# Inter-Overlay Cooperation in High-Bandwidth Overlay Multicast

Guang Tan and Stephen A. Jarvis
Department of Computer Science, University of Warwick,
Coventry, CV4 7AL, United Kingdom
{gtan,saj}@dcs.warwick.ac.uk

## Abstract

*The cooperation of end users can be exploited to boost the performance of high-bandwidth multicast. While* intra-overlay cooperation, *the mechanism for cooperation within a single overlay (multicast group), has been extensively studied, little attention has been paid to* inter-overlay cooperation. *In this paper we explore the possibility and effects of cooperation among co-existing heterogeneous overlays in the context of live media streaming, where bandwidth is the bottleneck resource. To motivate such a kind of cooperation, we design a reputation-based incentive mechanism that differentiates user' streaming qualities based on the amount of data actually forwarded by individual users. This not only stimulates users to contribute as much forwarding bandwidth as possible, but also motivates those with spare bandwidths in resource-rich overlays to find downstream users in external, often resource-poor, overlays so as to accumulate more reputation scores. Under this mechanism, an adaptive bandwidth exporting/reclaiming algorithm is developed which allows users to dynamically allocate bandwidth according to the resource availability of multiple overlays. Simulation results are reported with enhanced system performance in terms of users' average media quality.*

## 1 Introduction

Overlay multicast has emerged as an effective paradigm to provide large-scale data dissemination over the Internet. Using this approach, some high-bandwidth multicast applications such as live media streaming have been successfully deployed on the Internet. A dominant factor affecting the performance of such applications is the bandwidth contribution from individual users, and as a result the behavior of contributing and the mechanism to encourage such behaviors have been actively studied in recent years. For instance, Chu et al. [4] first consider altruism as a key element of P2P streaming broadcast. They show that the level of altruism has very important impact on the overlay; even a small degree of altruism brings significant benefits to overall system performance. Empirical studies, however, show that in practise altruism alone often does not ensure sufficient bandwidth supply, especially for high-bandwidth video streaming, as evident from the large proportion of *free-riders* on today's Internet [12] [14].

To tackle this problem, some incentive mechanisms have been designed to stimulate individual contributions. A commonly used approach is to relate the bandwidth contribution with the ability to choose good upstream nodes, which imply a good media quality. In [6], for example, a score-based incentive mechanism is proposed in which the contribution level of a user is represented by a score. High-score peers are offered more flexibility in choosing desired data suppliers, while low-score peers have limited, if any, options in parent selection, and hence receive low quality stream. This topic has attracted a good deal of attention from the research community.

To date, the study on user cooperations has been mostly limited to the case of single overlays. A question yet to be answered is: *In an overlay network where users are strategic and act in their own interest, is it possible or worthwhile to extend the cooperation to multiple co-existing overlays?* In this paper we address this issue. Our study is motivated by the following two observations:

1. Co-existing multiple overlays are often heterogenous in terms of media bit rate, type of end-users and bandwidth distribution, which potentially lead to great diversity in bandwidth supply. For example, an 800Kbps stream may impose a higher bandwidth requirement than can be offered by average end-users, whereas in an overlay with a 300Kbps stream users may easily find their bandwidth being underutilized. Another cause for the heterogeneity is the different types of users. For example, an academic event may attract an audience mainly from colleges or research institutes, which have abundant bandwidth resources, while the major audience of a soap opera may be ordinary Cable Moderm/ADSL users which have a much lower average bandwidth. Finally, due to the great variance of in-

dividual users' bandwidths [13] [12], multiple overlays can have large gaps in bandwidth supplies during a relatively short period of time even if they have identical bandwidth distributions and streaming rates. Therefore, it can be anticipated that if the spare bandwidth in the resource-rich overlays could be "exported" to those resource-poor overlays, the overall system performance can be improved.

2. The media content delivered over multiple overlays (or channels) are often owned by the same publisher (e.g., PPLive [20]), which has an inherent motivation to promote the cooperation among different overlays as long as this helps improve the overall performance of the streaming system, as this usually translates to audience rating, advertising revenue and market share. Furthermore, it is convenient for the publisher to control the way users participate in the system. In other words, it can design the game rules and enforce them via, for example, proprietary software.

In order to foster the inter-overlay cooperation, we first propose a reputation-based incentive mechanism. A node scores by forwarding a certain amount of data (e.g., 500K Bytes) to others, and nodes with higher scores are entitled to preempt the (better) tree positions of other leaf nodes with lower scores, so that the media quality can be improved. This way nodes will try to make full use of their bandwidths, and those with spare bandwidths in a resource-rich overlay are naturally motivated to find child nodes in other resource-poor overlays in order to accumulate higher scores. Nodes' reputation scores are maintained in a decentralized, secure and fault-tolerant fashion using a *referee node* mechanism based on a distributed hash table (DHT) [15] that interconnects all the overlays.

