Service Differentiation Based Incentive Mechanism for P2P Streaming in Hybrid Overlay Network

Suphakit Awiphan[†], Zhou Su[†] and Jiro Katto[†] (member)

Abstract In Peer-to-Peer (P2P) networks, an incentive mechanism is a necessary component to deal with the free-riding behavior. The challenge is that direct reciprocal incentives; e.g., tit-for-tat, which consider the cooperation of peers in a pair-wise manner, are not suited with P2P streaming. In this paper, we propose a new service differentiation mechanism to provide a redistribution incentive for P2P streaming in a hybrid overlay network. The contribution of a peer can be measured from the number of video sub-streams that it uploads to other peers. By sending one request message, the number of sub-streams that each peer can retrieve is varied by its contribution level. An altruistic peer thus has to send less request messages and will experience smoother video quality than a selfish peer. Through simulations, we demonstrate that our solution can provide service differentiation among peers with better streaming quality than the tit-for-tat scheme.

Keywords: Incentive Mechanism, Service Differentiation, P2P Streaming, Hybrid Overlay

1. Introduction

With the growth of residential Internet access in recent years, there has been significant interest in the use of peer-to-peer (P2P) technologies for streaming live media to users. The key factor making this approach attractive is that the P2P model is cost effective and more scalable than the client-server model. In P2P networks, the peers are expected to contribute their available resources, such as upload bandwidth, to the system in return for consuming the resources of other peers. While the performance of P2P networks relies on the voluntary resources sharing by each peer, there is an inherent tension between personal rationality and public welfare. Some studies have demonstrated that a great number of peers do not contribute to the network or other peers at an acceptable level¹⁾. This non-cooperative behavior, called 'free-riding', is known as a serious threat to P2P networks as it leads to performance degradation. Accordingly, to encourage the cooperation of peers and reduce the bottleneck at the suppliers, a simple yet effective incentive mechanism becomes necessary.

In P2P live streaming systems, the content is generally delivered among peers over multicast overlay

networks²⁾. The overlay architectures, which have been developed for P2P streaming, typically organize end-hosts into either a tree or a mesh topology. With the tree-based method, a peer forwards (pushes) received video data towards other peers in a tree-shaped overlay without a request requirement. With the mesh-based method, a peer establishes a partnership with other peers to form a mesh overlay. During a session, the peers periodically exchange buffer map with each other so that they can request (pull) the required video data blocks from their partners and provide available blocks to other partners appropriately.

To address the problems of free-riding in both tree and mesh overlays, researchers have proposed various incentive mechanisms³⁾⁻⁶⁾. Most of them apply a multiple description coding (MDC) technique with the tit-for-tat strategy of BitTorrent⁷⁾. A fundamental algorithm is that a peer has to exchange descriptions with another peer in a pair-wise fashion in order to obtain more descriptions. A peer who shares more descriptions thus experiences better video quality than a selfish peer. Nevertheless, some studies have verified that the tit-for-tat incentive mechanism is unsuitable for P2P streaming systems⁸⁾. Specifically, a cooperative peer often cannot transmit data in return to another peer from whom it is receiving data. This difficulty happens because data transmission in P2P streaming imposes more stringent real-time requirements than P2P file-sharing. The difference in

Received July 19, 2012; Revised September 17, 2012; Accepted November 21, 2012

[†] Graduate School of Science and Engineering, Waseda University
(Tokyo, Japan)

playback time makes the available data blocks of one peer unnecessary to another peer. Therefore, developing an incentive mechanism for P2P streaming systems without relying on the tit-for-tat scheme is a challenging work.

In addition, there are some drawbacks in the earlier tree and mesh approaches. Specifically, the tree suffers from node dynamics and is subject to frequent overlay reconstruction. The mesh typically has long transmission delay and a high control overhead⁹⁾. Recently, the focus has moved towards hybrid overlay networks in which the tree and mesh schemes are both implemented^{10)–12)}. The performance evaluations by some studies demonstrate the superior efficiency and robustness of the hybrid approaches compared to the pure tree and mesh based method¹¹⁾¹²⁾. Also, hybrid overlay networks require an efficient incentive mechanism similar to other overlay architectures. Nevertheless, the point is that both push and pull data delivery of the tree and the mesh method, respectively, are deployed in the hybrid overlay. This results in the technical challenges in designing an efficient incentive mechanism for hybrid overlay networks.

Recent researches on incentive mechanisms for P2P live streaming systems mainly focus on 1) the ease of implementation; 2) the current contribution of peers rather than their historical contribution; 3) the asymmetry among peers; 4) the robustness to malicious peers; and 5) the fully distributed system. These five issues remain challenges for researchers^{1)13)–16)}. Some researchers proposed new game-theoretic models¹³⁾¹⁴⁾ to address the mentioned issues. In the game-theoretic model, rational peers are strategic players who want to maximize their payoffs (e.g., the streaming quality) while reduce their costs (e.g., utilized upload bandwidth). In each time round, a peer decides how much it should contribute to achieve optimal utility. Moreover, there has been an interest to use an idea of social networks for building new incentive mechanisms¹⁴⁾¹⁵⁾. However, it seems that the social-based incentive is suited to P2P streaming systems where upload capacities of peers are sufficient to provide sustainable bit-rates¹⁴⁾. Furthermore, to the best of our knowledge, an incentive mechanism for hybrid overlay networks has not been well considered.

