Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/070641

Performance Bounds for P2P Streaming System with Transcoding

Shuai Zeng[1,*], *Lemin Li*[1] *and Dan Liao*[2]

[1] School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, China
[2] Institute of Electronic and Information Engineering in Dongguan, UESTC, Chengdu, China

Received: 6 Apr. 2013, Revised: 10 Aug. 2013, Accepted: 11 Aug. 2013
Published online: 1 Nov. 2013

Abstract: In this paper, we develop a fluid model that seeks to expose the fundamental characteristics and mathematical theory of peer-to-peer streaming system with transcoding. We find out and prove that, to provide peers receiving data above some flow rate, there is a lower bound of server upload bandwidth in this kind of system. We give a flow rate allocation algorithm to achieve the minimal server upload bandwidth in the proof. We compare this lower bound with the minimal demand of server upload bandwidth of no transcoding system. And, we prove that, the demand of server upload bandwidth in transcoding peer-to-peer streaming system using the proposed algorithm is the necessary (not sufficient) condition for the no transcoding one. At last, we give the simulations experiment to show the difference of server load in transcoding and no transcoding systems.

Keywords: Peer-to-peer Network, Network Optimization, Flow Rate Allocation, Streaming System.

1 Introduction

With the popular of peer-to-peer (P2P) streaming, peers become more and more complex. These clients, such as PC, TV, tablet, PDA, cellphone, and so on, have various screen sizes, color depth and multimedia qualities. Specially, they may have different multimedia coding algorithm with heterogeneous hardware and software, and they may have different bandwidth with heterogeneous network [1,2]. In traditional P2P multimedia streaming system, one single overlay network or server cannot support all of these clients. However more networks and servers may need more resources.

In recent years, there are some literature which studies on transcoding (or named peer-assisted transcoding [4]) technique utilized in P2P streaming system. [3] proposes a multimedia streaming architecture in which transcoding services coordinate to transform the streaming data into different formats in P2P system. [4] proposes a system named PAT (Peer-Assisted Transcoding) to enable effective online transcoding and seek to reduce the bandwidth consumption and computing overhead in P2P network. In [5], the transcoding technique is used in some total new network environments. The paper discusses issues that are relevant to enabling P2P streaming in networked consumer electronics, NAT/firewall traversal, and codec inflexibility. [6,9] also discusses the video transcoding in P2P network of IPTV system. [7] proposes a P2P transcoding method for heterogeneity mobile streaming. The paper seeks to increase the flexibility of coding data, which bases on diverse display size, computing power, memory, and media capabilities in devices. [8] designs a transcoding system for P2P streaming based on farming computing architecture. [2] summarizes and categorizes the P2P streaming system with transcoding by different network environments and transcoding service places. [10] presents a P2P streaming system named CloudStream, which is a cloud-based video proxy that can deliver streaming videos by transcoding the original video in real time to a scalable codec. And [11] propose a collaborative strategy that leverages the peering architecture of P2P networks and makes the computational resources of peers sharable and collaborative.

These researches announces that, compared with traditional designs, P2P streaming system with transcoding have better performances in some situations, such as heterogeneous drivers, various multimedia coding, peer churn, and so on. Nevertheless, existent studies just focus on network protocols design and

* Corresponding author e-mail: maillinger@gmail.com

multimedia coding algorithm, which lacks of mathematically investigate and deeply understand their systems in network fluid theory. Even more importantly, there exists no relative research that focuses on what conditions the transcoding suits and how much benefit exactly in quantity the new technique taken to the system.