Given the incentive rules, we then examine the strategies that individuals may use to maximize their own benefits in terms of reputation scores. Specifically, we design an adaptive bandwidth exporting/reclaiming algorithm to help nodes allocate their bandwidths between local and external overlays in response to the changing circumstances of overlay resources. This algorithm monitors the utilization of local bandwidth and increases/decreases the exported bandwidth slots dynamically, giving priority to local overlays. Our simulation results show the effectiveness of the proposed incentive mechanism.

To facilitate a fine-grained bandwidth allocation, we adopt a multiple-tree mode [2] for an overlay. That is, in each overlay the stream is evenly divided into multiple independent substreams, and bandwidth exporting and reclaiming are performed on the substream basis.

The remainder of this paper is organized as follows. Section 2 documents related work; Section 3 establishes the architecture that interconnects the multiple co-existing overlays; Section 4 details the reputation-based incentive mechanism; Section 5 presents the bandwidth exporting/reclaiming algorithm; Section 6 presents the experimental results and Section 7 concludes the paper.

## 2 Related Work

A variety of models have been proposed for encouraging individual contributions by relating contribution with service quality. In [3], Chu et al. propose a taxation model, in which resource-rich peers are required to contribute more bandwidth to the system, and subsidize the resource-poor peers. In a payment-based [18] system, peers try to buy the best possible service from peers providing service at a minimum price, while those service providers strategically decide their respective prices in a *pricing game*, in order to maximize their economic revenues in the long run. In a reputation system [5], peers earn reputation by sharing, and highly reputed peers are more likely to obtain better service than peers with a low reputation.

Our incentive mechanism differs from previous work in two main aspects. First, it extends the cooperation from a single overlay to multiple overlays; second, the reputation scores are maintained in a completely distributed manner, which is in contrast with all the above schemes.

In the field of network routing, some efforts have been devoted to address the interaction between simultaneously co-existing overlays. This issue is identified as a "grand challenge" in networking research by a recent US-NSF-sponsored workshop report [1]. Kwon et al. [11] present *Synergy*, an overlay internetworking architecture for allowing autonomous, possibly heterogeneous, overlay networks to collaborate. The architecture aims at supporting transparent interactions among different overlays to improve performance and promote information sharing. Jiang et al. [7] investigate the overlay optimal routing among multiple overlays in a game theoretic approach and devise some pricing schemes to resolve the fairness anomalies of resource allocation caused by the optimal routing strategies. Finally, the authors in [9] study the dynamic interaction between overlay routing and Traffic Engineering (TE) using a *non-cooperative non-zero two player game* model. These studies are mainly focused on the routing performance and the sharing of underlying network resources.

To the best of our knowledge, the only published work that involves both inter-overlay cooperation and media streaming is by Liao et al. [8]. In their system, called *AnySee*, the resource shared among different overlays has been extensively used in optimizing streaming paths, reducing service delay and enhancing streaming reliability. However, this work does not explicitly consider the motivation behind such a kind of cooperation, and the optimization emphasis is not on "high bandwidth", which leads to completely different issues and strategies from our work.

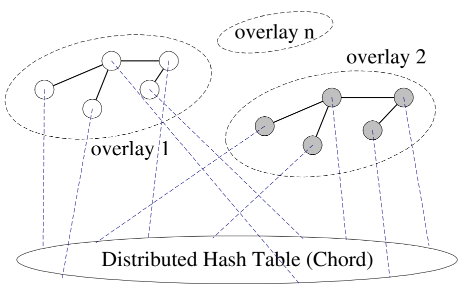


Figure 1: The multiple overlays interconnected by a DHT. Each node joins a main overlay that consists of several multicast trees for all the substreams (only one is shown in the figure for clarity). At the same time it belongs to the global DHT.

## 3 Interconnecting the Multiple Overlays

To enable the interaction of multiple independent overlays, we use a DHT to organize all the users across different overlays into a uniform space. DHT is a technique that assigns a set of keys to participating nodes, and can efficiently route messages to the unique owner of any given key in a distributed environment like the Internet. As shown in Figure 1, in addition to the main (or local) overlay it belongs to, each node is mapped to a unique ID in an identifier ring as produced by Chord [15], a representative form of DHT. The DHT serves two purposes. First, it allows a participating node to discover the members of other overlays in a convenient way. Since each node's ID in the DHT is generated independently of its overlay ownership, the $O(\log n)$ (where $n$ is the number of all nodes) neighbors of a node link itself to many external overlays. The other purpose of the DHT is that its routing function enables a node to efficiently find the *referee* nodes of another node. The referee node plays an important role in the incentive mechanism which will be discussed in the next section.

