contributed by user d. Under a tax rate $0 \leq t \leq 1$, the target received video rate of user d is $r_d = (1-t)u_d + \frac{t}{N}\sum_{i=1}^{N} u_i$, where N is the total number of peers in the system. The received rate consists of two parts: a fraction of its own contribution, and a fair share from the pool of taxed bandwidth. As shown in Fig. 1, the tax rate t adjusts the balance between individual peers' welfare and the social welfare. As t approaches zero, the received video rate approach the contributed rate, mimicking the 'tit-for-tat' strategy. As t approaches one, the received video rate is the same for all peers, thus achieve the social optimum. We define $(1-t)U_d$ as peer d's entitled rate, and then map this rate to layers. All layers other than the entitled layers are denoted as excess layers.

In this paper, we develop analytical model to study the tradeoff between efficiency, fairness and incentive in taxation based layered

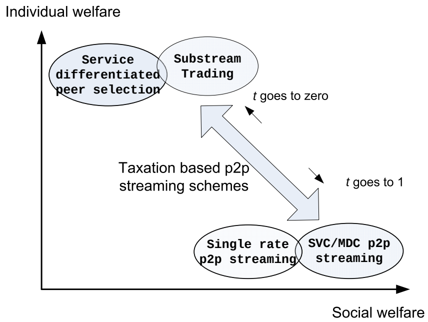


Figure 1: Tax rate t controls the balance between social welfare and individual welfare.

P2P streaming. We show that taxation mechanisms can be devised to strike the right balance between social welfare and individual welfare. We develop practical taxation-based P2P layered streaming designs, including layer subscription strategy, chunk scheduling policy, and mesh topology adaptation. Extensive trace-driven simulations further demonstrate that the proposed designs can effectively drive the layered P2P streaming systems to converge to the desired operating points in a distributed fashion.

1.1 Related Work

Layered coding, such as SVC or MDC, is particularly effective in handling heterogeneous users. Layered coding is applied to P2P streaming to improve the social welfare over the traditional single layer P2P streaming [12, 21, 18]. In terms of achieving individual fairness/welfare, Cohen advocated a tit-for-tat algorithm in the seminar paper[4]. [8] proposes a score-based incentive mechanism for P2P live streaming. [6] proposes a service differentiated peer selection algorithm that gives peers with higher contributions more flexibility in choosing neighbors, thus obtain better viewing quality. Some measurement and theoretical studies on P2P incentives can be found in [5, 19]. We use taxation mechanisms to strike the right balance between social welfare and individual welfare. The work in [10, 14] also employs taxation-based incentive mechanism. A video is encoded into substreams using multiple descriptions coding (MDC). Individual substreams are distributed along *trees* formed by peers. The number of trees joined by a peer is dynamically adjusted to reflect the entitled video quality determined by the taxation policy. However, tree-based streaming is more vulnerable to peer churn than mesh-based streaming [20, 13]. In this paper, we study mesh-based SVC P2P streaming with taxation.

2. SYSTEM MODEL

In this section, we develop analytical model to study layered P2P streaming with taxation.

2.1 P2P Layered Streaming with Taxation

We consider a SVC system where the source server encodes a video stream into L layers with nested dependency. Layer l can be decoded if all the layers below l are received. A peer can subscribe up to k, $k \le L$, layers. The server multicasts each layer to all peers subscribed to it. There are L simultaneous multicast sessions, one for each layer, in the P2P overlay network. As commonly assumed for P2P overlay networks, we focus on the case that peer uplinks are the only bandwidth bottlenecks. Network coding has been shown to achieve the maximum multicast rate in general network topology [1, 15]. We allow the server and peers apply network coding to video blocks to reach the maximum multicast rate allowed by peer upload capacities and the peering topology.