In this paper, we introduce a new service differentiation based incentive mechanism for P2P streaming in a hybrid overlay network. The idea is to control the number of video sub-streams which a partner

transmits to a requesting peer. By sending one request message for sub-streams, the altruistic peer, who contributes more number of sub-streams, has a greater chance to receive all of the requested sub-streams from a partner peer. On the other hand, the selfish peer has to send many request messages to get the required sub-streams. As a result, the altruistic peer can obtain smoother video quality compared to the selfish peer.

The key novelties of the proposed approach are: 1) the contribution level of a peer is determined by the number of uploaded sub-streams, not the quantity of transferred data. This reduces the complexity of calculations; 2) a hierarchical hybrid overlay is designed to manage the information exchange between peers so that they do not need to know every other peer's contribution value; 3) different from the direct reciprocal method, the proposed incentive mechanism does not restrict a peer to exchange data with another peer who is uploading data to it only (i.e., a peer can upload available sub-streams to any requesting peers in different sub-trees). So, the limitation of the direct reciprocity, such as tit-for-tat, could be resolved; and 4) the contribution value of a peer is maintained with a specific group of peers located on the same level of other sub-trees. This solution improves the contribution value distributions, which are mostly done in a random manner, of earlier approaches.

We evaluated the proposed incentive mechanism through ns2 simulations, which demonstrate that the streaming quality of each peer could be differentiated according to the number of sub-streams it uploads to other peers. The simulation results also show that our solution provides higher average streaming quality than the direct tit-for-tat incentive mechanism.

The remainder of this paper is organized as follows. In Section 2, we introduce the background and related works. Section 3 describes our proposal in detail and we then evaluate the performance of our proposal in Section 4. Finally, Section 5 concludes the paper and offers future research plans.

2. Related Works

2.1 Overview of Incentive Mechanisms

There have been several approaches which are proposed for promoting the collaboration of peers in P2P networks. In general, we can categorize these incentive mechanisms into three groups, namely, monetary-based, reputation-based, and reciprocity-based methods¹⁾. Firstly, the monetary-based approach is to charge peers for the service they receive. In this method, a central

authority is required to manage transactions and track balance of each user. However, due to the employment of virtual currency and some security issues (e.g., user identification, fraud payments, and vulnerability to the single-point-of-failure problem), the monetary-based incentive method turns out to be highly impractical.

Secondly, in the reputation-based approach, the reputation information of a peer is built corresponding to its long-term behavior. Peers with good reputation are offered better service quality. This approach can be further divided into two groups, which are autonomous reputation and global reputation. With the autonomous reputation scheme, peers locally keep reputation information of other peers they connect with (i.e., the reputation values are not disseminated). Since peers' identities have to be maintained across sessions to keep long-term histories, it seems difficult to implement this approach in anonymous systems. On the other hand, the global reputation scheme stores reputation information of several peers either within a group of peers or at a central server. In the former case, the control overhead from information exchanging becomes a major concern. In the latter case, even though it provides fast free-rider detection, the scalability and implementation in pure P2P networks are issues.

Lastly, the reciprocity-based approach can be considered to be a service differentiation scheme similar to the reputation-based approach. However, the reciprocity-based method typically measures peers' contributions only for the current session and maintains short-term histories. The reciprocity-based method can be based on direct reciprocity or indirect reciprocity. Specifically, with the direct reciprocity, a peer only considers a pair-wise data exchange with another peer. In contrast, with indirect reciprocity, a peer assesses other peers' cooperation level according to their contribution to the entire network¹⁾.

Among various incentive mechanisms, the reciprocity-based method seems to be the most practically feasible and effective solution. BitTorrent, one of the most popular P2P file-sharing systems, implements the direct reciprocal incentive, called 'tit-for-tat'. In particular, BitTorrent divides a file into a collection of blocks or chunks with equal importance. Before using the file, all data blocks have to be downloaded. To achieve this, the peers periodically exchange with each other the information of data blocks' availability; called a 'buffer map', and then request the missing data blocks. In order to provide a redistribution incentive, a peer transmits a

data block to another peer if the other transmits one in return. Hence, the peers are encouraged to contribute to the system because the more upload bandwidth they contribute, the faster the speed at which they can download the file⁷⁾.