## 4 The Incentive Mechanism

Our incentive mechanism uses the amount of data that has been successfully received by all the child nodes of a node to represent its *reputation* (in the form of a score). The score can then be used to compete for good parents in a preemptive manner. To prevent cheating and collusion behavior, a node's reputation score is maintained by a set of randomly chosen score *referee nodes* instead of the node itself. The design is detailed in the following sections.

### 4.1 Keeping Score

Assuming a node has a key $k$ in a Chord ring with $2^m$ identifiers, then the $r$ referee nodes are the owners (successor nodes) of keys $(k + j \cdot 2^m/r) \mod 2^m, (j = 0, 1, \cdots, r-1)$. While receiving data from some parent, a node connects to the nearest referee (in terms of network latency) of that parent using the routing functionality provided by the DHT. Upon receiving a certain amount of data (e.g., 500 K Bytes), the node reports this amount to the connected referee of its parent, which then increases the score of the parent by one, and synchronizes with other referees on this record. (Again it can reach the other referees by DHT routing.) The parent periodically requests a digitally signed certificate of score from one of the referees (e.g. the nearest one), and can show it when preempting for some substream with another node. The receiver of the score certificate can verify it on the referee node with some probability.

Each node also maintains heartbeat connections with all its referees. If it detects that one of its referees, say $r_0$, fails, the node can use the failure recovery procedure of the Chord protocol to find a replacement for $r_0$ in the Chord ring. This new referee node is then instructed to synchronize with other referees on the reputation scores of all relevant nodes. During the reparation and synchronization periods, $r_0$ cannot be used to verify a node's score information. In this case the other referees can be enquired of the score of a certain node. Note that in an environment like the Internet, the score information maintained by the multiple referees need not be strictly consistent since perfect accuracy is not essential for this mechanism. In addition, the difference of a certain score value is upper bounded by a synchronization interval, which can be controlled to be small as compared to the accumulative score value of a node.

If the score certificate provided by a node is found to be false, the referee nodes will publicize (by, for example, a lightweight flooding in which a node probabilistically sends the information to its neighbors) the ID/address of the cheating node to the network. As a result the cheating node will find it being refused by most nodes when looking for parents and hence receive a very poor service quality. Even if the node can rejoin the network using a new identity, it can obtain no benefits because the ID-associated reputation score starts from zero, which means it is unable to preempt other nodes' parents. The proposed referee node selection method also makes collusion difficult; since the referees are not designated by a node itself, but chosen by the underlying DHT. In addition, under the random identifier generation function, the probability of two general nodes being referees for each other is very small, meaning that there is little chance for two nodes to cheat on behalf of each other.

### 4.2 Parent Preemption based on Score

When a newly arriving node enters a multicast tree, it selects the nearest node with spare forwarding bandwidth[1]

[1]For convenience of discussion, we sometimes discretize the bandwidth into bandwidth slots (or slots for short), each slot representing the bandwidth of a substream.

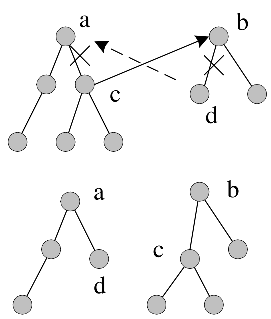


Figure 2: An example of parent preemption.

among its neighbors (we assume it has already obtained a set of neighbors) as its parent for each substream. When it leaves, the children nodes need to re-find parents. These basic processes are the same as the methods proposed in previous studies [13].

The unique feature of our tree maintenance algorithm lies in the parent preemption mechanism based on reputation scores. That is, a node with a high score can preempt the parent of a node with a lower score in order to reduce end-to-end latency and optimize its streaming quality. In this paper the streaming quality is quantified as follows:

$$Q = \log(1 + L_{max}^{-\alpha} \cdot N_p), \tag{1}$$

where $L_{max} = \max\{L_0, L_1, \cdots, L_S\}$ is the maximum end-to-end latency of all the $S$ substreams, and $N_p$ the number of substreams received by one node. The $\log$ function captures the diminishing returns of an increased media bit rate on the perceived media quality, and the $\alpha$ parameter controls the impact of end-to-end latency on the streaming quality. In a streaming broadcast application without user interaction, the $\alpha$ can be fairly small (e.g., 0.1). Note that $Q$ can also be defined in many other forms. This has no fundamental effect on our parent preemption algorithm. For convenience of comparison, we often use the metric *relative quality* instead, which is defined as the ratio of $Q$ to the maximum quality value obtained when all substreams are directly transmitted from the root through the physical network. That is,

$$Q' = \frac{Q}{\log(1 + L^{-\alpha} \cdot S)}, \tag{2}$$

where $L$ is the distance of a node to the root in the physical network.