Table 1: Notations

Notation	Description
V	set of nodes in the system
E	set of overlay links
S	video source server
$R = V \backslash S$	receiving peers
L	number of layers
r^l	rate of layer l
$\vec{x_d} = \{x_d^l\}$	layers received by peer d
$g_{ij}^{l,d}$	information flow of layer l on link $\langle i, j\rangle$ to peer d
f_{ij}^l	bandwidth needed for layer l on link $\langle i, j\rangle$
U_d	peer d's uplink capacity
$F_d(\vec{x_d})$	utility function of peer d
t	tax rate

Let a directed graph $\mathcal{G} = (V, E)$ be the overlay topology of the P2P streaming system under study. Let S be the video source server, and $R = V \backslash S$ be the set of peers interested in receiving the video. Let $\vec{x_d} = (x_d^1, x_d^2, \cdots, x_d^L)$ represent the set of layers received by peer d: x_d^l equals to 1 if peer d received layer l, 0 otherwise. The video rate for layer l is r^l. To model network coding, we introduce $g_{ij}^{l,d}$ to denote the *information flow* of layer l on link $\langle i, j\rangle \in E$ to destination peer d. For a given peer d and layer l, $\{g_{ij}^{l,d}, \langle i, j\rangle \in E\}$ form a legitimate flow with rate r^l from the source S to d and satisfy the flow conservation on all nodes in the network. Denote by $f_{ij}^l \triangleq \max_d g_{ij}^{l,d}$ the maximum information flow on $\langle i, j\rangle$ for all receivers of layer l. The multicast session for layer l is supportable if and only if a bandwidth of f_{ij}^l is allocated to layer l on link $\langle i, j\rangle$.

We are interested in seeking the optimal P2P streaming solution to maximize the aggregate video experience of all peers while meeting the taxation constraint. By adopting the PSNR-Rate model, [2], we quantify a user's video experience by a utility function: $F_d(\vec{x_d}) = \beta \log(\sum_{l=1}^{L} x_d^l r^l)$. With notations summarized in Table 1, the optimal streaming solution can be found by solving the utility maximization problem **P1**.

P1: Utility Maximization under Taxation
Objective:

$$\max \sum_{d \in R} \log(\sum_{l=1}^{L} x_d^l r^l) \tag{1}$$

Constraints:

$$\sum_{\langle i,j\rangle \in E} g_{ij}^{l,d} - \sum_{\langle j,i\rangle \in E} g_{ji}^{l,d} = \begin{cases} x_d^l r^l, & i = S \\ -x_d^l r^l, & i = d \\ 0, & otherwise \end{cases}$$
$$\forall d \in R, \forall l \le L \tag{2a}$$

$$g_{ij}^{l,d} \le f_{ij}^l, \quad \forall l \le L, \forall d \in R, \forall \langle i,j\rangle \in E \tag{2b}$$

$$\sum_{l} \sum_{\langle i,j\rangle \in E} f_{ij}^l \le U_i, \quad \forall i \in V \tag{2c}$$

$$\sum_{l} x_d^l r^l \ge (1-t) \sum_{l} \sum_{\langle d,j\rangle \in E} f_{dj}^l \quad \forall d \in R \tag{2d}$$

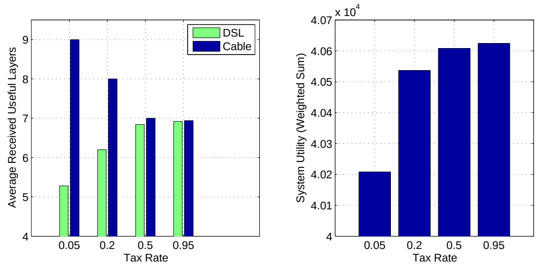
$$x_d^{l+1} \le x_d^l, \quad \forall l \le L, d \in R \tag{2e}$$

Constraint (2a) of **P1** guarantees the information flow conservation on each peer. In (2b), f_{ij}^l corresponds to the maximum information flow on $\langle i, j\rangle$ for all receivers of layer l. (2c) is the uplink capacity constraint for all layers on all peers and the server . (2d) guarantees that each peer should at least receive video at its entitled rate, which is proportional to its upload contribution. In SVC bitstream, higher layers depend on lower layers, and so peer d may request $l+1$ layer only if it has received all layers up to l. (2e) captures this dependency among layers.