2.2 Incentive Approaches for P2P Streaming

The authors of 3) and 4) proposed tit-for-tat incentives for P2P streaming in a multiple-tree overlay network by utilizing multiple descriptions coding (MDC). In particular, a source encodes a video into multiple descriptions and distributes each description over a separate tree. The peers located in different trees exchange descriptions with each other. A cooperative peer is able to receive more descriptions and obtain better playback quality than a selfish peer. This approach, however, restricts a peer to trade only the descriptions corresponding to the tree to which it belongs¹⁶⁾. Moreover, if the video is encoded into multiple layers, it will be more complex to manage over multiple trees¹⁰⁾.

In 5) and 6), the MDC and layered MDC techniques are applied with the tit-for-tat scheme in the mesh overlay for providing service differentiation. With the layered MDC approach, a video is coded into layers with different importance where each layer has to be received in a consecutive manner. The higher layer will refine the video quality generated by the lower layer. While the mesh overlay can accommodate node dynamics better, it usually suffers from delay and overheads due to its per-block data pulling operation⁹⁾¹⁰⁾.

Another important issue is that P2P streaming systems require a peer to receive and transmit data in a timely manner. As time elapses, the older data blocks become obsolete. This causes asymmetry among peers in media streaming systems. For example, assume that there is a peer $P1$ who is at playback point $t+i$ and a peer $P2$ who is at playback point t . The reciprocal data exchange between the two peers may not be able to achieve. This is because $P1$ can send data blocks to $P2$, but $P2$ may have no data blocks in which $P1$ is interested. Using the tit-for-tat method in P2P streaming thus limits some peers in contributing to the system or even receiving data from other peers⁸⁾.

Accordingly, the authors of 8) realize the limitations of employing tit-for-tat incentive in P2P streaming. They suggested that the collaboration of a peer could be measured based on the exchange of control information among peers rather than the quantity of data that the peer transmits. While their approach facilitates the peer

that cannot reciprocate in the exchange of data to collaborate in the system, the efficiency of their solution has not been demonstrated. In our previous studies¹⁷⁾, we proposed an overlay network to enforce peer's collaboration. While the system provided balanced bandwidth utilization among peers, there were two main drawbacks: 1) a high control overhead due to auxiliary connection constructions and maintenance; and 2) its vulnerability to peer dynamics, since only a push-based data delivery scheme was used.

Recently, there have been efforts to reconcile the tree and mesh methods in order to form a hybrid overlay network. CoolStreaming¹⁰⁾, an innovative P2P streaming system, adopts a hybrid of push/pull data delivery in its content delivery to address the drawbacks of previous systems. In CoolStreaming, when a peer subscribes to a sub-stream by sending a request (pulling) to one of its partners, the partners will continue sending (pushing) all blocks in need of that sub-stream to the requesting peer. This mechanism thus significantly reduces the overhead and delay caused by the data pulling operation. The authors of 16) proposed an incentive mechanism based on this form of sub-stream delivery. In their approach, two peers exchange a sub-stream with each other in a tit-for-tat fashion. While their approach is effective, the usage of tit-for-tat seems to be a concern as mentioned earlier.

Inspired by this observation, we have designed a service differentiation mechanism for video streaming in a hybrid overlay network. Our work differs from other approaches in that we do not use the direct tit-for-tat incentive (i.e., the reciprocal of data transmission does not require to perform in a pair-wise manner). We believe that our solution is suitable for the P2P streaming system and can be developed further for its implementation in a practical system.

3. Proposed Incentive Mechanism

In this section, we describe the design of the proposed incentive mechanism. We consider a P2P streaming system consisting of a single source with peers who rationally share their resources. In order to avoid confusion, we use the word "partners" for those peers who establish a relationship to exchange data availability information together, and the word "parent" for a peer who actually transfers video data to another peer, who is called a "child". The main notation used in this paper is summarized in **Table 1**.

Table 1 Main notation.

p	Client peer
q	Parent peer
r	Children peer
j	Partner peer
S	Video sub-stream; $S = (S_1, \dots, S_T)$
t	Sub-tree number; $t = (1, \dots, T)$
l	Level number
C_p	Contribution value of a peer p
G_p	Utility function of a peer p
U	Number of uploaded sub-streams
E	Number of sub-streams qualified to download
m_s	Number of request messages to download sub-stream s
α	Weight parameter

Fig.1 An organized view of the proposed overlay when the video is divided into $T=4$ sub-streams. Note that some connections are not shown in order to improve the clarity of the figure.

3.1 Hybrid Overlay and Data Delivery

The organized view of the overlay is shown by **Fig.1**. We construct the hybrid overlay in a mesh-over-tree fashion. The core overlay is conceived in a tree structure rooted at the source. In accordance with the degree of the source, the tree is divided into multiple T sub-trees, which facilitates the source to deliver video sub-streams. Peers in each sub-tree then form a mesh by establishing a partnership with other peers in different sub-trees. Accordingly, a peer has one permanent parent located in the same sub-tree, and maintains a group of partners in each different $T-1$ sub-tree.

We then group the peers according to their tree hops from the source and assign each peer to a level l , where $l = (1, 2, 3, \dots, L)$. A peer can calculate its level by increasing the level of a parent located in the same sub-tree by one. During a session, a list of active children of each peer is distributed to other peers in the same level. Consequently, each peer can obtain from its parent a list of active peers, which are located in the different sub-trees, to establish a partnership.