In the algorithm, a node always tries to optimize the substream with the largest latency (the latency of a missing substream is defined to be infinity) at one time, as shown in Figure 2. The preempted nodes are restricted to be leaf nodes in order to avoid unstable tree adjustment and oscillation of nodes' tree positions. In practise, the leaf level of a multicast tree generally contains a large proportion of nodes, which provide plenty of options for a node in optimizing the streaming quality. When a node finds a parent it can preempt, it joins the parent along with the subtree under it, and the preempted leaf node is forced to find a new parent. The parent of the preempting node, if one exists, can be an option for the preempted node in search for a new parent. The algorithm is executed by a node periodically.

### 4.3 Fault Tolerance of the Referee Nodes

If all the referees of a node fail simultaneously, the reputation score of a node will be unverifiable and thus it can only accumulate the reputation score from scratch. This undesirable situation can be avoided by increasing the referee set size $r$. However, a large $r$ means a high control complexity and communication overhead. This tradeoff can be quantitatively analyzed given some reasonable assumptions: (1) the network has a large number of nodes and has evolved for a long time; (2) all nodes have the same lifetime distribution $F(t)$ with finite mean $\mu$ and variance $\sigma$; (3) the joining time of a certain node is uniformly random within the lifetime intervals of all its referees, and (4) the referee selection is independent of a node's age. If we let $E_s$ denote the mean time for a failed referee being replaced and synchronized by another one, then the following theorem holds (see [16] for the proof).

**Theorem 1.** *In a stationary state, the probability of a node finding all of its referees in reparation/synchronization state (i.e. the state of being replaced) at any given point of time is given by*

$$\pi = \left( \frac{2\mu E_s}{\sigma^2 + \mu^2 + 2\mu E_s} \right)^r.$$

Suppose, for example, that $E_s$ is 120 seconds, and $F(t)$ is a lognormal distribution $F(t) = \Phi\left(\frac{\ln t - 5.08}{2.03}\right)$ [17], where $\Phi$ is the standard normal distribution function, then $\pi < 3 \times 10^{-8}$ for $r = 3$, which indicates that the probability of one node's score information being unverifiable is negligible with a referee set size of only 3.

## 5 Bandwidth Exporting/Reclaiming

Under the proposed incentive mechanism, nodes are stimulated to earn as many scores as possible in order to obtain the best media quality. There are two factors that determine a node's capability of earning scores: the number of forwarding bandwidth slots and its utilization. Given a certain number of bandwidth slots, a node in a resource-rich overlay may not be able to find enough downstream nodes to fully utilize them, hence the reputation score is not maximized. A natural way to improve this situation is to *export* some spare bandwidth slots to external resource-poor overlays. As shown in Figure 3(a), a fraction of a node's

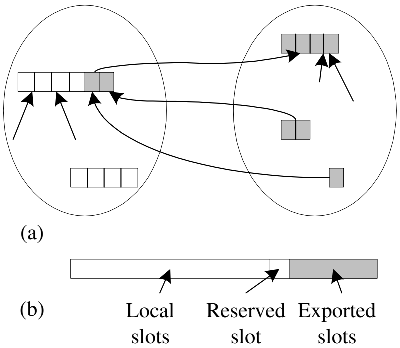


Figure 3: Illustration of bandwidth exporting/importing. (a) A node in the left overlay exports two slots to the right overlay. (b) The bandwidth allocation of a node.

bandwidth slots can be allocated to nodes in other overlays if these slots are unlikely to be used in the main overlay in the near future. To adapt to the changing network environment, the number of exported slots may be increased or decreased dynamically. These operations are taken care of by the bandwidth export/reclaim algorithm, which is executed on every node periodically, and mainly involves the following operations.

**Monitoring/allocating local bandwidth** Each node monitors the number of used local slots $B_{used}$ periodically and maintains a history record for this, and let variable $B_{used}^{max}$ store the maximum number of used slots during a past time window. If $B_{idle} = B_{total} - B_{used}$ is more than 2, then $B_{exported} = B_{idle} - 1$ bandwidth slots can be exported. Here the algorithm does not export all the idle slots, but reserve a slot for local overlay in order to meet possibly increasing demand for bandwidth from the local overlay. This is equivalent to giving priority to local overlay in bandwidth allocation. Another condition of the exporting is $B_{exported} > 1$, because to export slots to an external overlay, the current node must first *import* at least one substream of that overlay[2]. The condition $B_{exported} > 1$ ensures that the bandwidth contribution to the external overlay is positive. An example of bandwidth slot allocation is illustrated in Figure 3(b).

