

# Peer-to-Peer Streaming of Layered Video: Efficiency, Fairness and Incentive

Hao Hu<sup>†</sup>, Yang Guo<sup>\*</sup>, and Yong Liu<sup>†</sup>

<sup>†</sup>Electrical & Computer Engineering, Polytechnic Institute of NYU, Brooklyn, NY 11201

<sup>\*</sup> Research and Innovation, Technicolor, Princeton, NJ 08540

**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 first develop utility maximization models to understand the interplay between efficiency, fairness and incentive in layered P2P streaming. We show that taxation mechanisms can be devised to strike the right balance between social welfare and individual peer welfare. We then 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.

## I. INTRODUCTION

P2P live video streaming has recently emerged as a cost-efficient IPTV solution on the Internet. It has attracted substantial attentions in research community and industry. Several widely deployed commercial P2P live streaming systems [1]–[3] routinely attract hundreds of thousands of users to watch live video broadcast online. Extensive research [4]–[10] has studied various aspects of system design, ranging from P2P overlay construction, data sharing strategy, adoption of advanced video coding, to the application of network coding technique. In-depth measurement [11]–[13] study further allows us to observe the working systems in the field.

The 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. With scalable video coding (SVC) [14], 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% [15]. It is now

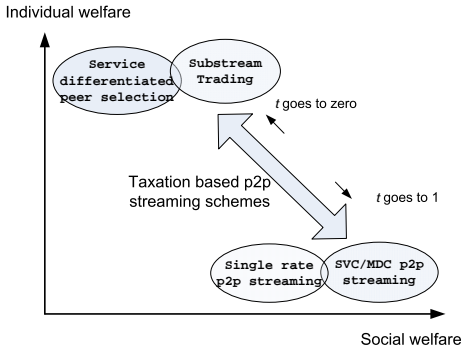
practical to adopt SVC into P2P video streaming to extend its design space.

The adoption of SVC into P2P streaming faces two key design challenges: 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 peers, the notion of fairness is much more subtle than that in the 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. A good layered P2P streaming solution has to strike the right balance between efficiency, fairness and incentive.

In this paper, we develop analytical models and practical streaming designs to understand and control the interplay between efficiency, fairness and incentive in layered P2P streaming. Specifically, we develop network-coding based utility maximization models to obtain the most efficient layered streaming solution. The choice of peer utility function reflects the target fairness among peers when they are considered only as video consumers. To incorporate contribution-awareness, we adopt *taxation* as a peer-incentive mechanism and augment the utility maximization models to make the solution incentive-compatible. Taxation based incentive mechanism [16], [17] offers a flexible framework that allows the tradeoff between individual users' fairness/welfare and the system-wide social welfare (see Fig. 1). 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 video rate on a peer consists of two parts: a fraction of its own contribution, and a fair share from the pool of taxed bandwidth. The tax rate  $t$  adjusts the balance between individual peers' welfare and the social welfare. As  $t$  approaches zero, the received video rate approaches 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 as described in Section II.

Our contribution can be summarized as follows:

- We develop utility maximization models to study the interplay between efficiency, fairness and incentive in lay-



**Fig. 1:** Social welfare vs. personal welfare. The tax rate  $t$  determines the balance between social welfare and personal welfare.

ered P2P streaming. The models enable us to numerically investigate the impact of peering strategies and chunk scheduling policies on the fundamental trade-offs between the three.

- We adopt taxation mechanism to adjust the balance between the social welfare and individual peer welfare. We develop practical taxation-based P2P layered streaming designs, including layer subscription strategy, chunk scheduling policy, and mesh topology adaptation.
- We evaluate our system designs with extensive trace driven simulations. The results show that the proposed layered streaming designs can effectively drive P2P streaming systems to the desired operating points in a distributed fashion.

The remaining of the paper is organized as follows. We briefly go through the related work in the following section. Utility-maximization models and numerical studies are presented in Section II. Taxation-based layered streaming designs are presented in Section III. The performance of the proposed designs are evaluated in Section IV. The paper is concluded in Section V.

### A. Related Work

The majority of existing P2P live streaming systems can be classified into two groups: one group targeting at maximizing the aggregate received video quality, namely social welfare; the other striving for the fairness among peers, or individual welfare. Fig. 1 depicts the relationship between these two groups of strategies, which have contradicting goals that are not attainable simultaneously.

Layered coding, such as SVC [14] or MDC [18], allows users to receive the same video with different qualities. A user's perceived video quality is proportional to the number of received video layers. SVC encodes a video into multiple layers with nested dependency - an upper layer becomes decodable only if all layers beneath it have been received and decoded successfully. In contrast, MDC encodes a video into layers that are independently decodable. Hence MDC is more flexible than SVC. Our goal, however, is to design a layered P2P protocol that can be implemented and deployed in the field. The current designs of MDC still incur much

higher bandwidth overhead than that of SVC. We thus choose SVC over MDC in this paper. We address the challenge of supporting SVC type of inter-correlated layered video in P2P streaming.

Layered coding has been applied to P2P streaming to improve the social welfare over the traditional single layer P2P streaming. For instance, [4] studies how to use layered coding to fully utilize the available peer upload bandwidth in a tree-based P2P overlay multicast. [19] proposes a 3-stage chunk scheduling algorithm for mesh-based layered video streaming to achieve high throughput and low video quality jitter. In terms of achieving individual fairness/welfare, Cohen advocated a tit-for-tat algorithm in the seminar paper [20]. [21] proposes a score-based incentive mechanism for P2P live streaming. [22] proposes a service differentiated peer selection algorithm that gives peers with higher contributions more flexibility in choosing neighbors, thus obtain better viewing quality. [23] and [10] utilize layered coding to achieve fairness. Substreams/layers are traded among peers, and peers contributing more are able to receive more in reciprocity. In contrast, our paper develops taxation mechanisms to strike the right balance between social welfare and individual peers' welfare. The practical taxation-based P2P streaming protocol is designed to drive the system converge to the desired operating point.