In this paper, we are interested in the basic fluid theory for P2P streaming system with transcoding. We develop a fluid model that seeks to expose some fundamental characteristics and limitations of P2P streaming system with transcoding. There are some literature that discusses and analyzes the issues of P2P streaming system upload bandwidth by mathematic fluid model in various situations. [12] develops a basic stochastic model and fluid theory for the typical P2P streaming system. [13] derives and proves the performance bounds for minimum server load and maximum streaming rate in P2P streaming system. [14, 15, 16] discusses the issue of achievable streaming rate in P2P streaming system with multiple multimedia channels. [17] provides a taxonomy of P2P streaming system, depending on whether the given topology is a full mesh graph or an arbitrary graph, whether the number of peers a node can have is bounded or not, and so on. In these different situations, the paper discusses the maximum rate achievable by all receivers respectively. And in [18], the authors develop a fluid model for P2P streaming system with network coding and mathematically analyze the performance of this kind of system. In this paper, our analysis and results are based on both previous research and the features of P2P streaming system with transcoding. Furthermore, we compare P2P streaming system with transcoding with no transcoding system both in mathematical analysis and simulation experiment.

The remainder of the paper is organized as follows. Section 2 describes and insights the basic models of P2P streaming system with transcoding. In Section 3, we compute and prove the minimal bandwidth demand of server for given flow rate of all peers, and we give an algorithm to achieve this minimal load in the proof. We also compare this minimal value with the one in no transcoding situations in this section. And we give our simulation result in Section 4. Finally, we conclude this paper and discuss future work in Section 5.

2 Model and Algoritm

P2P streaming system with transcoding can support multiple multimedia coding rates in one overlay network. (For brevity, use TRANSCODING to mean P2P streaming system with transcoding in the remainder of this paper.) As shown in Figure 1, in a TRANSCODING, system is divided into several regions. In the same region, peers all receive the same multimedia coding data. This means every region has a certain coding rate. We call a TRANSCODING is an m-region system when there are m regions living in the system. By the order of

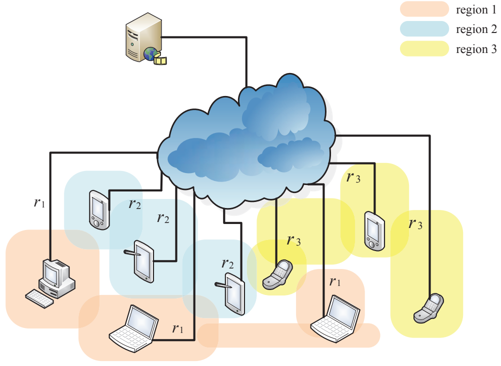

Fig. 1: Example of P2P streaming system with transcoding.

multimedia quality, we denote by r_i $(i = 1, \ldots, m)$ for the coding rate of the regions, and we call a region region i , when their coding rate is r_i. A peer in region can play multimedia program smoothly when it receives fresh data bits from system at least at rate r_i.

Definition 1. In an m-region TRANSCODING, all participating peers receive the multimedia data at least at rate r_i $(i = 1, \ldots, m)$ of their own region, we say that the system provides general universal streaming (GUS) or the system runs on GUS.

As we know, the essential purpose of P2P streaming system is to ensure QoE, which is all peers receiving multimedia data from the system above multimedia coding rate, and, at the same time, minimize the server bandwidth consumption. Denote by for the minimal demand of server upload bandwidth guaranteeing the system providing GUS. Notice the definition of universal streaming (US) in [12]. That an m-region TRANSCODING can provide GUS means that every subsystem of region r_i $(i = 1, 2, \ldots, m)$ can provide US. Compared with the traditional P2P streaming system, which is committed to minimize the demand of server for providing US, this paper seeks to find $u_{s\min}$.

Before giving $u_{s\min}$ in next subsection, we continue to introduce some relative notations and expressions of the system model. In our system, there is one server and total n peers in m regions. Let $\mathbf{P}$ be the set of all peers. Let n_i be the number of peers in region i and $\mathbf{P}_i$ be the set of peers in region i . Denote by s for the server and p_{ij} for the jth peer in region i . We have