The level is used for managing partnership selection among peers in different sub-trees. In particular, we design that the peer can establish a partnership with other peers located on the same level (or an upper level

if there is no peer on the same level at that time) of different sub-trees only. This is because the peers located on the same level seem to have the same range of playback time. However, when the tree expands, the number of peers in the deep level becomes larger. Therefore, the maximum number of partners on the same level of each different sub-tree that a peer can maintain is limited also. With this configuration, the peers do not need to know every other peer's information in the overlay. Moreover, the size of a partner's information kept by each peer is small.

For data delivery, we implement the video stream division technique of 10) in our overlay. Specifically, the video stream is divided into blocks with an equal size at the source. Each block is assigned with a sequence number according to its playback order. Also, in order to reduce an interruption caused by parent departure or failure, the video stream is decomposed into multiple S sub-streams, denoted by $\{S_1, S_2, S_3, \dots, S_S\}$, which could be encoded by MDC.

In the proposed hybrid overlay, the peer receives one sub-stream, corresponding to the sub-tree to which it belongs, from the parent without a request requirement. For the remaining sub-streams, the peer can retrieve (pull) them from partners in any different sub-trees. For example, assume that there are 4 sub-streams, a peer located in sub-tree 1 receives sub-stream S_1 from its parent in the same sub-tree, and can request sub-streams S_2, S_3 , and S_4 from partners either in the 2nd, 3rd or 4th sub-tree.

The buffer map, which specifies the latest received data block of each sub-stream, helps the peer to choose to which partner it should send a request message. Note that the buffer map has S elements equal to the number of sub-streams. At the partner, after receiving the request and verifying the contribution amount of the requesting peer, it will continue transmitting all blocks related to that sub-stream to the requesting peer. With this data delivery scheme, the control overhead generated by data requesting can be reduced. Furthermore, since peers may receive video sub-streams from multiple parents simultaneously, the resilience to peer dynamics is improved¹⁰⁾.

In the following subsections, we describe our service differentiation based incentive mechanism. The method to measure the contribution level of a peer at the supplier side will be introduced in Section 3.2; then the mechanism for controlling the utility of a peer at the receiver side will be described in Section 3.3.

3.2 Contribution at Supplier Side

Most incentive mechanisms evaluate the collaboration of a peer based on the quantity of data that it uploads to other peers. Such an approach appears to require computational resources. As an alternative, we propose a method to evaluate the contribution level of a peer based on the number of uploaded sub-streams.

Assume that peer p is currently uploading data to other peers located in the same or upper level, then the contribution value of peer p is evaluated from the number of uploaded sub-streams to a child peer r , where $r = (1, \dots, R)$. Let U be the number of sub-streams that a peer is uploading. We then get the contribution function C of a peer p whereby:

$$C_p = \sum U_r, r = 1, \dots, R \quad (1)$$

Note that, in a case where a peer is altruistic, its contribution value may be higher than the number of sub-streams divided at the source.

To maintain the contribution values of peers, some approaches use a central server. However, the centralized system seems to be vulnerable to the single-point-of-failure problems and requires extra costs. While the distributed method is more effective, many approaches distribute the contribution value of each peer in a random fashion (i.e., the peers, who are responsible for keeping the peers' contribution value, are not specified). Therefore, we design a well-organized distribution method by utilizing the structure of our hybrid overlay. The algorithm is to distribute the contribution value of a peer, C_p , to a group of peers located on the same level of separate sub-trees only. **Fig.2** shows an example of contribution value distribution of peer p from a child r to the partners that are located on the same level.

It is also important to note that, we are not concerned with the past records of peers. The peer's contribution is a current value of the latest report. Once a new value of

Fig.2 A distribution of contribution value of peer p who is uploading sub-stream to child r , towards the partners located in the same level.

a peer's contribution is reported, its past contribution value will be erased¹³⁾. Hence, the peers have to contribute to the system constantly, in order to keep their contribution value high. Moreover, the load to store peers' contribution is very low.

3.3 Utility at Receiver Side

Service differentiation can be achieved by controlling the number of sub-streams that a partner peer j accepts to transmit to a peer p . Assume that the number of sub-streams that the partner j accepts to transfer to the requesting peer p is $E_{j,p}$, where E is a positive integer from 1 to S . We designate that:

$$E_{j,p} \leq \alpha C_p \quad (2)$$

where α is a positive integer used for weighting the contribution amount. The value of α determines how much the peers should contribute to the system in order to maximize their own utility. For instance, if we assume that $\alpha = 1$, then by sending one request message, the maximum number of sub-streams that a peer may download from a partner is equal to the number of sub-streams that it is currently uploading to other peers. In a case where $\alpha > 1$, the ratio between the number of sub-streams that a peer should upload and the number of sub-streams that it can download will be greater (i.e., a peer has to share their resources more to achieve the maximum utility).