The work in [16], [17] 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 [6], [24]. In this paper, we study mesh-based SVC P2P streaming with taxation. To the best of our knowledge, the problem of applying taxation to SVC type of layered P2P video streaming has not been systematically studied before.

## II. MODELING LAYERED P2P STREAMING

To gain insights into layered P2P streaming, we first develop analytical models for layered P2P streaming systems with arbitrary topologies. The models not only allow us to analytically study the interplay between the efficiency, fairness and incentive, but also offer us guidelines in designing the practical system.

### A. Maximizing Efficiency

When peers are cooperative, they are willing to contribute their upload bandwidth without any incentive. The design objective of the system is to maximize the aggregate video quality on all peers. With layered coding, the perceived video quality on a peer is an increasing function of the number of video layers received. PSNR (Peak Signal-to-Noise Ratio) is the standard objective metric to evaluate the quality of a compressed video and thus can be adopted as the utility function in layered video streaming. PSNR of a video coded at rate  $r_c$  can be approximated by a logarithmic function

$\beta \log(r_c)$  [25], where  $\beta$  is a constant related to the video feature. This approximation is also valid in the SVC case [14]. Let  $r_d$  be the total rate of received video layers on peer  $d$ . If the aggregate uploading capacity of the server and all peers is  $U$ , the aggregate receiving rate on all peers is naturally bounded by  $\sum_{d=1}^N r_d \leq U$ . Since  $\log(\cdot)$  is a concave function, the aggregate utility can be maximized when all peers receive video at the same rate, i.e.,  $r_d = \frac{U}{N}$ . For single-layer video streaming, it was shown in [26] that, if peers are fully connected, a two-hop relay streaming can achieve this optimal rate. The solution for layered video streaming naturally follows if we let all peers subscribe to the same number of video layers allowed by the rate  $\frac{U}{N}$  and deliver each layer to all peers using the two-hop relay scheme. However, it is unrealistic to have fully connected mesh in a large-scale streaming system. For arbitrary streaming topology, the utility maximization in layered streaming deserves more study.

1) *Network Coding Model*: 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  $k$ ,  $k \leq L$ , 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.

It is difficult to accurately model a mesh-based P2P streaming system with arbitrary overlay topology due to the content bottleneck [8]. Fortunately, by assuming network coding, we can model the system as a closed-form optimization problem for arbitrary topologies with peers' uplink capacity as the constraint. We allow the server and peers apply *network coding* to video blocks. Network coding has been shown to achieve the maximum multicast rate for single multicast session in general network topology [27]. For multiple multicast sessions, *inter-session network coding* might be needed to achieve the maximal multicast rates. However, the complexity of inter-session network coding is generally too high to be justified by its additional performance gain on top of intra-session network coding. In this paper, we only focus on intra-session network coding. The server and peers apply network coding to video blocks in the same layer.

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  $R = V \setminus S$  be the set of peers interested in receiving the video. Let  $\vec{x}_d = (x_d^1, x_d^2, \dots, x_d^L)$  be the binary vector 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  $l$ . According to the theory of intra-session network coding [27], [28], 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.

**TABLE I:** Notations

Notation	Description
$V$	set of nodes in the system
$E$	set of overlay links
$S$	video source server
$R = V \setminus S$	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 $d$

**P1: Utility Maximization**

**Variables:**

$g_{ij}^{l,d}$	continuous non-negative variable
$f_{ij}^l$	continuous non-negative variable
$x_d^l$	binary variable

**Objective:**

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

**Constraints:**

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

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

$$x_d^{l+1} \leq x_d^l, \quad \forall l \leq L, d \in R \quad (2c)$$

$$\sum_l \sum_{\langle i, j \rangle \in E} f_{ij}^l \leq U_i, \quad \forall i \in V \quad (2d)$$

By adopting the PSNR-Rate model, 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 I, the optimal streaming solution can be found by solving the utility maximization problem **P1**.

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  $l$ . 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  $l$ . (2c) captures this dependency among layers. (2d) is the uplink capacity constraint for all layers on all peers and the server.

**B. Achieving Fairness**

In the traditional resource allocation problems, utility maximization achieves different notions of fairness between competing resource consumers. In P2P video streaming, each peer plays a dual-role of server and consumer. We ignore peer's server role in the contribution-oblivious utility maximization. The obtained optimal solution can also be interpreted as fairness among peers without considering their contributions. Within the fairness context, it is straightforward to show that the solution of the utility maximization problem **P1** achieves

the *proportional fairness* [29] among peers under the given overlay topology  $\mathcal{G}$  and node upload capacity profile  $\mathcal{U}$ .

Another commonly used fairness measure is the *weighted fairness*. In the context of layered streaming, because of the layer dependency in SVC encoding, peers have to first retrieve lower layers. A solution achieving weighted fairness should give priority to streaming lower layers to peers. Weighted fair streaming solution can be obtained by replacing the PSNR-rate utility function in **P1** with a weighted-sum function. We assign weights to layers in a decreasing order. Let  $w^l$  be the weight assigned to layer  $l$ . We have  $w^i > w^j$ , if  $i < j$ . Instead of using PSNR-rate model, the video experience of a peer is characterized by the summation of the weights of all the received layers:  $F_d(\vec{x}_d) = \sum_{l=1}^L x_d^l w^l$ . 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 (2c). 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  $x_d^k = 1, \forall k < l$ . Formally, the original non-linear mixed integer programming problem is relaxed into the following linear programming problem.