P1 is a nonlinear mixed programing problem. It can be relaxed to reduce its computation complexity. Instead of using the PSNR-rate model, the video experience of a peer can be quantified by the weighted sum of all the received layers: $F_d(\vec{x_d}) = \sum_{l=1}^{L} x_d^l w^l$, where w^l is the weight assigned to layer l, with $w^i > w^j$, if $i < j$. In other words, the marginal gain of receiving a lower layer outweighs that of receiving higher layers. As a result, the optimal solution with the weighted-sum utility function will easily satisfy the constraint (2e). If we further relax the binary variables x_d^l in **P1** to continuous variables within $[0,1]$, the optimal solution will naturally have the property that $x_d^l > 0$ only if $x_d^k = 1, \forall k < l$. The original non-linear mixed integer programming problem is relaxed into a linear programming problem. More details can be found in our technical report [9].

2.2 Numerical Studies

Using the model, we conducted numerical case studies to understand the impact of taxation on layered P2P streaming. We consider a layered P2P streaming system with 40 peers, 15 of which have cable connections with upload bandwidth of 1,000 Kbps, and 25 of which have DSL connections with upload bandwidth of 400 Kbps. Each peer connects to six neighbors. To model the clustering effect observed in P2P streaming systems, we let a peer connects to another peer with the same type of access with 70% probablity. The server has upload capacity of 8 Mbps, and connects to eight random peers. The video is coded into 10 layers, each layer with a rate of 100 Kbps. The layer weight w^l is set as $2^{(10-l)}$. We vary the tax rate from 0.05 to 0.95 and solve relaxed **P1** using AMPL.



(a) Received Useful Layers (b) System Utility

Figure 2: Impact of taxation on fairness and system utility. (a) Averaged received layers for heterogeneous peers; (b) Achieved system-wide utility.

As can be seen from Fig. 2(a), when the tax rate is small, Cable peers with higher upload capacity obtain more layers. The service differentiation provides good incentives for them to participate in P2P sharing. As the tax rate increases, the differences between Cable peers and DSL peers decrease. On the other hand, the system wide utility increases with tax rate.

Clearly, the trade-off between efficiency, fairness and incentive can be balanced by adjusting the tax rate t. Higher tax rate introduces higher system utility and smaller tax rate moves closer to tit-for-tat type of fairness. When the tax rate tends to 0, the taxation degrades to the “tit-for-tat” or “bit-for-bit” strategy. In such a system, the system utility is obviously the lowest. Some poor peers can only receive a small portion of the video and thus obtain a rather degraded quality even though they contribute all of their uplink bandwidth. At the opposite side, when tax rate approaches 1, all peers retrieve the same video rate regardless of their contributions. Both scenarios are not desirable. Using the model, we have also conducted numerical studies to investigate the impact of peer heterogeneity and peering strategies. Due to the space limit, we refer interested readers to our technical report [9] for more details.

3. DISTRIBUTED PROTOCOL DESIGN

While the model allows us to understand the the trade-offs of taxation-based layered video streaming, our ultimate goal is to design distributed mesh-based streaming protocols to dynamically balance the needs of fairness, incentive and system efficiency. In our design, peers form a mesh over which the video is distributed. A tracker serves as the bootstrapping node for the system. The key design issues for such a layered P2P streaming protocol are *layer subscription*, *chunk scheduling*, and *mesh topology adaptation*.

Multiple virtual overlays, one for each SVC video layer, are formed among participating peers. Due to the dependency among video layers, the upper virtual overlays must have fewer peers than lower overlays. A peer uses layer subscription scheme to determine how many layers to subscribe to, and how many overlays to join. The chunk scheduling algorithms on peers allocate bandwidth among different overlays to balance the streaming needs of different layers. Finally, the mesh topologies need to be dynamically adjusted to adapt to the changing layer subscription due to peer churn and/or other network dynamics.

In the theoretic framework developed in Section 2, network coding is an essential component to achieve the optimum multicast efficiency in general overlay topology. The gain of adopting network coding in real P2P systems is still an open question [3]. In layered P2P streaming systems, applying network coding to individual layers incurs extra coding/decoding overhead, increases video playback delays, and makes the protocol design more complex. Recent study [11] showed that when peers are fully connected and peer uplinks are the only bottlenecks, network coding is not needed. Even though we don’t assume peers are fully connected, our distributed design does not employ network coding. We will show through simulations that the performance of the proposed mesh-based P2P streaming design is very close to the performance bound allowed by network coding.