Let J be the number of partners of a peer p . We characterize the utility function G of a peer p as follows:

$$G_p = \sum_j^J E_{j,p} \quad (3)$$

In our approach, sub-streams are assumed to have equal importance. If a peer receives all sub-streams, it can reconstruct the video flawlessly, otherwise the peer will experience discontinuous video playback. Using our service differentiation based incentive mechanism, a peer with a high contribution value has a high probability of getting the required sub-streams by sending a lower number of request messages, resulting in smooth video playback quality. In contrast, a selfish peer has to send more request messages to different partners in order to get the required sub-streams.

We let m_s denote the number of request messages to download a sub-stream S . The optimal sub-stream requesting can be written as follows:

$$\min \sum_{S=S_1}^{S_S} m_s \quad (4)$$

Fig.3 and **Fig.4** demonstrate the operations of a

Fig.3 An example operation of the proposed incentive mechanism in the case of an altruistic peer who is uploading three sub-streams.

Fig.4 An example operation of the proposed incentive mechanism in the case of an uncooperative peer who is uploading only one sub-stream.

proposal in the case of a contributor and an uncooperative peer, respectively. In this example, we assume that there are 4 video sub-streams. A peer p has 4 partners. One of them, denoted by child r , is receiving U sub-streams from peer p . Another one, denoted by parent q , is transmitting a sub-stream to peer p after it receives the request. From Fig.3, the peer p contributes to the system by sending three sub-streams. When the peer p requests sub-streams S_3 and S_4 , it can receive both sub-streams. This is because parent q knows that the contribution value C_p is three. On the other hand, in Fig.4, the peer p is uncooperative. It uploads only one sub-stream. Therefore, when it requests for sub-streams S_3 and S_4 , it can only receive either S_3 or S_4 . Also, it has to send more request messages to receive the required sub-streams.

3.4 Handling Collusion Attacks

With the indirect reciprocity-based incentive mechanism, the proposal could resolve the limitation of the direct reciprocity-based approach in P2P streaming systems. However, it is worth to note that the indirect reciprocity-based approach is vulnerable to collusion attacks¹⁸⁾. Specifically, a group of colluding free riders can attempt to report to other peers that they contribute

to each other without actually transferring data among themselves. Thus, the colluding free riders can evade the detection of free-riding and gain the high quality of service. Such collusion attacks occur frequently in P2P systems in which the indirect reciprocity-based incentive mechanism is used. Though, it appears that direct reciprocity-based approaches (e.g., tit-for-tat) are not vulnerable to this kind of collusion attack due to its pairwise mechanism.

In literature [18], the authors demonstrate an efficient solution against collusion attacks^[18]. The approach is to verify that data transmission and contribution actually occur between peers. The authors have proposed two kinds of peers, namely, provers and verifiers. Provers are untrusted peers claiming to have data and upload them to other peers. Verifiers are peers who obtain the list of provers and verify if data transmission actually occurs between those peers. The verifier can achieve this by simultaneously presenting "puzzles" (i.e., the hash of the file bits indexed by a randomly chosen file indices) to the provers. Then, the verifier records the durations required by each prover to report its solution of the puzzle. Any peer whose solution is incorrect or solution time is greater than a threshold is suspected to be a malicious peer. We believe that integrating this solution with our proposal will result in more robust incentive mechanism. As our future work, we will improve the robustness of the proposed incentive mechanism against collusion attacks.

4. Performance Evaluations

To evaluate the performance of our proposal, we have conducted simulations through the ns2.28 simulator^[19]. We focused on "received quality" which is defined by the ratio between the number of received video data blocks and the number of encoded video data blocks during a streaming session. The packet losses could happen due to bottleneck at the network, bottleneck at the selfish peers, and long playout delay. For comparison, we implement the system with a direct reciprocal incentive (tit-for-tat) where a peer transmits a sub-stream to another peer if the other transmits one in return. Also, the system without an incentive mechanism is included in the comparisons.

We simulate a P2P streaming system with 50 to 300 peers. The underlying topology is generated by the GT-ITM tool^[20], where an Internet-like topology^[21] is used. There are transit-domains and stub routers in the generated topology. The source and client peers are

randomly connected to one of the stub routers, while the transit-domains act as the routers. On top of the link-layer topology, the proposed hybrid overlay is built for the media streaming.

The content streaming rate from the source is set to 1 Mbps. The bandwidth, link latency, and degree of a source are 10 Mbps, 20 ms, and 4, respectively. The video is divided into 4 sub-streams. A weight of the contribution amount, α is 1.

We assume that there are three groups of users having different (download/upload) bandwidth of connections; 1) FTTH users with 100/100 Mbps links; 2) VDSL users with 100/50 Mbps links; and 3) ADSL users with 24/1 Mbps links. The link latency of each client peer varies from 20 to 100 ms by Pareto distribution. The peer joins the system at a random time between 1 to 15 seconds assigned by Uniform Random Number Generation. The simulation length is 300 seconds.