$$\mathbf{P} = \{p_{ij}\}, \quad for\ i = 1, \ldots, m;\ j = 1, \ldots, n_i$$

$$\mathbf{P}_i = \{p_{ij}\}, \quad for\ j = 1, \ldots, n_i$$

$$n = |\mathbf{P}| = \sum_{i=1}^{m} n_i, \quad n_i = |\mathbf{P}_i|$$

Denote by $\mathbf{R}$ for the set of coding rate in the system, which is $\mathbf{R} = \{r_1, \ldots, r_m\}$. As the assumptions mentioned in previous section, we have that the r_i $(i = 1, \ldots, m-1)$

coding data can be transcoded to $\forall r_k \in \mathbf{R} : k > i$ coding data by the peers in region i. Denote by u_s for the upload bandwidth of s and u_{ij} for the upload bandwidth of p_{ij} . Let $\mathrm{u}(\cdot)$ be the function of upload capacity summation. For example, $\mathrm{u}(\mathbf{P}) = \sum_{\mathbf{P}} u_{ij} = \sum_{i=1}^{m} \sum_{j=1}^{n_i} u_{ij}$. Let $\overline{u}$ be the average upload bandwidth of all peers and $\overline{u_i}$ be the average upload bandwidth of peers in region i. We have

$$\overline{u} = \frac{\mathrm{u}(\mathbf{P})}{|\mathbf{P}|} = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n_i} u_{ij}}{n}$$

$$\overline{u_i} = \frac{\mathrm{u}(\mathbf{P}_i)}{|\mathbf{P}_i|} = \frac{\sum_{j=1}^{n_i} u_{ij}}{n_i}$$

Other notations introduced in the system model are summarized in Table 1. We give the proof of $u_{s\min}$ in this subsection. We divide the proof into two parts. We give a bound for $u_{s\min}$ in part one and prove GUS can be supported with this bound in part two. Notice that part two is also a flow rate allocation algorithm to achieve the minimal server load (the minimal demand of server upload bandwidth) of TRANSCODING.

Theorem 1. Let $u_{s\min}$ denote minimal bandwidth demand of server for a m-regions TRANSCODING providing GUS, then

$$u_{s\min} = \max_i \left(r_i + \sum_{k=0}^{i-1} n_k (r_k - \overline{u_k}) \right), \tag{1}$$
$$n_0 = r_0 = \overline{u_0} = r_{m+1} = 0, \quad i = 1, \ldots, m+1$$

Proof:

Part one:

Notice that for the whole system $\mathbf{P} = \bigcup_{i=1}^{m} \mathbf{P}_i$, it is obviously $u_{s\min} \geq r_1$. And for the subsystem $\bigcup_{i=2}^{m} \mathbf{P}_i$, $u_{s\min} + n_1 (\overline{u_1} - r_1) \geq r_2$, where region 1 at least costs $n_1 (r_1 - \overline{u_1})$ bandwidth. By analogy, for the subsystem $\bigcup_{k=i}^{m} \mathbf{P}_k$ $(i = 2, \ldots, m)$, $u_{s\min} + \sum_{k=1}^{i-1} n_k (\overline{u_k} - r_k) \geq r_i$. . And for the whole system, we have $u_{s\min} + \sum_{i=1}^{m} n_i (\overline{u_i} - r_i) \geq 0$.

Therefore

$$\begin{cases} u_{s\min} \geq r_1 \\ u_{s\min} \geq r_i + \sum_{k=1}^{i-1} n_k (r_k - \overline{u_k}), \quad for\ i = 2, \ldots, m \\ u_{s\min} \geq \sum_{i=1}^{m} n_i (r_i - \overline{u_i}) \end{cases} \tag{2}$$

For convenience, let $n_0 = r_0 = \overline{u_0} = r_{m+1} = 0$. Then, combining these three inequalities gives

$$u_{s\min} \geq \max_i \left(r_i + \sum_{k=0}^{i-1} n_k (r_k - \overline{u_k}) \right), \tag{3}$$
$$n_0 = r_0 = \overline{u_0} = r_{m+1} = 0, \quad i = 1, \ldots, m+1$$

It remains to show that if

$$u_s = \max_i \left(r_i + \sum_{k=0}^{i-1} n_k (r_k - \overline{u_k}) \right), \tag{4}$$
$$n_0 = r_0 = \overline{u_0} = r_{m+1} = 0, \quad i = 1, \ldots, m+1$$

then GUS can be supported.