3.1 Dynamic Layer Subscription

Under linear tax rate t, the target video download rate of peer d is $r_d = (1-t)U_d + \frac{t}{N}\sum_i U_i$, where $(1-t)U_d$ is peer d’s entitled rate and $\frac{t}{N}\sum_i U_i$ is peer d’s excess rate. Entitled and excess rates are then mapped to the number of entitle and excess layers. Tax rate t is a global configuration parameter and is known to all peers. Therefore peers can compute the number of entitled layers locally. However, the calculation of the number of excess layers needs global information - all active peers’ uplink bandwidth. The number of excess layers on a peer also varies as other peers join and leave the system. We develop a distributed algorithm that probes peers’ number of excess layers and dynamically adjusts peers’ layer subscriptions. The algorithm allows system to approach the utility maximization under taxation, and handle the peer churn and network dynamics nicely.

Let L_i denote peer i’s entitled layers, and l_i denote the highest layer it is subscribed to. Motivated by the distributed utility maximization achieved by TCP in congestion control [17], we propose a distributed layer subscription algorithm with *Additive Increase Additive Decrease (AIAD) and exponential backoff*. Upon joining the streaming session, peer i sets its initial layer subscription, l_i, to be L_i, the number of its entitled layers. It also starts a *retry timer*, $t_i = rand(1,T)$, where T is the retry time period. Upon the expiration of the retry timer, if all currently subscribed layers can be received and at least one neighbor peer possesses chunks of layer

$l_i + 1$, peer i increases its subscribed layer by one, $l_i = l_i + 1$, and enters a *trial period* of T'. Peer i sends out requests for chunks in the newly added layer. If peer i is able to successfully obtain most of requested chunks of the new layer at the end of the trial period, it passes the test and the new layer subscription is accepted. Otherwise, peer i reverts back to original subscription, and enters an exponential back-off stage. The retry timers is set to be $t_i = rand(1, 2^k T)$, where k is number of consecutive failures. Meanwhile, peer i runs a parallel *subscription decrease process* to ensure that it can receive all subscribed layers. Subscription decrease process periodically monitors the status of received layers. If the top subscribed layer, l_i, becomes undecodable, and peer is not in the aforementioned trial period, peer i reduces the number of subscribed layers to $l_i = \max(l_i - 1, L_i)$.

3.2 Chunk Scheduling

Each peer maintains a downloading window that moves forward periodically. Peers periodically exchange chunk availability with their neighbors using buffer-maps. Neighbors help each other retrieve missing chunks. Chunk scheduling decides how to issue chunk requests to neighbor peers, and how to serve the chunk requests from neighbor peers. The goal is to properly utilize peers' uplink bandwidths so that peers always receive the entitled layers and receive the subscribed excess layers with high probability.

3.2.1 Chunk requesting

In SVC coded video, lower layer bit-stream is more important than higher layer bit-stream. Hence in principle, lower layer chunks should be requested before higher layer chunks. In order to increase the data chunk diversity and improve the chance that two peers always have chunks to exchange, we further assume that data chunks belonging to the entitled layers are equally important. This is reasonable because the aggregated upload bandwidth in the system is sufficient to deliver the entitled layers to all peers. There is no need to distinguish different entitled layers. The chunks are requested in the order of their importance: from entitled layer chunks to excess layer chunks. A peer selects one neighbor peer that owns the missing chunk to request for the chunk. The probability of choosing a specific peer is proportional to its serving rate to the requesting peer. For example, if requester R serves neighbors A, B and C with 20Kbps, 50Kbps and 30Kbps respectively, it then sends the chunk request to one of them with probability (0.2, 0.5, 0.3).

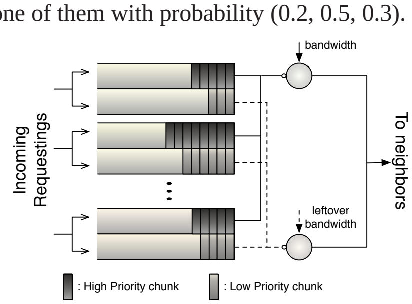


Figure 3: Peer serves neighbors. Low priority chunks will be served only if high priority queues are empty.