4.1 CDF of Received Quality

We verify the cumulative distribution function (CDF) of the received quality in the system that employs our service differentiation mechanism. We assume a scenario where the average shared upload bandwidth is not high enough to support the maximum playback rate for all peers (i.e., the average upload bandwidth in the system is about 530 Kbps, while the streaming rate is 1 Mbps). The shared upload bandwidth of each peer is set according to the following criteria:

- 20% of the peers have 100/100 Mbps connections, but upload 5 sub-streams as a maximum. That is, the peers only utilize about 1,200 Kbps of their upload bandwidth.
- 50% of the peers have 100/50 Mbps connections, but upload less than 3 sub-streams. That is, only approximately 512 Kbps of their upload bandwidth can be utilized.
- The remaining 30% of the peers have 24/1 Mbps connections uploading less than 2 sub-streams, which is about 256 Kbps of the upload bandwidth.

Fig.5 shows the cumulative distribution of the received quality for different shared upload bandwidths. We can observe that the received quality of peers varies according to the shared upload bandwidth (i.e., the number of uploaded sub-streams). The peers, who share 512 and 256 Kbps of upload bandwidth, receive less than 80% of the data blocks encoded at the source. This is because these low contribution peers have to send more request messages in order to receive all of the sub-streams. Moreover, their requests are likely to be

Fig.5 Cumulative distribution of the received quality of peers in the system with the proposed incentive mechanism under different shared upload bandwidths.

Fig.6 Cumulative distribution of the received quality of peers in the system with the proposed incentive mechanism under the same connection bandwidth but with different numbers of uploaded sub-streams.

rejected because the available degrees have been occupied for sending sub-streams to the contributors. For the peers whose upload contribution is about 1,200 Kbps, the received quality is greater than 75%.

We then verify that the received quality is essentially differentiated based on the peers' contributions. We setup the second scenario where all of the peers have 24/1 Mbps links, but 30% of them contribute more than 4 sub-streams, 50% of them contribute less than 4 sub-streams, and 20% of them are selfish peers, contributing less than 3 sub-streams.

Fig.6 demonstrates the cumulative distribution of the received quality under the homogeneous links but with different contribution amounts. We can see that the received quality is differentiated from approximately 30% to 100%. The result of each contribution amount is similar to the heterogeneous connection bandwidth case. In particular, the peers who contribute less upload bandwidth (i.e., the number of uploaded sub-streams) receive a lower number of data blocks, resulting in discontinuous video quality. This evaluation result confirms that the peers' received quality is related to the contribution amount, not available link bandwidth.

In addition, we evaluate the second scenario with a

Fig.7 Cumulative distribution of the received quality of peers in the system with the proposed incentive mechanism with different numbers of peers.

different number of peers. **Fig.7** shows the CDF of the received quality when there are 50, 100, 200, and 300 peers in the system with the proposed service differentiation mechanism. We can see that there is no significant difference when the number of peers is changed. The peers' received quality can be properly differentiated from 20% to 100%. When the number of peers is 50, 20% of the peers can obtain the video quality at the full rate. However, when the number of peers is increased, the received quality drops slightly. We believe that when the number of peers becomes large, each peer has a greater choice of partners from which to request the data blocks. As a result, a small portion of peers may not be requested for data blocks. This results in their low contribution values.

4.2 Comparison with Other Methods

We also compare our method with the tit-for-tat based incentive mechanism and the system without an incentive mechanism. The peers' bandwidth and contribution amounts are heterogeneous similar to the first scenario of the previous subsection. We show the cumulative distribution of the received quality of all comparisons in **Fig.8**.

It can be observed that in the system that uses tit-for-tat strategy, the received quality is obviously lower than with our method. This is because the tit-for-tat strategy assesses the contribution value in a pair-wise fashion whereas the asymmetry among peers is not well considered. Therefore, it is possible that a peer cannot transmit data blocks in return to its partner if the playback time of two peers is very different. This result verifies that the tit-for-tat strategy, which has been used in many incentive mechanisms for P2P streaming, is not suited to the P2P streaming systems.

For the system without an incentive mechanism, the received quality is very close to our method. However, a further evaluation of the received quality for each type

Fig.8 Cumulative distribution of the received quality of peers under different incentive methods.

Fig.9 Cumulative distribution of the received quality of cooperative peers and selfish peers compared between the system with the proposal and the system without an incentive mechanism.

of peers reveals that there is unfairness among the peers in the system with no incentive mechanism. **Fig.9** displays the received quality of the contributors and the selfish peers compared between the system without incentive and the system with our proposal. We can see that in the system without incentive, the received quality is not well-differentiated. Both contributors and selfish peers receive the same range of quality. In this case, the selfish peers may receive high quality even if they contribute less to the system. On the other hand, the contributors may receive low quality due to the bottleneck created by the selfish peers. This eventually discourages those peers to contribute their resources, since there is no reward for their collaboration. Finally, in the system that employs the proposed incentive mechanism, we can differentiate the received quality to the contributors and the selfish peers. The peers are encouraged to contribute to the system accordingly.