Part two:

Let $\mathbf{P}_0 = \{s\}$. Consider the subsystem $\mathbf{P}_0 \bigcup \mathbf{P}_1$ firstly.

When $r_1 \geq n_1 (r_1 - \overline{u_1})$, i.e. $r_1 \leq \frac{n_1 \overline{u_1}}{n_1 - 1}$.

Consider a multimedia stream of rate r_1. Divide this multimedia stream into n_1 substreams, with the jth substream having rate

$$s_{1j}^{o} = \frac{u_{1j} r_1}{\mathrm{u}(\mathbf{P}_1)} = \frac{u_{1j} r_1}{n_1 \overline{u_1}}, \quad for\ j = 1, \ldots, n_1$$

Notice that $\sum_{j=1}^{n_1} s_{1j}^{o} = r_1 \leq u_s$. So the server can copy the jth substream to the p_{1j} respectively. Furthermore, because

$$(n_1 - 1) s_{1j}^{o} = \frac{(n_1 - 1) u_{1j}}{n_1 \overline{u_1}} \cdot r_1 \leq \frac{(n_1 - 1) u_{1j}}{n_1 \overline{u_1}} \cdot \frac{n_1 \overline{u_1}}{(n_1 - 1)} = u_{1j}$$

, p_{1j} can copy its stream to each of the other $n_1 - 1$ peers in region 1. Thus each peer in region 1 receives a substream from the server and also receives $n_1 - 1$ additional substreams from the other $n_1 - 1$ peers in the same region. The total rate at which p_{1j} receives is

$$tr_{1j} = s_{1j}^{o} + \sum_{k:k \neq j} s_{1k}^{o} = \sum_{j=1}^{n_1} s_{1j}^{o} = r_1$$

Hence the rate r_1 can be supported in region 1, which means, in this situation, the system can provide US for region 1.

When $r_1 < n_1 (r_1 - \overline{u_1})$.

In this situation, divide the multimedia stream into $n_1 + 1$ substreams, with th substream having rate

$$\begin{cases} s_{1j}^{o} = \frac{u_{1j}}{n_1 - 1}, \quad for\ j = 1, \ldots, n_1 \\ s_{1 n_1 + 1}^{o} = r_1 - \frac{\mathrm{u}(\mathbf{P}_1)}{n_1 - 1} = r_1 - \frac{n_1 \overline{u_1}}{n_1 - 1} \end{cases}$$

And the server copy two substreams to each peer in region 1: the jth substream s_{1j}^{o} and the substream $s_{1 n_1 + 1}^{o}$. The server can do this because

$$\sum_{j=1}^{n_1} s_{1j}^{o} + n_1 s_{1 n_1 + 1}^{o} = \frac{n_1 \overline{u_1}}{n_1 - 1} + n_1 \left(r_1 - \frac{n_1 \overline{u_1}}{n_1 - 1} \right)$$
$$= n_1 (r_1 - \overline{u_1}) \leq u_s$$

Furthermore, because

$$(n_1 - 1) s_{1j}^{o} = (n_1 - 1) \cdot \frac{u_{1j}}{n_1 - 1} = u_{1j}, \quad for\ j = 1, \ldots, n_1$$

p_{1j} can copy its stream s_{1j}^{o} to each of the other $n_1 - 1$ peers in region 1. Thus each peer in region 1 receives two substreams from the server and also receives n_1-1 additional substreams from the other n_1-1 peers in the same region. The total rate at which p_{1j} receives is

$$tr_{1j}=s^o_{1n_1+1}+s^o_{1j}+\sum_{k:k\neq j}s^o_{1k}=s^o_{1n_1+1}+\sum_{j=1}^{n_1}s^o_{1j}=r_1$$

Hence the rate can be supported in region 1, which means, when $r_1<n_1(r_1-\overline{u_1})$, the system can provide US for region 1 also.