**Identifying external overlays to export bandwidth** After determining the number of exported bandwidth slots, a node needs to identify an external overlay to export these slots. As introduced before, it is easy for a node to discover a set of external overlays. The node then chooses an overlay that is most short of bandwidth based on a metric called *bandwidth availability index* (BAI). The BAI is defined as

$$BAI = \frac{\sum_{i=1}^{N} B_i}{N \times S}, \tag{3}$$

where $N$ is the number of nodes in the overlay, $B_i$ the number of outgoing bandwidth slots of node $i$ and $S$ the number of substreams. Clearly, a high BAI means that a node has more parents to choose on average, and a BAI of 1 is the lower bound for an overlay to export bandwidth. The node estimates the BAI of an overlay by collecting the bandwidth information of a certain number of nodes in that overlay, either by direct inquiry or using second-hand information. It then chooses the overlay with the smallest BAI as the overlay to assist.

Before the exported bandwidth slots can be used, a node needs to *import* some substreams from that overlay. Because this will consume existing bandwidth resource, and considering that all substreams are independent and symmetric, the algorithm chooses to import only one randomly chosen substream. This way the exported slots can only be used by nodes requesting that particular substream. To attract nodes from external overlays to be its children, a node may propagate the bandwidth exporting information to that overlay via its DHT neighbor nodes.

**Reclaiming bandwidth** As time passes, the BAI of an overlay may change, and a node exporting bandwidth may find the BAI of its main overlay decrease towards 1. In this case the node needs to reclaim some exported slots. The decision of reclaiming, however, does not depend on the calculation of BAI, but on local bandwidth utilization. If a node finds that the reserved slot has ever been used during the past time window, it treats this as a signal of increased bandwidth demand on itself, and then tries to reclaim an exported slot. It checks the exported slots to see if there is an unused slot; if so, it simply moves this slot from the exported slots group to the local slots group, otherwise it disconnects an external node with the lowest score and then reclaims that slot. A structured description of the above procedures is given in Algorithm 1.

## 6 Performance Evaluation

An event-driven simulator has been developed to study the performance of the proposed schemes. The GT-ITM transit-stub model [19] is used to generate an underlying network topology consisting of 15600 nodes. Link delays between two transit nodes, transit nodes and stub nodes, and two stub nodes are chosen uniformly between $[15, 25]$ ms, $[5, 9]$ ms and $[2, 4]$ ms, respectively. Of all the 15360 stub nodes, a subset of nodes are randomly selected to participate in a certain number of overlays with configured probabilities. The arrival rate of nodes are controlled so that

[2] Since the bottleneck resource is forward bandwidth, we assume a node always has enough incoming bandwidth to receive a full stream of its local overlay plus an extra substream from an external overlay.

**Algorithm 1** Bandwidth exporting/reclaiming algorithm

```
Calculate $B_{used}^{max}$ for the past time window
$B_{idle} \leftarrow B_{total} - B_{used}^{max}$
if $B_{idle} == B_{exported} + 1$ then
  /* normal case, no change on the allocation*/
else if $B_{idle} > B_{exported} + 1$ then
  /*to export more slots */
  calculate the BAIs of up to 5 external overlays
  $m \leftarrow$ the external overlay with the smallest BAI
  if BAI of this overlay > 1 and BAI of overlay $m < 1$ and $B_{idle} > 2$ then
    if $B_{exported} > 0$ or this node has successfully imported an external substream from overlay $m$ then
      $B_{exported} \leftarrow B_{idle} - 1$
    end if
  end if
else if $B_{idle} == B_{exported}$ then
  /* Reserved slot maybe occupied, need to reclaim */
  if $B_{exported} > 0$ then
    if $B_{exported}$ has been fully used then
      $c \leftarrow$ the external child with smallest score
      force $c$ to re-find parent
    end if
    $B_{exported} \leftarrow B_{exported} - 1$
  end if
end if
```

the maximum number of all active nodes are no more than 6000.

For each overlay, the location for the root node is fixed at a randomly chosen stub node. The root is assumed to support 10 full streams. Other nodes' outgoing bandwidths follow a Bounded Pareto distribution. The nodes' lifetimes follow a lognormal distribution [17] [14]. The parameters of the lognormal distribution are chosen according to the statistical findings in [17] so that the mean lifetime is 1809 seconds. Nodes enter the system in a Poisson process, and the arrival rate is determined from 6000 divided by the mean lifetime. By default, each node executes the bandwidth exporting/reclaiming algorithm and the parent preemption algorithm every 30 seconds.

Two primary metrics to be used in the performance evaluation are the relative media quality as defined in Eq.(2) and the resource availability index as defined in Eq.(3).