### P2: Linear Approximation

Variables:

$g_{ij}^{l,d}$	continuous non-negative variable
$f_{ij}^l$	continuous non-negative variable
$x_d^l$	$\in [0, 1]$ , continuous variable

Objective:

$$\max \sum_{d \in R} \sum_{l=1}^L x_d^l w^l \quad (3)$$

Constraints: (2a), (2b), (2d)

The solution of the linear programming problem **P2** gives *weighted priority* for peers to receive lower layer video. Another commonly employed fairness criterion is the *max-min fairness*. Similar to the max-min network flow allocation [30], the max-min fairness in layered streaming can be achieved using the following ‘‘onion-peeling’’ solution.

- 1) set iteration counter  $k = 1$ ; Initialize network topology  $\mathcal{G}(1) = \mathcal{G}$ , receiving peers  $R(1) = R$ , and node upload capacity profile  $U(1) = U$ ; Initialize the layer counter to  $l = 1$  and the receiving vector  $x_d^l = 0, \forall d, l$ .
- 2) for a given P2P streaming system  $\{\mathcal{G}(k), R(k), U(k)\}$ , find the maximum supportable single-layer multicast rate  $r^*(k)$  by solving the linear programming problem **P3**( $\mathcal{G}(k), R(k), U(k)$ ), record the consumed upload bandwidth  $\{A_d^*(k), d \in R(k)\}$  in this iteration according to output (5).
- 3)  $w(k) = 0$ ; while  $(w(k) + r^l < r^*(k))$

$$\{x_d^l = 1, \forall d \in R(k); w(k) += r^l; l ++ \}$$

- 4) if  $(w(k) == 0)$  exit;
- 5) update the remaining peer upload bandwidth

$$U_d(k+1) = U_d(k) - A_d^*(k)w(k)/r^*(k);$$

- 6) update topology  $\mathcal{G}(k)$  and receiver set  $R(k)$  in the following way:

- a) for any peer  $d$  with zero remaining upload bandwidth,  $U_d(k+1) == 0$ , remove from  $\mathcal{G}(k)$  all egress links of  $d$ ,  $\langle d, \cdot \rangle$ ;
  - b) remove a peer from  $\mathcal{G}(k)$  and  $R(k)$  if it has no ingress link  $\langle \cdot, d \rangle$ .
- 7) set  $k = k + 1$ , go back to step 2.

Essentially, we solve the problem iteratively. At each iteration, we find the maximum multicast rate to all active receivers (Step 2). The multicast rate is utilized to send the same number of layers, starting from the lowest layer, to all receivers (Step 3 and 4). Peer upload bandwidth is updated (Step 5). A peer is removed from the receiver set and streaming system if all its neighbors have no upload bandwidth (Step 6). The remaining peers can receive additional layers in the next iteration (Step 7). In such approach, we guarantee that, the receiver set for a higher layer is always a subset of the receiver set of a lower layer, and at each layer, the receiver set is maximized.

At each iteration, we only need to solve a linear programming problem **P3**, which has much less variables than in **P1** and **P2**. We will evaluate **P2** and **P3** in the following section.

### P3( $\mathcal{G}, R, U$ )

Variables:

$r$	supportable multicast rate
$g_{ij}^d$	information flow for peer $d$ on link $\langle i, j \rangle$
$f_{ij}$	physical flow on link $\langle i, j \rangle$
$A_i$	consumed upload bandwidth on peer $i$

Objective:

$$\max r$$

Constraints:

$$\sum_{\langle i,j \rangle \in V} g_{ij}^d - \sum_{\langle j,i \rangle \in V} g_{ji}^d = \begin{cases} r, & i = S \\ -r, & i = d \\ 0, & \text{otherwise} \end{cases} \quad \forall d \in R, \quad (4a)$$

$$g_{ij}^d \leq f_{ij}, \quad \forall d \in R, \forall \langle i, j \rangle \in E \quad (4b)$$

$$\sum_{\langle i,j \rangle \in E} f_{ij} \leq U_i, \quad \forall i \in V \quad (4c)$$

Output:

$$r^*; A_i^* = \sum_{\langle i,j \rangle \in E} \max_d g_{ij}^{*d} \quad (5)$$

### C. Providing Incentives

In the previous efficiency and fairness study, we do not consider any incentive issues. This could cause 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.

1) *Taxation*: It is well-known that, in social welfare theory, taxation can help to improve the total utility of the whole society while maintaining a certain level of fairness. An optimal tax rate is usually non-linear and is complicated to determine given that taxes distort user behaviors. Here, we adopt the simple linear taxation method in [16], [17], that is

$$r_d = (1 - t)U_d + \frac{t}{N} \sum_i U_i, \quad (6)$$

where  $t$  is the tax rate,  $N$  is the total number of peers. Unlike the definition in [17], 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. The trade-off between efficiency, fairness and incentive can be balanced by adjusting the tax rate  $t$ . Higher tax rate introduces higher system utility; smaller tax rate moves closer to tit-for-tat type of fairness. When the tax rate equals 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 is 1, all peers retrieve the same video rate regardless of their contributions. Clearly, both scenarios are not desirable.

2) *P2P Layered Streaming with Taxation*: Two kinds of P2P layered streaming designs can be considered under taxation: *Equal share* and *Biased share*.

*Equal share*: In this case, the taxation pool is equally shared by all tax payers, i.e., participating peers in the system, which is exactly following (6). To perfectly implement the taxation scheme, one has to fully utilize the upload bandwidth available on all peers in the system. A sophisticated scheduling design is needed to meet this requirement by avoiding wasting bandwidth as much as possible.

*Biased share*: We only require a peer to receive all its entitled layers, i.e.,  $r_d \geq (1 - t)U_d$ . The bandwidth in the common taxation pool is distributed to maximize the system-wide utility. Towards this goal, we augment the utility maximization models studied in the previous sections by imposing an additional constraint on peer's receiving rate. At a given tax rate  $0 < t \leq 1$ , the utility maximization problem **P2** can be reformulated as **P4** follows.

Constraint (8d) guarantees that every peer should at least receive video at a rate proportional to its uploading contribution.