Then, whether $r_1\geq n_1(r_1-\overline{u_1})$ or $r_1<n_1(r_1-\overline{u_1})$, the 1-region subsystem $\mathbf{P}_0\bigcup\mathbf{P}_1$ can run on GUS, and the total rest of available upload bandwidth is

$$u'_{\text{all}}=u'_s+\sum_{h=1}^{n_1}u'_{1h}=u_s+n_1(\overline{u_1}-r_1)\geq r_2$$

Next, we consider the subsystems $\bigcup_{k=0}^{i}\mathbf{P}_k$, $for\ i=2,\ldots,m$. Notice that, the total rest of available upload bandwidth of $\bigcup_{k=0}^{i-1}\mathbf{P}_k$ is

$$u^{(i-1)}_{\text{all}}=u^{(i-1)}_s+\sum_{g=1}^{i-1}\sum_{h=1}^{n_g}u^{(i-1)}_{gh}=u_s+\sum_{k=1}^{i-1}n_k(\overline{u_k}-r_k)$$

And as (4), we have $u^{(i-1)}_{\text{all}}\geq r_i$.

When $r_i\geq n_i(r_i-\overline{u_i})$, i.e. $r_i\leq\frac{n_i\overline{u_i}}{n_i-1}$.

Consider a multimedia stream of rate r_i, which can be transcoded from the stream of rate r_k, $for\ k=1,\ldots,i-1$. Divide this multimedia stream into n_i substreams, with the jth substream having rate

$$s^o_{ij}=\frac{u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot\frac{u^{(i-1)}_s}{u^{(i-1)}_{\text{all}}}\cdot r_i=\frac{u_{ij}}{n_i\overline{u_i}}\cdot\frac{u^{(i-1)}_s}{u_s+\sum_{k=1}^{i-1}n_k(\overline{u_k}-r_k)}\cdot r_i,$$

$$(for\ j=1,\ldots,n_i\ )$$

And divide this multimedia stream into n_i substreams, with the jth substream having rate

$$s^{gh}_{ij}=\frac{u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot\frac{u^{(i-1)}_{gh}}{u^{(i-1)}_{\text{all}}}\cdot r_i=\frac{u_{ij}}{n_i\overline{u_i}}\cdot\frac{u^{(i-1)}_{gh}}{u_s+\sum_{k=1}^{i-1}n_k(\overline{u_k}-r_k)}\cdot r_i,$$

$$(for\ j=1,\ldots,n_i)$$

Notice that

$$\sum_{j=1}^{n_i}s^o_{ij}=\frac{\sum_{j=1}^{n_i}u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot\frac{u^{(i-1)}_s}{u^{(i-1)}_{\text{all}}}\cdot r_i=u^{(i-1)}_s\cdot\frac{r_i}{u^{(i-1)}_{\text{all}}}$$

. As $u^{(i-1)}_{\text{all}}\geq r_i$, we have $\sum_{j=1}^{n_i}s^o_{ij}\leq u^{(i-1)}_s$. So the server can copy s^o_{ij} to the $p_{ij}(j=1,\ldots,n_i)$ respectively. And, as

$$\sum_{j=1}^{n_i}s^{gh}_{ij}=\frac{\sum_{j=1}^{n_i}u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot\frac{u^{(i-1)}_{gh}}{u^{(i-1)}_{\text{all}}}\cdot r_i=u^{(i-1)}_{gh}\cdot\frac{r_i}{u^{(i-1)}_{\text{all}}}\leq u^{(i-1)}_{gh}$$