3.2.2 Chunk serving

Chunk serving is more sophisticated. Individual peers maintain two FIFO queues for each neighbor (see Fig. 3). One queue is called entitled queue and the other is called excess queue. Entitled queue holds chunk requests for entitled layers, while excess queue holds chunk requests for excess layers. The chunk requests in excess queues are sorted in ascending order of video layers, with lowest layer chunk requests at the head. The entitled queues have strict priority over the excess queues. Excess queues would not be served unless all entitled queues become empty. If entitled queues become empty, the leftover bandwidth serves the requests in excess queues in a round robin fashion. The requests that have passed their playback deadlines are cleared out of the queues and won't be served. Before serving, peer executes a pre-bandwidth allocation in proportion to neighbors' contribution. If, say, one neighbor did not use up the pre-allocated bandwidth, the excess portion would then be used to serve other neighbors who need more bandwidth than the allocated.

3.3 Mesh Topology Adaptation

In this section, we consider how to efficiently adapt the mesh topology to achieve different design goals. Well organized mesh is critical in removing content bottleneck and improving resource utilization. Mesh topology adaptation is achieved through neighbor adaptation. A peer periodically contacts the tracker to retrieve a list of candidate neighbors. It then applies the adaptation strategy as described below to ensure the overlay topology converging to the desired topology. Peer adaptation is a good alternative to the complex network coding in removing content bottleneck. Every peer has a preset peer out-degree. If the number of neighbors falls below the preset out-degree, a peer increases the number of neighbors by adding neighbors randomly selected from the candidate list. If the current number of neighbors is ok, a peer still selects one peer with low contribution and replaces it with a new peer from the candidate list. Specifically, a peer uses a *replacement index* to determine which peer to be replaced. Suppose peer i needs to adapt its neighbors. Let c_j^l be the number of retrieved chunks of layer l from peer j, and w_l be the weight associate with layer l. The replacement index for peer j is defined to be $\sum_{l \in i's\ entitled\ layers} c_j^l w_l$. In addition, weight w is set to be as such that $w_l > w_k$ if $l > k$. The neighbor with the smallest replacement index is selected and swapped out. The length of the adaptation period is chosen as ten seconds in our design. The design philosophy is two-fold.

- *Layer Level:* a neighbor offering high level layers up to the entitled layers should stay. There are fewer peers in the higher virtual overlay. Peers who can offer high layer chunks are more precious and are more likely of the same class (with the same entitled layers).
- *Chunk Level:* among all neighbors offering chunks at the same layer, those uploading more chunks should stay.

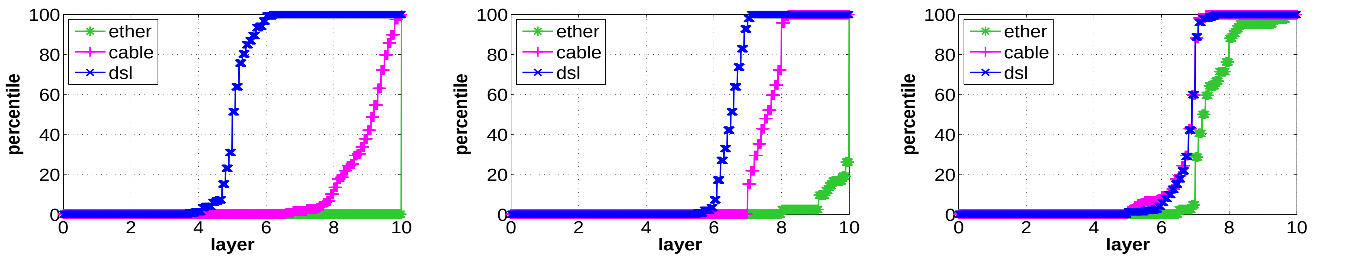
4. PERFORMANCE EVALUATION

We conduct trace-driven simulations to evaluate the performance of the proposed taxation-based P2P layered streaming design. Specifically, we investigate the following aspects: (1) the effectiveness of taxation-based incentive mechanism; (2) peer uplink bandwidth utilization; (3) mesh overlay adaptation, and (4) the convergence and optimality of AIAD layer subscription scheme.