#### D. Numerical Studies

To gain insights on the interplay between efficiency, fairness and incentive, we conducted numerical studies on example systems with different topologies. For each system, we solve **P2** and **P3** and **P4** using AMPL [31]. The obtained numerical results allow us to study the impact of streaming topology on the system performance. We first solve **P2** and **P3** on a streaming overlay topology with 40 peers. There is one server and three types of peers with different upload bandwidth as summarized in Table II. The server has degree of 8, and only connects to Ethernet or Cable peers. We vary the degree

#### P4: Utility Maximization under Taxation

Variables:

$$\begin{aligned} g_{ij}^{l,d} & \text{ continuous non-negative variable} \\ f_{ij}^l & \text{ continuous non-negative variable} \\ x_d^l & \in [0, 1], \text{ continuous variable} \end{aligned}$$

Objective:

$$\max \sum_{d \in R} \sum_{l=1}^L x_d^l w^l \quad (7)$$

Constraints:

$$\sum_{(i,j) \in E} g_{ij}^{l,d} - \sum_{(j,i) \in E} g_{ji}^{l,d} = \begin{cases} x_d^l r^l, & i = S \\ -x_d^l r^l, & i = d \\ 0, & \text{otherwise} \end{cases} \quad \forall d \in R, \forall l \leq L \quad (8a)$$

$$g_{ij}^{l,d} \leq f_{ij}^l, \quad \forall l \leq L, \forall d \in R, \forall (i,j) \in E \quad (8b)$$

$$\sum_l \sum_{(i,j) \in E} f_{ij}^l \leq U_i, \quad \forall i \in V \quad (8c)$$

$$\sum_l x_d^l r^l \geq (1 - t) \sum_l \sum_{(d,j) \in E} f_{dj}^l \quad \forall d \in R \quad (8d)$$

TABLE II: Nodes' Settings

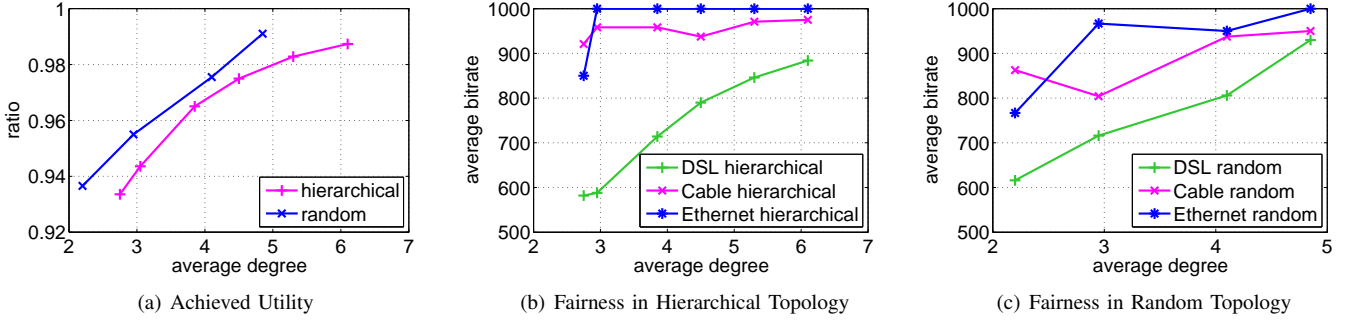
Type	Uplink Capacity	Number
Server	8000 Kbps	1
Ethernet peer	4000 Kbps	3
Cable peer	1000 Kbps	12
DSL peer	400 Kbps	25

of peers to investigate the impact of peer connectivity. The maximum peer degree is 10. Peers can operate at two different modes to select neighbors. One is the hierarchical mode in which a peer prefers to connect to peers with the same type (with 70% probability). The other is the random mode in which peers randomly connect to other peers, regardless of their types. The video stream is coded in 20 layers, each layer is encoded at rate of 50 Kbps.

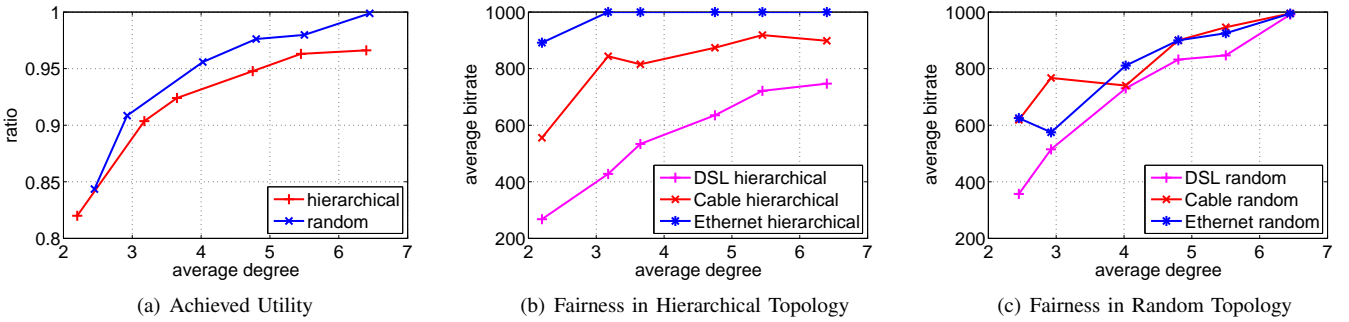
Fig. 2 shows the solutions of **P2** under different average peering degree. Fig. 2(a) compares the aggregated utility in hierarchical and random cases. The Y-axis value is obtained by dividing the aggregated utility by the maximum possible utility (here it is the case when every receiver gets  $\frac{U}{N}$ ). The system utility increases as peering degree increases. The random mode gives better system wide utility. Fig. 2(b) and 2(c) compare the average receiving rate for each type of peers under the two peering modes. The hierarchical mode gives higher service differentiation. In the protocol design described at Section III-C, we introduce mesh topology adaptation to ensure the topology suits the taxation strategy.