$p_{gh}(g=1,\ldots,i-1;h=1,\ldots,n_g)$ can copy s^{gh}_{ij} to $p_{ij}(j=1,\ldots,n_i)$ respectively. So $p_{ij}(j=1,\ldots,n_i)$ gets

$$s^o_{ij}+\sum_{g=1}^{i-1}\sum_{h=1}^{n_g}s^{gh}_{ij}=\frac{u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot\frac{u^{(i-1)}_s}{u^{(i-1)}_{\text{all}}}\cdot r_i+\frac{u_{ij}r_i}{\mathrm{u}(\mathbf{P}_i)}\cdot\frac{\sum_{g=1}^{i-1}\sum_{h=1}^{n_g}u^{(i-1)}_{gh}}{u^{(i-1)}_{\text{all}}}$$

$$=\frac{u_{ij}r_i}{\mathrm{u}(\mathbf{P}_i)}$$

Furthermore, because

$$(n_i-1)\left(s^o_{ij}+\sum_{g=1}^{i-1}\sum_{h=1}^{n_g}s^{gh}_{ij}\right)=\frac{(n_i-1)u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot r_i=\frac{(n_i-1)u_{ij}}{n_i\overline{u_i}}\cdot r_i$$

$$\leq\frac{(n_i-1)u_{ij}}{n_i\overline{u_i}}\cdot\frac{n_i\overline{u_i}}{n_i-1}=u_{ij}$$

$p_{ij}(j=1,\ldots,n_i)$ can copy $s^o_{ij}+\sum_{g=1}^{i-1}\sum_{h=1}^{n_g}s^{gh}_{ij}$ to each of the other n_i-1 peers in region i. Thus each peer in region i receives a substream from each peer or the server in the set $\bigcup_{k=0}^{i-1}\mathbf{P}_k$ and also receives n_i-1 additional substreams from the other n_i-1 peers in the same region. The total rate at which p_{ij} receives is

$$tr_{ij}=\sum_{j=1}^{n_i}\left(s^o_{ij}+\sum_{g=1}^{i-1}\sum_{h=1}^{n_g}s^{gh}_{ij}\right)=\frac{\sum_{j=1}^{n_i}u_{ij}}{\mathrm{u}(\mathbf{P}_i)}\cdot r_i=r_i$$

Hence the rater_i an be supported in the region i, which means, in this situation, the system can provide US for the region i.

When $r_i<n_ir_i-n_i\overline{u_i}$.

Consider a multimedia stream of rate r_i which can be transcoded from the stream of rate r_k, $for\ k=1,\ldots,i-1$. Divide this multimedia stream into n_i+1 substreams, with the jth substream having rate

$$\begin{cases}s^o_{ij}=\frac{u_{ij}}{n_i-1}\cdot\frac{u^{(i-1)}_s}{u_s+\sum_{k=1}^{i-1}n_k(\overline{u_k}-r_k)}, & for\ j=1,\ldots,n_i\\ s^o_{in_i+1}=\left(r_i-\frac{n_i\overline{u_i}}{n_i-1}\right)\left(\frac{u^{(i-1)}_s}{u_s+\sum_{k=1}^{i-1}n_k(\overline{u_k}-r_k)}\right)\end{cases}$$

And divide this multimedia stream into n_i+1 substreams, with the jth substream having rate

$$\begin{cases} s_{ij}^{gh} = \frac{u_{ij}}{n_i-1} \cdot \frac{u_s^{(i-1)}}{u_s + \sum_{k=1}^{i-1} n_k(\overline{u_k}-r_k)}, & for\ j = 1,\ldots,n_i \\ s_{in_i+1}^{gh} = \left(r_i - \frac{n_i\overline{u_i}}{n_i-1}\right)\left(\frac{u_{gh}^{(i-1)}}{u_s + \sum_{k=1}^{i-1} n_k(\overline{u_k}-r_k)}\right) \end{cases}$$

Notice that

$$\sum_{j=1}^{n_i} s_{ij}^o + n_i s_{in_i+1}^o = \frac{n_i(r_i - \overline{u_i})\, u_s^{(i-1)}}{u_s + \sum_{k=1}^{i-1} n_k(\overline{u_k} - r_k)}$$