In **Fig.10**, we demonstrate the control overheads of the system without incentive and the system with the proposed incentive. The overhead is the total bytes of control packets generated to maintain the overlay. More specifically, in both systems, the control overheads consist of join, leave, and sub-stream request messages. In the system with the proposed incentive, there is

Fig.10 Total control overhead.

Table 2 Comparison between control overhead and amount of video data sent from a source.

Bit-rate (kb/s)	Control overhead (kB)		Amount of video data sent from a source (kB)
	Without incentive	With the proposed incentive	
512	89	672	≈ 158,745
1024	91	670	≈ 375,000
2048	91	671	≈ 745,975

additional contribution notification message which is generated when a peer report a new value of another peer's contribution. The experiment from **Fig.10** shows that the system with proposed incentive mechanism has greater control overhead than the system without incentive. This is the fundamental trade-off between redistribution incentive and overhead. According to the result in **Table 2**, control overheads of both systems become very small relatively against the amount of video data sent from a source, such that the generated overhead does not adversely affect the performance of the system.

4.3 Impact of Peer Churn

In this subsection, we have conducted simulations to evaluate the performance of the systems under a high rate of peer churn. We set the total simulation time to 500 seconds, and then setup two scenarios, namely, "without peer churn" and "with peer churn".

In the first scenario where peer churn does not exist, we assume that 300 peers randomly join the overlay during 1 to 50 seconds of the simulation time. After that, all peers stay throughout the session. In the second scenario, we assume a severe peer churn scenario where 300 peers simultaneously join the overlay during the first 15 seconds of the simulation time. We then assign the lifetime of peers using Uniform Random Number Generation (minimum = 20 seconds, maximum = 50 seconds). A peer will leave the overlay after its lifetime expires, and then rejoin the overlay again with 5 seconds interval, such that a peer joins and leaves the overlay frequently.

Fig.11 Cumulative distribution of the received quality of peers under a scenario without peer churn and a scenario with a severe peer churn.

From **Fig.11**, we can see that a severe peer churn does not affect the system with proposed incentive much. In particular, the cumulative distribution of the received quality of the system under a severe peer churn is lower than that of the system without peer churn. This slight decrement of overall received quality is due to frequent joining and leaving of peers. However, the cumulative distribution of the received quality of the system without peer churn still closes to that of the system with a severe peer churn. Also, the received quality is well differentiated. We think that peer churn does not affect our proposal much because we consider instant peers' contribution rather than their histories. Therefore, peers have to share available data continuously even their parent or neighbors change.

5. Conclusion

In this paper, we introduce a novel service differentiation based incentive mechanism for P2P streaming in a hybrid overlay network. Our work focuses on the number of sub-streams that a peer uploads. The contribution amount can be evaluated from the number of uploaded sub-streams rather than the quantity of the transferred data. The designed hybrid overlay also facilitates the distribution of contribution value.

The simulation results confirm the benefits of our proposed strategy and encourage us to develop the incentive mechanism further. In future work, we will consider more complex case of data delivery where each video sub-stream has different importance.

[References]