In Fig. 3, we compares the aggregated utility, average received rate for each type of peers in both hierarchical and random modes by solving **P3**. The system-wide utility also increases as peering degree increases. And hierarchical mode again gives higher service differentiation. The service differentiations in both modes are less than those in Fig. 2. This is because max-min fairness put strict priority on sending lower layers to all peers.

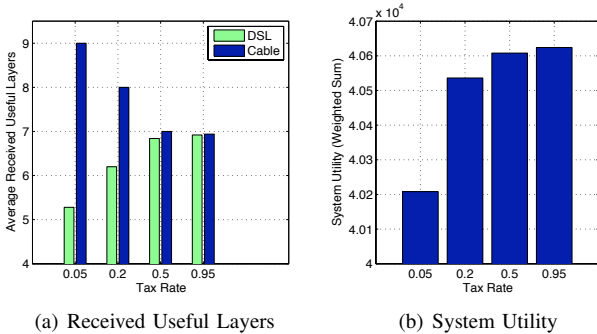
To investigate the impact of taxation, we study a numerical



**Fig. 2:** Weighted fairness in hierarchical and random topologies. (a) System-wide utility under hierarchical and random topologies with different peer connectivity; (b) Averaged received bitrate for heterogeneous peers under hierarchical topology with different peer connectivity; (c) Averaged received bitrate for heterogeneous peers under random topology with different peer connectivity.



**Fig. 3:** Max-min fairness in hierarchical and random topologies. (a) System-wide utility under hierarchical and random topologies with different peer connectivity; (b) Averaged received bitrate for heterogeneous peers under hierarchical topology with different peer connectivity; (c) Averaged received bitrate for heterogeneous peers under random topology with different peer connectivity.



**Fig. 4:** Impact of taxation on fairness and system utility. (a) Averaged received layers for heterogeneous peers under different tax rate; (b) Achieved system-wide utility under different tax rate.

example as shown in Fig. 4. The system has 15 cable peers with uploading capacity of 1000 Kbps and 25 DSL peers with 400 Kbps and they are hierarchically connected with degree 6. The video source is coded in 10 layers, each with 100 Kbps. The layer weight  $w^l$  is set as  $2^{(10-l)}$ . We vary the tax rate from 0.05 to 0.95. As can be seen from Fig. 4(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, system-wide utility increases with tax rate.

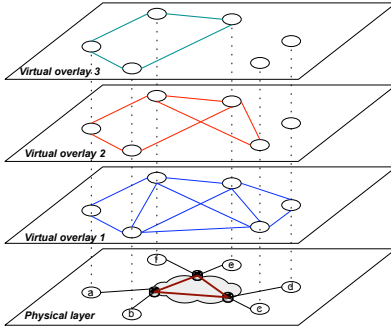
From the numerical results, we obtained the following guidelines for taxation-based P2P layered streaming design. 1) When the tax rate is small and thus the system is geared towards high service differentiation, hierarchical topology is preferred. 2) When the tax rate increases and the resulting system operates on a high utilization point, a more random topology is preferred. 3) Receiving entitled layers should be guaranteed for all peers and the uplink bandwidth should not be devoted to excess layers unless there is no more request for entitled layers.

### III. LAYERED P2P STREAMING PROTOCOL DESIGN

While the analytical models allow us to understand the trade-offs in 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*.

As shown in Fig. 5, multiple virtual streaming overlays, one for each SVC video layer, are formed among participating peers. Due to the dependency among video layers, upper streaming overlays must have fewer peers than lower overlays. A peer uses layer subscription scheme to determine how many layers to subscribe to. 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.



**Fig. 5:** Taxation based SVC P2P live streaming: virtual overlays and bandwidth allocation.

In the theoretic framework developed in Section II, network coding is adopted 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 [28]. 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 [32] 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.

#### A. 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 entitled 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

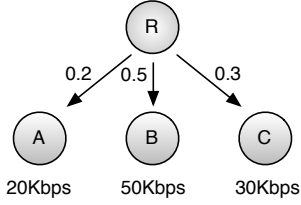
subscriptions. The algorithm allows the system to approach the utility maximization under taxation, and adapt to peer churn and network dynamics nicely.

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 [33], 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 timer 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  $l_i = \max(l_i - 1, L_i)$ .

#### B. 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. In the following, we present the peer chunk requesting and chunk serving algorithms.

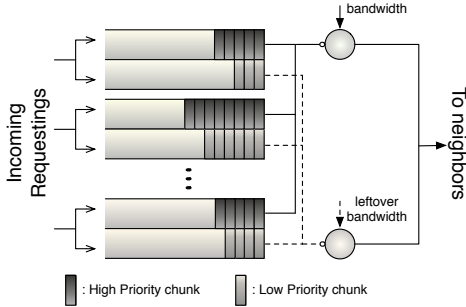
1) *Chunk requesting*: In SVC coded video, lower layer bit-stream is more important than higher layer bit-stream. Hence in principle, lower layer chunks should be requested before higher layer chunks. In order to increase the data chunk diversity and improve the chance that two peers always have chunks 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 that peer. For example, if requester R serves neighbors A, B and C with 20Kbps, 50Kbps



**Fig. 6:** Peer requests missing chunk from neighbor. Requester R sends the request to A, B or C with probability 0.2, 0.5 and 0.3 respectively.

and 30Kbps respectively, it then sends the chunk request to A, B and C with probability 0.2, 0.5, and 0.3 respectively, as shown in Fig. 6.