We consider two experiment scenarios. In the first scenario, there are three equal-sized overlays, $O_1, O_2$ and $O_3$, whose streaming rates and average node bandwidths are given in Table 1. Each substream is assumed to be 50Kbps. Through this simple configuration we show the effectiveness of the proposed methods and the effects of some parameters.

In the second scenario we add more heterogeneity and dynamics to the network. There are 10 overlays, $O_1, O_2, \cdots, O_{10}$, and nodes join $O_i (i = 1, 2, \cdots, 10)$ with probability $p_i = 0.045 + 0.01i$ so that the average size of overlay $i$ is $330 + 60i$. For each overlay, the nodes' bandwidths follow a bounded Pareto distribution whose parameters are changed every hour so that the mean is uniformly random between $[240, 580]$Kbps. The streaming rate is fixed at 400Kbps and each substream is 50Kbps. This imaginary scenario is intended to be a test for the schemes' ability to adapt to a complex network environment. In this setting, we are interested in the average- and worst- case performance of the whole system.

| Scenario | Overlay | Stream rate (Kbps) | Avg. Bw. (Kbps) | Avg. size | Avg. BAI |
|---|---|---|---|---|---|
| 1 | 1 | 400 | 700 | 2000 | 1.75 |
| | 2 | 400 | 550 | 2000 | 1.37 |
| | 3 | 800 | 400 | 2000 | 0.50 |
| 2 | 1,2,...,10 | 400 | Changing between [240,580] | 330, 390, ... 870 | Changing |

Table 1: Parameter setting in the experimental scenarios.

### 6.1 Effects of the Incentive Mechanism

We make snapshots of the nodes' streaming qualities and bandwidth with and without the incentive mechanism after the network enters a steady state. The nodes are divided into two classes: the young class in which nodes' ages are smaller than 400 seconds, and the old class containing the other nodes. For the young class, no correlation can be found between the bandwidth and quality (the figures are omitted due to the space limitation); while for the old class, the incentive mechanism leads to a noticeable correlation between these two properties. As shown by Figure 6, with the incentive (no bandwidth exporting), the nodes with large bandwidths are more likely to have high qualities. Indeed, the correlation is found to be more pronounced if the we divide the age into more intervals and compare nodes in each interval. These results reflect a desirable characteristic of the incentive mechanism, that is, a node's streaming quality does not depend on a single metric such as bandwidth or time, but on the combination of these metrics which give a more accurate measurement of the "effective contribution" made by a node to the network.

Figure 5 shows the same snapshot of the old class with the bandwidth exporting/reclaiming algorithm enabled. It can be seen that correlation of bandwidth and quality is slightly stronger than found in the previous setting. This is because the high-bandwidth nodes can make better use of their bandwidths by exporting their spare bandwidths to those needy overlays.

Figure 6 shows the average quality of overlay 3 changing over a time period of about 3 hours. A major observation is that the incentive mechanism helps improve the overlay performance. The reason behind this is that the parent preemption algorithm potentially arranges the high-bandwidth nodes higher positions in the tree than those low-bandwidth nodes, thus leading to a wider and shorter tree, which means

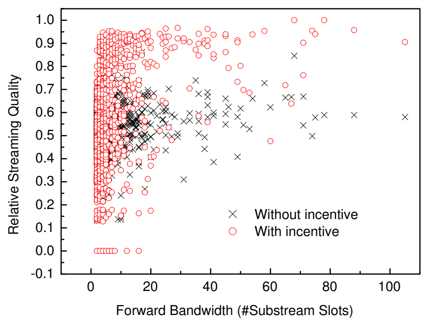


Figure 4: Incentive without BER.

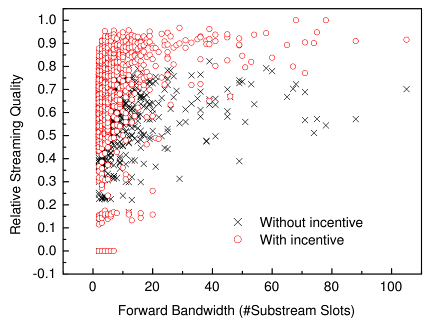


Figure 5: Incentive with BER.

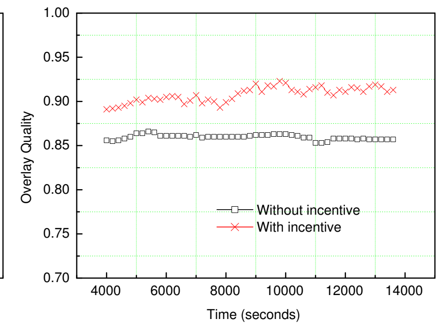


Figure 6: Effect of incentive on average quality.