- 1) M. Karakaya, I. Korpeoglu and O. Ulusoy: "Free riding in Peer-to-Peer Network", *IEEE Internet Computing Mag.*, **13**, 12, pp.92-98 (Mar-Apr. 2009)
- 2) D. Pendarakis, S. Shi, D. Verma, and, M. Waldvogel: "ALMI: An Application Level Multicast Infrastructure", *Proc. USENIX USITS 2001*, pp.49-60 (Mar. 2001)
- 3) J.D. Mol, D.H.P. Eperma and H.J. Sips: "The Orchard Algorithm: Building Multicast Trees for P2P Video Multicasting without Free-riding", *IEEE Trans. on Multimedia*, **9**, 8, pp.1593-1604 (Dec. 2007)
- 4) D. Manzato and N. Fonseca: "An Incentive Mechanism for Peer-to-Peer Networks with Live Streaming", *Proc. IEEE Globecom 2006*, pp.1-6 (Dec. 2006)
- 5) Z. Liu, Y. Shen, S. Panwar, K. Ross and Y. Wang: "P2P Video Live Streaming with MDC: Providing Incentives for Redistribution", *Proc. IEEE ICME 2007*, pp.48-51 (July 2007)
- 6) Z. Liu, Y. Shen, S. Panwar, K. Ross and Y. Wang: "Using Layered Video to Provide Incentives in P2P Live Streaming", *Proc. ACM P2P-TV 2007*, pp.311-316 (Aug. 2007)
- 7) B. Cohen: "Incentives Build Robustness in BitTorrent", *Workshop on Economics of Peer-to-Peer Systems*, Berkeley (June 2003)
- 8) T. Silverston, O. Fourmaux and J. Crowcroft: "Towards an Incentive Mechanism for Peer-to-Peer Multimedia Live Streaming Systems", *Proc. IEEE P2P*, pp.125-128 (Sep. 2008)
- 9) N. Magharei, R. Rejaie and Y. Guo: "Mesh or Multiple-Tree: A Comparative Study of Live P2P Streaming Approaches", *Proc. IEEE INFOCOM 2007*, pp.1424 - 1432 (May 2007)
- 10) B. Li, et al: "Inside the New CoolStreaming: Principles, Measurements and Performance Implications", *Proc. IEEE INFOCOM 2008*, pp.1031-1039 (Apr. 2008)
- 11) M. Zhou and J. Liu: "A Hybrid Overlay Network for Video-on-demand", *Proc. IEEE ICC 2005*, pp.1309-1313 (May 2005)
- 12) F. Wang, Y. Xiong and J. Liu: "mTreebone: A Hybrid Tree/Mesh Overlay for Application-layer Live Video Multicast", *Proc. ICDCS 2007*, pp.49 (Sep. 2007)
- 13) X. Xiao, Q. Zhang, Y. Shi and Y. Gao: "How Much to Share: A Repeated Game Model for Peer-to-Peer Streaming under Service Differentiation Incentives", *IEEE Trans. Parallel Distrib.* **23**, 2, pp.288-295 (Feb. 2012)
- 14) T.M. Wang, W.T. Lee, T.Y. Wu, H.W. Wei and Y.S. Lin: "New P2P Sharing Incentive Mechanism based on Social Network and Game Theory", *Proc. WAINA 2012*, pp.915-919 (Mar. 2012)
- 15) O. Abboud, T. Zinner, E. Lidanski, K. Pussep and R. Steinmetz: "StreamSocial: A P2P Streaming with Social Incentives", *Proc. IEEE WOWMOM 2010*, pp.1-2 (June 2010)
- 16) Z. Liu, Y. Shen, K. Ross, S. Panwar and Y. Wang: "Substream Trading: Towards an Open P2P Live Streaming System", *Proc. IEEE ICNP 2008*, pp.94-103 (Oct. 2008)
- 17) S. Awiphan, Z. Su and J. Katto: "A Contribution-Aware Multiple Parent Overlay Network for P2P Media Streaming", *Proc. Packet Video Workshop 2010*, pp.118-125 (Dec. 2010)
- 18) M.K. Reiter, V. Sekar and Z. Zhang: "Making Contribution-Aware P2P Systems Robust to Collusion Attacks Using Bandwidth Puzzles", *Technical Report CMU-CS-09-136*, Carnegie Mellon University (2009)
- 19) The network simulator ns2, <http://www.isi.edu/nsnam/ns>
- 20) Modeling topology of large internetworks, <http://www.cc.gatech.edu/projects/gtlim>
- 21) O. Heckmann, M. Piringer, J. Schmitt and R. Steinmetz: "On Realistic Network Topologies for Simulation", *Proc. ACM SIGCOMM Workshop on MoMeTools*, pp.28-32 (2003)

Suphakit Awiphan received the B.Eng. degree in computer engineering from Chiang Mai University, Chiang Mai, Thailand in 2005, and the M.Eng. degree in computer engineering from Waseda University, Tokyo, Japan in 2009. He received the Royal Gold Medal award from the Engineering Institute of Thailand under H.M. the King's Patronage in 2004. Since 2006, he has been a scholar of Royal Thai Government. His current research interests include the P2P Networks, Application-layer Multicast and Content Centric Networking. He is currently working towards the Ph.D. degree in the Graduate School of Science and Engineering, Waseda University. He is a student member of IEEE, IPSJ, and IEICE.

Zhou Su received the B.E. and M.E. degrees from Xi'an Jiaotong University, Xi'an, China, in 1997, 2000, and Ph.D. degree from Waseda University, Tokyo, Japan, in 2003, respectively. He was an exchange student between Waseda and Xi'an Jiaotong University from 1999 to 2000. From 2001 he had been a research associate at Waseda University and he is currently an assistant professor at the same university. His research interests include multi media communication, web performance and network traffic. He received the SIEMENS Prize in 1998, and ROCKWELL Automation Master of Science Award in 1999. He is a member of IEEE and IEE.

Jiro Katto received B.S., M.E. and Ph.D. degrees in electrical engineering from University of Tokyo in 1987, 1989 and 1992, respectively. He worked for NEC Corporation from 1992 to 1999. He was also a visiting scholar at Princeton University, NJ, USA, from 1996 to 1997. He then joined Waseda University in 1999. Since 2004, he has been a professor of the Department of Computer Science, Science and Engineering, Waseda University. He has also been a director of the Electronic and Information Technology Development Department, New Energy and Industrial Technology Development Organization from 2004 to 2008. His research interest is in the field of multimedia communication systems and multimedia signal processing. He received the Best Student Paper Award at SPIE Conference of Visual Communication and Image Processing in 1991, and received the Young Investigator Award of IEICE in 1995. He is a member of IEEE, ACM, IPSJ, IEICE, and ITE.