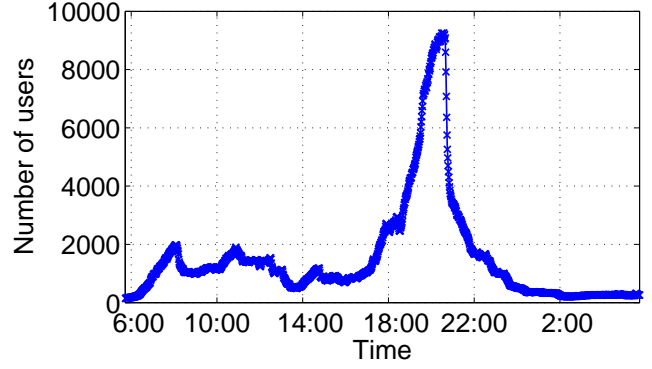
2) *Chunk serving:* Chunk serving is more sophisticated. Individual peers maintain two FIFO queues for each neighbor (see Fig. 7). One queue is called entitled queue and the other is called excess queue. Entitled queue holds chunk requests for entitled layers, while excess queue holds chunk requests 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.



**Fig. 7:** Peer serves neighbors. Low priority chunks will be served only if high priority queues are empty.

### C. Mesh Topology Adaptation

As discussed in Section II, hierarchical mesh topology is more favorable than random topology in providing differentiated services while random mesh is better for maximizing the system-wide utility. In this section, we consider how to efficiently adapt the mesh topology to achieve different design goals. 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 described below to ensure the overlay topology converge to the desired topology.



**Fig. 9:** PPlive Online Users

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  $\sum_{l \in i's \text{ entitled layers}} c_j^l w_l$ . In addition, the layer weights  $w$  are set 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 philosophy behind this design 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.

Simulation results in Section IV indicate that the mesh topology will converge to desired structures. At a small tax rate, which emphasizing differentiated services, hierarchical topology is achieved. At a large tax rate, which optimizing the system-wide utility, more random topology is realized.

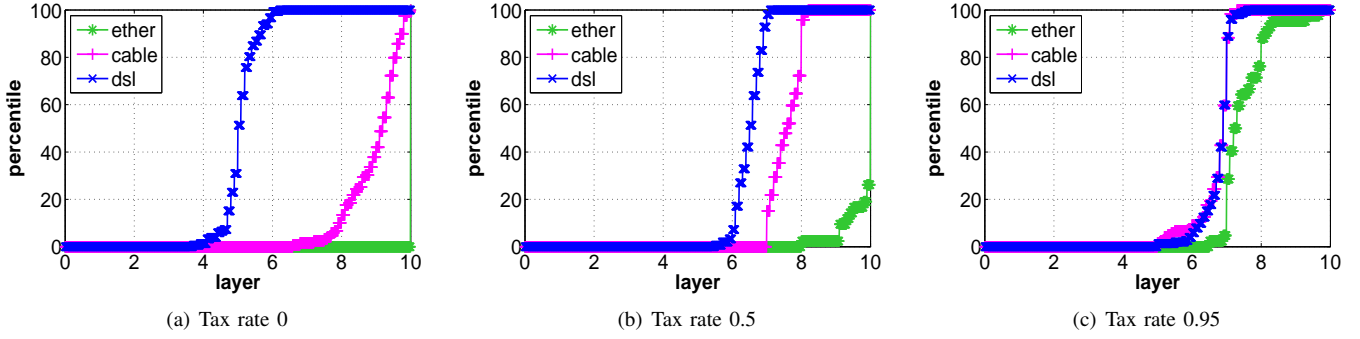
## IV. PERFORMANCE EVALUATION

We conduct extensive trace-driven simulations to evaluate the performance of the proposed taxation-based P2P layered streaming design. Specifically, we investigate the following issues: (1) the effectiveness of taxation-based incentive mechanism; (2) peer uplink bandwidth utilization; (3) the mesh topology adaptation; (4) user/peer perceived video quality; and (5) the convergence and optimality of AIAD layer subscription.

### A. Simulation Setup

A flow-level event-driven simulator is developed in C++. Unless stated otherwise, the simulations are driven by a trace





**Fig. 8:** Received video layers for different types of peers under various tax rates. (a) Cumulative distribution of received layers for different peers under tax rate 0; (b) Cumulative distribution of received layers for different peers under tax rate 0.5; (c) Cumulative distribution of received layers for different peers under tax rate 0.95.

collected from the measurement study of PPlive [11], a real-world P2P live streaming system. The trace was collected from Nov 22nd 17:43, 2006 to Nov 23rd 17:43, 2006. All peer arrivals and departures are recorded during this time. More than 100,000 participants are observed and the number of concurrent peers varies from 100 to more than 9,000. Fig. 9 shows the evolution of the number of concurrent peers.

The video is encoded into ten layers with layer rate of 100 Kbps. A ten layers SVC coded video can be created by combining hierarchical B structure [34] coding and coarse-grain scalability (CGS) [14] coding with acceptable overhead. More layers give the framework larger operational region and more room to adjust the balance between social welfare and individual welfare. More layers also impose more challenges on the underlying P2P system to properly handle the layer subscription, chunk scheduling, and overlay adaption.

There are three types of peers: DSL peers (400 Kbps), Cable peers (800 Kbps) and Ethernet peers (1500 Kbps). The fraction of individual peer types and their respective uplink bandwidths are summarized in Table III. In our simulation, there is one video server with upload capacity of 10 Mbps.

**TABLE III:** Peer upload bandwidth distribution

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

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.

## B. Simulation Results

1) *Effectiveness of Taxation Based Incentive Mechanism:* To reduce the randomness introduced by short-lived peers, only peers with life time greater than one minute are counted

in this experiment. 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. 8, 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.

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 IV lists the uplink bandwidth utilization (UBU) and the wasted bandwidth ratio (WBR). Overall, the uplink bandwidth utilization is consistently over 90%, indicating that the protocol can efficiently 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.

**TABLE IV:** System Bandwidth Utilization

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

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.