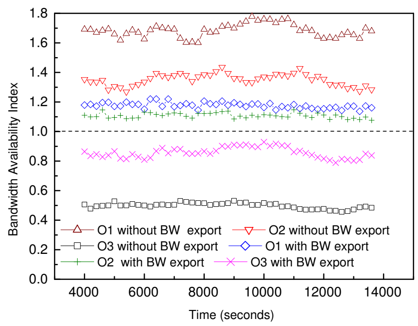


Figure 7: Effect of BER on BAIs.

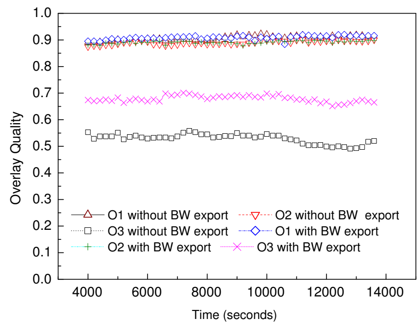


Figure 8: Effect of BER on overlay quality.

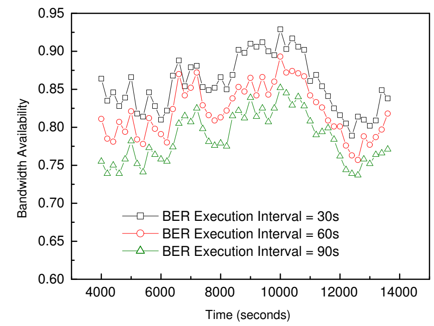


Figure 9: Effect of BER execution interval on BAIs.

a smaller average service latency.

### 6.2 Effects of Bandwidth Exporting/Reclaiming

Figure 7 shows the BAIs of the three overlays changing over time with and without the inter-cooperation. Without bandwidth exporting, the BAIs of $O_1$, $O_2$ and $O_3$ are around 1.7, 1.4 and 0.5 respectively. After applying the bandwidth exporting/claiming algorithm, the BAIs change to around 1.2, 1.1 and 0.9 respectively. Figure 8 further shows the *overlay quality*, the average relative quality of all nodes of an overlay, as a function of time. we can see that the inter-overlay cooperation improves the quality of $O_3$ by nearly 30% while not affecting the performance of the other overlays. This shows that the bandwidth exporting algorithm effectively makes use of the spare bandwidths without interfering with the normal operations of the exporting overlays.

### 6.3 Effects of Some Parameters

Figure 9 compares the BAI of overlay 3 under different execution intervals of the bandwidth exporting/reclaiming algorithm. The smaller the interval, the more promptly a node reacts to the bandwidth demanding on itself. Generally, this means that a node can export the spare bandwidth more quickly when it is available, therefore the resource-poor overlays can benefit more from this. Figure 10 compares the BAI of overlay 3 under varying bandwidth monitoring window sizes. As introduced in Section 5, a node maintains a monitoring window for the utilization of its bandwidth, and the maximum number of used local bandwidth slots is used to decide the number of exported slots. As expected, a smaller window size leads to a higher BAI, since this means that a node can export its bandwidth more aggressively, thus benefiting more to the resource-poor overlay. It should be noted that both parameters should not be too small because this will bring more dynamics to the network and impose higher communication overheads on individual nodes.

### 6.4 Overall System Performance Results

We study the overall performance of the whole streaming system in the second scenario. Figure 11 and Figure 12 show the average and minimum values of all overlay qualities changing over time respectively. The average overlay quality measures the performance of all the overlays, while the minimum overlay quality reflects how the most resource-poor overlays benefits from the inter-overlay cooperation. It can be observed that the inter-overlay cooperation improves the system performance in both metrics, and the proposed algorithms adapt well to the changing situations of the network.

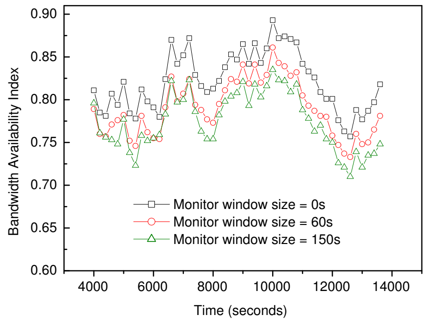


Figure 10: Effect of monitoring window size on BAIs.

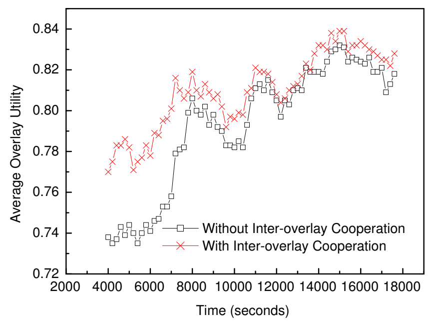


Figure 11: Average overlay quality (scenario 2).