4.1 Simulation Setup

A flow-level event-driven simulator is developed in C++. Unless stated otherwise, the simulations are driven by peer arrival and departure trace collected from the measurement study of PPlive [7]. The trace was collected from Nov 22nd 17:43, 2006 to Nov 23rd 17:43, 2006. All peer arrivals and departures in a video channel are recorded during this period. The number of concurrent peers varies from 100 to more than 9,000 reflecting the high dynamics in P2P streaming system. The video is encoded into ten layers with layer rate of 100 Kbps.

There are three types of peers: DSL peers (400 Kbps), Cable



(a) Tax rate 0 (b) Tax rate 0.5 (c) Tax rate 0.95

Figure 4: Received video layers for different types of peers under various tax rates. (a) CDF of received layers for different peers under tax rate 0; (b) CDF of received layers for different peers under tax rate 0.5; (c) CDF of received layers for different peers under tax rate 0.95.

Table 2: Peer upload bandwidth distribution

Peer Type	Uplink Bandwidth	Percentage
DSL	400 Kbps	45%
Cable	800 Kbps	40%
Ethernet	1500 Kbps	15%

Table 3: System Bandwidth Utilization

Tax rate	UBU	WBR
0	99.4%	0.4%
0.5	97.9%	0.1%
0.95	93.1%	0.6%

peers (800 Kbps) and Ethernet peers (1500 Kbps). The fraction of individual peer types and their respective uplink bandwidths are summarized in Table 2. In our simulation, there is one video server with upload capacity of 10 Mbps.

The peer download window is set to 30 seconds. Peers exchange buffer-maps every second to calculate the missing chunk downloading schedule. Mesh topology adaptation is conducted every ten seconds. The values of T and T' in AIAD layer subscription algorithm are set to be 5 seconds and 10 seconds, respectively.

4.2 Simulation Results

4.2.1 *Effectiveness of Taxation Based Incentive*

In taxation based P2P streaming, a peer's received video quality, or the number of layers, reflects its bandwidth contribution and the system wide tax rate. In addition, the peers with similar bandwidth contributions receive similar video quality. Both are true as shown in Fig. 4, which depicts the Cumulative Distribution Functions (CDFs) of the numbers of received layers for different types of peers at different tax rates. The peers from the same class consistently receive a similar number of layers, while the numbers of video layers received by different peer classes are close to the optimum values–(5,9,10) under tax rate 0; (6,8,10) under tax rate 0.5 and (7,7,8) under tax rate 0.95.

4.2.2 *Bandwidth Utilization Efficiency*

Peers' uplink bandwidth utilization is a key performance metric for any P2P streaming system design. If the system is not well designed, the so-called "content bottleneck" lowers down the uplink bandwidth utilization, and degrades the average peers' received video quality.

Table 3 lists the uplink bandwidth utilization (UBU) and the wasted bandwidth ratio (WBR). Overall, the uplink bandwidth utilization is consistently over 90%, indicating the efficiency of the protocol that can alleviate content bottleneck even without network coding. Interestingly, UBU is worse at larger tax rates. As tax rate increases, the peers become more altruistic, which requires more bandwidth sharing among different types of peers. Due to the peer churn and mesh topology constraint, the bandwidth sharing may not be always possible, thus lower the utilization.

In layered video, received chunks become undecodable if the lower layers are not fully decoded. Wasted bandwidth ratio (WBR) defines the fraction of bandwidth that is used for delivering undecodable chunks. Again, the wasted bandwidth ratio is pretty small, pointing to an efficient protocol design.