3) *Mesh Overlay Adaptation*: Mesh overlay topology plays a key role in service differentiation. Table V and VI 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 type of 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. Numerical results in Section II-D suggest that random mesh topology is better at achieving high social welfare. Our mesh topology adaption scheme is able to reflect this requirement. 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 III.

4) *Smoothness of Received Video Quality*: Video quality is related to the number of received video layers. In addition, viewing quality is affected by the variations of the received video layers over time. Quality of experience (QoE) is degraded if the number of received video layers changes frequently. We define the following *smoothness index* to quantify the playback smoothness of the received video.

$$SI = \frac{1}{K} \sum_{k=0}^K \frac{|v(k) - v(k-1)|}{v(k)}, \quad (9)$$

where  $v(k)$  is the received decodable layers at time period  $k$ , and  $K$  is peer's total number of online time period. The time period is one second. Large smoothness index indicates bad viewing quality caused by constant layer increasing/dropping. Fig. 10 shows the CDFs of smoothness index under different tax rates. Under all scenarios, peers contributing more enjoy smoother video playback. We also observed that as tax rate increases, the smoothness indexes for bandwidth-rich peers increase, while the smoothness indexes for bandwidth-poor peers decrease. We suspect this is caused by the peering topology at different tax rates.

As discussed in Section IV-B3, a hierarchical topology is formed at tax rate 0. DSL peers are mainly connected with other DSL peers and have to actively look for bandwidth resources, which causes more layer changes. In contrast, Ethernet and Cable users are mainly connected with each other, and have abundant bandwidth within the cluster. Thus fewer layer changes. As tax rate increases, the overlay topology becomes more randomized. Different peers have equal/similar access to bandwidth, leads to similar smoothness index.

5) *Layer Subscription Convergence*: In order to examine the behavior of the layer subscription algorithm without the impact of peer churn, a static topology with 500 peers is used

**TABLE V:** 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 VI:** 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%

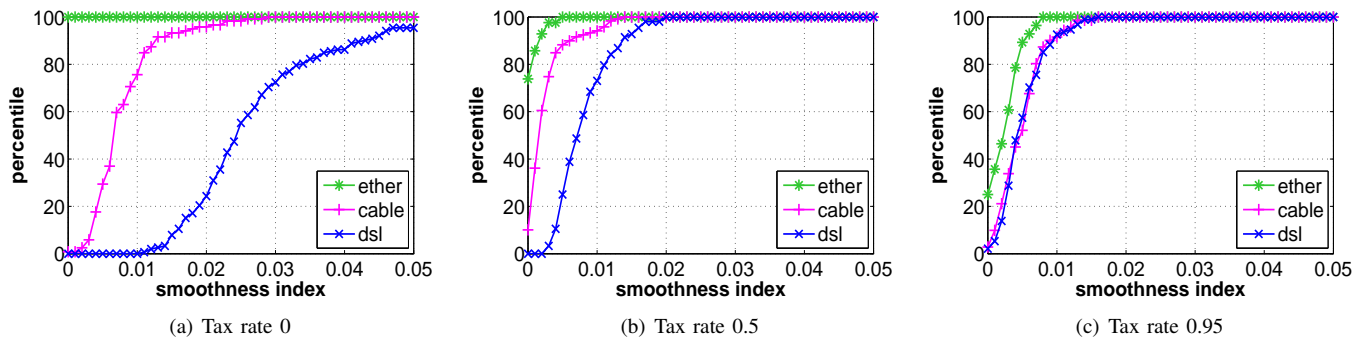
in this experiment. The peers' uplink bandwidth follows the distribution as stated in Table III. We randomly pick one peer from each bandwidth category and plot the evolution of its layer subscription. We also vary the tax rate to examine its impact. Fig. 11 shows the layer subscription process over time with tax rate of 0, 0.5, and 0.95, respectively. With tax rate of zero, peers are entirely selfish. The 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 the Ethernet peers (800 Kbps). Since video is encoded at 100 Kbps per layer, there are more "free" bandwidth in this case, introducing minor oscillations in layer subscription. In all cases, AIAD algorithm is able to quickly converge to the target subscription layer and peers stay in their optimal layers for most of the time.

## V. CONCLUSIONS

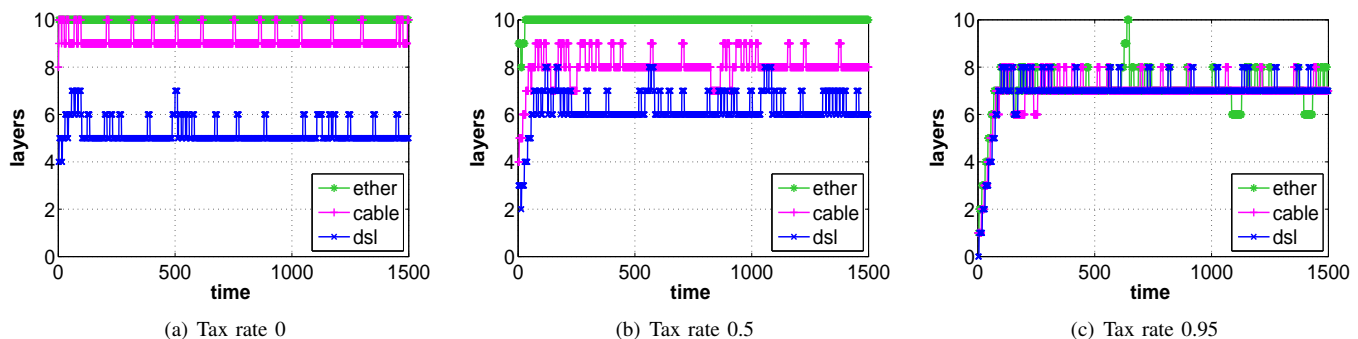
Designing an efficient P2P live streaming system that is fair to all peers and offers strong incentive for them to contribute is challenging. In this paper, we develop utility maximization models to understand the interplay between efficiency, fairness and incentive in layered P2P streaming. The models enable us to numerically investigate the impact of peering strategies and chunk scheduling policies on the fundamental trade-offs between the above three factors. We further 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.