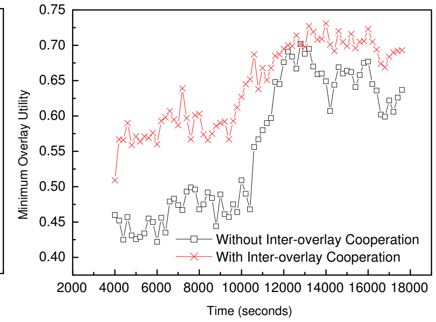


Figure 12: Minimum overlay quality (scenario 2).

## 7 Conclusion

This paper explores the possibility and effects of cooperation among co-existing heterogeneous overlays in the context of live media streaming. The contributions are twofold. First, to motivate cooperation both within a single overlay and across multiple overlays, we design an incentive mechanism that relates the expected media quality with the amount of data actually forwarded by individual users. This motivates those with unused bandwidths to find downstream users in resource-poor overlays so as to accumulate higher reputation scores. The second contribution is an adaptive bandwidth exporting/reclaiming algorithm that allows individual users to dynamically allocate bandwidth according to the resource availability of multiple overlays, while giving priority to its local overlay. Simulation results show the effectiveness of the proposed schemes.

## References


[1] M. Ammar and Co-authors. Report of the National Science Foundation Workshop on Fundamental Research in Networking. April 2003, http://www.cs.virginia.edu/ jorg/workshop1.

[2] M. Castro, P. Druschel, A-M. Kermarrec, A. Nandi, A. Rowstron and A. Singh. SplitStream: High-bandwidth multicast in a cooperative environment. *Proc. of SOSP'03*, 2003.

[3] Y. Chu, J. Chuang, and H. Zhang. A Case for Taxation in Peer-to-Peer Streaming Broadcast. *Proc. of ACM SIGCOMM Workshop on Practice and Theory of Incentives and Game Theory in Networked Systems (PINS)*. August, 2004.

[4] Y. Chu and H. Zhang. Considering Altruism in Peer-to-Peer Internet Streaming Broadcast. *ACM NOSSDAV*, June, 2004.

[5] M. Gupta, P. Judge and M. Ammar. A reputation system for peer-to-peer networks. *ACM NOSSDAV*, 2003.

[6] A. Habib and J. Chuang. Incentive Mechanism for Peer-to-Peer Media Streaming. *Proc. of IWQoS 2004*.

[7] W. Jiang, D. Chiu and J. Lui. On the Interaction of Multiple Overlay Routing. *Performance Evaluation*, 62(1-4), 2005

[8] X. Liao, H. Jin, Y. Liu, L. Ni and D. Deng. AnySee: Peer-to-Peer Live Streaming. *IEEE INFOCOM 2006*, April 2006.

[9] Y. Liu, H. Zhang, W. Gong, D. Towsley. On the interaction between overlay routing and underlay routing. *IEEE INFOCOM 2005*.

[10] V. G. Kulkarni. *Modeling and Analysis of Stochastic Systems*. Chapman & Hall Ltd. ISBN: 0-41204-991-0, 1996.

[11] M. Kwon and S. Fahmy. Synergy: An Overlay Internetworking Architecture. In *Proc. of the 14th IEEE International Conference on Computer Communications and Networks (ICCCN)*, 2005.

[12] S. Saroiu, P. Gummadi and S. Gribble. A Measurement Study of Peer-to-Peer File Sharing Systems. *Multimedia Computing and Networking (MMCN)*, 2002.

[13] K. Sripanidkulchai, A. Ganjam, B. Maggs and H. Zhang. The feasibility of supporting large-scale live streaming applications with dynamic application end-points. *Proc. of ACM SIGCOMM*, 2004, Portland, Oregon, USA.

[14] K. Sripanidkulchai, B. Maggs and H. Zhang. An analysis of live streaming workloads on the Internet. *4th ACM SIGCOMM IMC*, Oct., 2004.

[15] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications. *ACM SIGCOMM 2001*.

[16] G. Tan and S. Jarvis. Inter-Overlay Cooperation in High-Bandwidth Overlay Multicast. *TR-CS-06-010*. Dept. of CS, University of Warwick, UK.

[17] E. Veloso, V. Almeida, W. Meira, A. Bestavros, and S. Jin. A Hierarchical Characterization of A Live Streaming Media Workload. *IEEE/ACM Trans. on Networking*, 12(5), 2004.

[18] W. Wang and B. Li. Market-driven Bandwidth Allocation in Selfish Overlay Networks. *Proc. of INFOCOM 2005*.

[19] E. W. Zegura, K. Calvert and S. Bhattacharjee. How to Model an Internetwork. *IEEE INFOCOM '96*.

[20] PPLive Online TV Player. http://www.pplive.com.