4.2.3 *Mesh Overlay Adaptation*

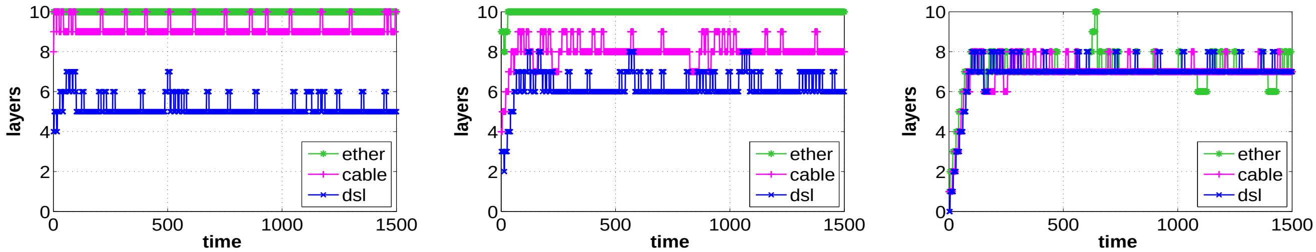
Mesh overlay topology plays a key role in service differentiation. Table 4 and 5 list the peer neighborhood statistics with the tax rate of 0 and 0.95, respectively. With tax rate 0, the optimal number of video layers for Ethernet users, Cable users, and DSL users are 10, 9, and 5, respectively. Around 76% of Ethernet peers' neighbors are either Ethernet or Cable users, and around 77% of Cable peers' neighbors are either Ethernet or Cable users. In contrast, DSL users mainly connect with other DSL users (70%). The strong bias towards connecting with similar peers leads to a hierarchical mesh topology, which allows Ethernet and Cable users to exchange higher video layers (from layer 6 to layer 10) that are not available at DSL users.

With tax rate 0.95, all peers are supposed to receive a similar number of video layers regardless of their individual bandwidth contributions. For DSL users, the fraction of DSL neighbors is reduced from 70% (with tax rate 0) to 44%. For Ethernet users, the fraction of DSL neighbors is increased from 24% to 40%. Compared with the mesh topology constructed at tax rate 0, this is a more randomized topology for the peer distribution in Table 2.

4.2.4 *Layer Subscription Convergence*

In order to examine the behavior of layer subscription algorithm without the impact of peer churn, 500 peers with fixed topology is used in this experiment. The peers' uplink bandwidth obeys the distribution as stated in Table 2. We randomly pick one peer from each bandwidth category and plot the evolution of its layer subscription. We also vary the tax rate to examine its impact.

Fig. 5 shows the layer subscription process with tax rate of 0, 0.5, and 0.95, respectively. With tax rate zero, peers are entirely selfish. Ethernet peers with bandwidth of 1500 Kbps receive all ten layers. The leftover bandwidth subsidizes other peers. As a result, the optimal layer subscription for Cable and DSL peers are 9 layers and 5 layers, respectively. With tax rate of 0.5, the optimal layer subscription for DSL, Cable, and Ethernet peers are 6 layers, 8 layers, and 10 layers, respectively. Finally, with tax rate of 0.95, peers are altruistic and every peer should receive 700 Kbps except for Ethernet peers (800 Kbps). Since video is encoded at 100 Kbps per layer, there are more "free" bandwidths in this case, introducing minor oscillations in layer subscription. In all cases, AIAD algorithm is



(a) Tax rate 0 (b) Tax rate 0.5 (c) Tax rate 0.95

Figure 5: Layer subscriptions over time for different peers under various tax rates. (a) Subscription dynamics for different peers under tax rate 0; (b) Subscription dynamics for different peers under tax rate 0.5; (c) Subscription dynamics for different peers under tax rate 0.95.

Table 4: Topology Statistics For tax rate 0

Neighbor Type	Ethernet	Cable	DSL
Ethernet	28.4%	19.0%	9.0%
Cable	47.3%	58.1%	21.1%
DSL	24.3%	22.9%	69.9%

Table 5: Topology Statistics For tax rate 0.95

Neighbor Type	Ethernet	Cable	DSL
Ethernet	21.1%	14.9%	14.5%
Cable	38.5%	39.9%	41.5%
DSL	40.4%	45.2%	44.0%

able to quickly converge to the target subscription layer and peers stay in their optimal layers for most of the time. The simulation results in [9] further show that peers receive good viewing quality consistently.