## REFERENCES

- [1] "PPLive, <http://www.pplive.com/>." [Online]. Available: <http://www.pplive.com/>
- [2] "PPStream, <http://www.ppstream.com/>." [Online]. Available: <http://www.ppstream.com/>



**Fig. 10:** Smoothness index for different types of peers under various tax rates. (a) Cumulative distribution of smoothness index for different peers under tax rate 0; (b) Cumulative distribution of smoothness index for different peers under tax rate 0.5; (c) Cumulative distribution of smoothness index for different peers under tax rate 0.95.



**Fig. 11:** Layer subscriptions over time for different types of 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.

- [3] "UUsee, <http://www.uusee.com/>." [Online]. Available: <http://www.uusee.com/>
- [4] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, "LION: Layered Overlay Multicast with Network Coding," *IEEE Trans. Multimedia*, vol. 8, pp. 1021–1032, Oct. 2006.
- [5] P. Decuetos and K. Ross, "Unified framework for optimal video streaming," in *Infocom 2004*, Hong Kong, February 2004.
- [6] 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.
- [7] Mea Wang and Baochun Li, "R2: Random push with random network coding in live peer-to-peer streaming," *IEEE J. Select. Areas Commun.*, vol. 25, pp. 1655–1666, Dec. 2007.
- [8] N. Magharei and R. Rejaie, "PRIME: Peer-to-Peer Receiver-driven MESH-based Streaming," in *Proc. IEEE INFOCOM*, May 2007.
- [9] M. Zhang, Q. Zhang, and S. Yang, "Understanding the Power of Pull-based Streaming Protocol: Can We Do Better?" *IEEE J. Select. Areas Commun.*, vol. 25, pp. 1678–1694, Dec. 2007.
- [10] 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.
- [11] X. Hei, C. Liang, Y. Liu, and K. W. Ross, "Insights into PPLive: A measurement study of a large-scale P2P IPTV system," in *Workshop on Internet Protocol TV (IPTV) services over World Wide Web*, Edinburgh, Scotland, May 2006.
- [12] S. Xie, G. Y. Keung, and B. Li, "A measurement of a large-scale peer-to-peer live video streaming system," in *Packet Video*, Oct. 2007.
- [13] S. Ali, A. Mathur, and H. Zhang, "Measurement of Commercial Peer-To-Peer Live Video Streaming," in *First Workshop on Recent Advances in Peer-to-Peer Streaming*, Aug. 2006.
- [14] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Trans. Circuit and System for Video Technology*, vol. 17, pp. 1103–1120, Sep. 2007.
- [15] M. Wien, H. Schwarz, and T. Oelbaum, "Performance Analysis of SVC," *IEEE Trans. Circuit and System for Video Technology*, vol. 17, pp. 1194–1203, Sep. 2007.
- [16] 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.
- [17] Y.-W. Sung, M. Bishop, and S. Rao, "Enabling Contribution Awareness in an Overlay Broadcasting System," in *SIGCOMM*, 2006.
- [18] Y. Wang, A. Reibman, and S. Lin, "Multiple Description Coding for Video Delivery," *Proceedings of the IEEE*, vol. 93, Jan. 2005.
- [19] 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.
- [20] B. Cohen, "Incentives build robustness in bittorrent," in *Proceedings of the 1st Workshop on Economics of Peer-to-Peer systems*, 2003.
- [21] P. K. Hoong and H. Matsuo, "Palms: A reliable and incentive-based p2p live media streaming system," *Lecture Notes in Electrical Engineering*, pp. 51–66, 2008.
- [22] A. Habib and J. Chuang, "Service differentiated peer selection: An incentive mechanism for peer-to-peer media streaming," *IEEE Trans. on Multimedia*, vol. 8, pp. 610–621, 2006.
- [23] Z. Liu, Y. Shen, S. S. Panwar, K. W. Ross, and Y. Wang, "Using Layered Video to Provide Incentives in P2P Live Streaming," in *Sigcomm 2007 workshop on peer-to-peer streaming and IP-TV*, 2007.
- [24] N. Magharei, R. Rejaie, and Y. Guo, "Mesh or multiple-tree: A comparative study of live P2P streaming approaches," in *Proc. IEEE INFOCOM*, May 2007.
- [25] M. Chen, M. Ponc, S. Sengupta, J. Li, and P. A. Chou, "Utility Maxi-

- mization in Peer-to-peer Systems,” in *Proceeding of ACM SIGMETRICS*, June 2008.
- [26] R. Kumar, Y. Liu, and K. Ross, “Stochastic Fluid Theory for P2P Streaming Systems,” in *Proc. IEEE INFOCOM*, May 2007.
  - [27] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” *IEEE Trans. on Information Theory*, vol. 46, pp. 1204–1216, 2000.
  - [28] 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.
  - [29] F. Kelly, A. Maulloo, and D. Tan, “Rate control in communication networks: shadow prices proportional fairness and stability,” *Journal of the Operational Research Society*, 1998.
  - [30] D. Bertsekas and R. Gallager, *Data Networks*. Upper Saddle River, NJ: Prentice Hall Inc., 1992.
  - [31] “Ampl, <http://www.ampl.com/>.” [Online]. Available: <http://www.ampl.com/>
  - [32] J. Li, P. A. Chou, and C. Zhang, “Mutualcast: An Efficient Mechanism for Content Distribution in a P2P Network,” in *Sigcomm Asia Workshop*, 2005.
  - [33] 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.
  - [34] H. Schwarz, D. Marpe, and T. Wiegand, “Hierarchical B pictures,” *Joint Video Team Doc. JVT-P014*, July 2005.