5. CONCLUSIONS

Designing an efficient P2P live streaming system that is fair to all peers and offers strong incentive for them to contribute is challenging. In this paper, we integrate taxation-based incentive mechanism into P2P layered streaming, and develop a practical streaming system. Taxation-based P2P streaming allows us to freely adjust the balance between the social welfare and individual peer welfare. Extensive trace-driven simulations demonstrate that the proposed designs can effectively drive layered P2P streaming systems to operating points with the desired balance between efficiency, fairness and incentive.

6. REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Trans. on Information Theory*, 46:1204–1216, 2000.

[2] M. Chen, M. Ponec, S. Sengupta, J. Li, and P. A. Chou. Utility Maximization in Peer-to-peer Systems. In *Proceeding of ACM SIGMETRICS*, June 2008.

[3] D. M. Chiu, R. W.-H. Yeung, J. Huang, and B. Fan. Can network coding help in p2p networks? In *Second Workshop of Network Coding*, Boston, USA, April 2006.

[4] B. Cohen. Incentives build robustness in bittorrent. In *Proceedings of the 1st Workshop on Economics of Peer-to-Peer systems*, 2003.

[5] K. Eger and U. Killat. Fair resource allocation in peer-to-peer networks. *Computer Communications*, 30:3046–3054, 2007.

[6] A. Habib and J. Chuang. Service differentiated peer selection: An incentive mechanism for peer-to-peer media streaming. *IEEE Trans. on Multimedia*, 8:610–621, 2006.

[7] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross. A measurement study of a large-scale P2P IPTV system. *IEEE Trans. on Multimedia*, 9:1672 – 1687, 2007.

[8] P. K. Hoong and H. Matsuo. Palms: A reliable and incentive-based p2p live media streaming system. *Lecture Notes in Electrical Engineering*, pages 51–66, 2008.

[9] H. Hu, Y. Guo, and Y. Liu. Peer-to-peer streaming of layered video: Efficiency, fairness and incentive. In *Technical Report, Polytechnic Institute of NYU*, 2009.

[10] Y. hua Chu, J. Chuang, and H. Zhang. A case for taxation in peer-to-peer streaming broadcast. In *ACM SIGCOMM workshop on Practice and theory of incentives in networked systems*, 2004.

[11] J. Li, P. A. Chou, and C. Zhang. Mutualcast: An Efficient Mechanism for Content Distribution in a P2P Network. In *Sigcomm Asia Workshop*, 2005.

[12] Z. Liu, Y. Shen, S. S. Panwar, K. W. Ross, and Y. Wang. Substream Trading: Towards an Open P2P Live Streaming System. In *ICNP*, 2008.

[13] N. Magharei, R. Rejaie, and Y. Guo. Mesh or multiple-tree: A comparative study of live P2P streaming apprаches. In *Proc. IEEE INFOCOM*, May 2007.

[14] Y.-W. Sung, M. Bishop, and S. Rao. Enabling Contribution Awareness in an Overlay Broadcasting System. In *SIGCOMM*, 2006.

[15] M. Wang and B. Li. R2: Random Push with Random Network Coding in Live Peer-to-Peer Streaming. *IEEE JSAC*, pages 1655–1666, December 2007.

[16] M. Wien, H. Schwarz, and T. Oelbaum. Performance Analysis of SVC. *IEEE Trans. Circuit and System for Video Technology*, 17:1194–1203, Sep. 2007.

[17] S. woo Cho and A. Goel. Pricing for fairness: distributed resource allocation for multiple objectives. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, 2006.

[18] X. Xiao, Y. Shi, Y. Gao, and Q. Zhang. Layerp2p: A new data scheduling approach for layered streaming in heterogeneous networks. In *IEEE INFOCOM*, 2009.

[19] M. Zghaibeh and K. Anagnostakis. On the impact of p2p incentive mechanisms on user behavior. In *NetEcon+IBC*, 2007.

[20] X. Zhang, J. Liu, B. Li, and P. Yum. Coolstreaming/DONet: A data-driven overlay network for efficient live media streaming. In *IEEE INFOCOM*, 2005.

[21] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang. LION: Layered Overlay Multicast with Network Coding. *IEEE Trans. Multimedia*, 8:1021–1032, Oct. 2006.