As (4), we have $u_s \geq \sum_{k=1}^{i} n_k(r_k - \overline{u_k}) i = 1,\ldots,m$, i.e.

$$u_s + \sum_{k=1}^{i-1} n_k(\overline{u_k} - r_k) \geq n_i(r_i - \overline{u_i})$$

So $\sum_{j=1}^{n_i} s_{ij}^o + n_i s_{in_i+1}^o \leq u_s^{(i-1)}$, which means the server can copy $s_{ij}^o + s_{in_i+1}^o$ to the $p_{ij}(j = 1,\ldots,n_i)$ respectively. And, as

$$\sum_{j=1}^{n_i} s_{ij}^{gh} + n_i s_{in_i+1}^{gh} = \frac{n_i(r_i - \overline{u_i})}{u_s + \sum_{k=1}^{i-1} n_k(\overline{u_k} - r_k)} \cdot u_{gh}^{(i-1)} \leq u_{gh}^{(i-1)}$$

$p_{gh}(g = 1,\ldots,i-1; h = 1,\ldots,n_g)$ can copy $s_{ij}^{gh} + s_{in_i+1}^{gh}$ to $p_{ij}(j = 1,\ldots,n_i)$ respectively. So $p_{ij}(j = 1,\ldots,n_i)$ gets flow rate

$$\left(s_{ij}^o + \sum_{g=1}^{i-1}\sum_{h=1}^{n_g} s_{ij}^{gh}\right) + \left(s_{in_i+1}^o + \sum_{g=1}^{i-1}\sum_{h=1}^{n_g} s_{in_i+1}^{gh}\right)$$

Furthermore, because

$$(n_i - 1)\left(s_{ij}^o + \sum_{g=1}^{i-1}\sum_{h=1}^{n_g} s_{ij}^{gh}\right) = u_{ij}$$

$p_{ij}(j = 1,\ldots,n_i)$ can copy $s_{ij}^o + \sum_{g=1}^{i-1}\sum_{h=1}^{n_g} s_{ij}^{gh}$ to each of the other $n_i - 1$ peers in region i. Thus each peer in region i receives two substreams from each peer or the server in the set $\bigcup_{k=0}^{i-1} \mathbf{P}_k$ and also receives $n_i - 1$ additional substreams from the other $n_i - 1$ peers in the same region. The total rate at which p_{ij} receives is

$$tr_{ij} = \sum_{j=1}^{n_i}\left(s_{ij}^o + \sum_{g=1}^{i-1}\sum_{h=1}^{n_g} s_{ij}^{gh}\right) + s_{in_i+1}^o + \sum_{g=1}^{i-1}\sum_{h=1}^{n_g} s_{in_i+1}^{gh} = r_i$$

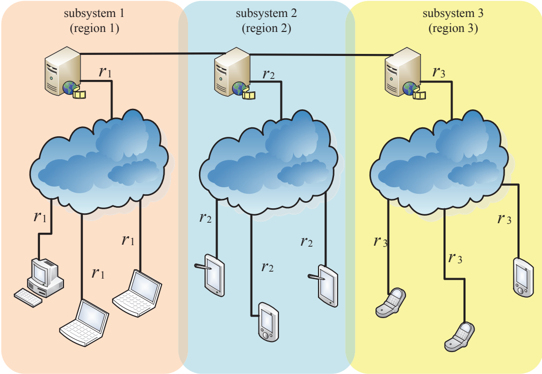

Fig. 2: Example of no transcoding P2P streaming system.

. Hence the rate r_i can be supported in region i, which means, in this situation, the system can provide US for region i also.

To sum up, when u_s satisfys (4), based on the flow rate allocation algorithm in 'part two', the TRANSCODING can provide US for all of the regions, which means the whole system can run on GUS. Considering (3), we have

$$u_{s\min} = \max_i \left(r_i + \sum_{k=0}^{i-1} n_k(r_k - \overline{u_k})\right),$$

$$n_0 = r_0 = \overline{u_0} = r_{m+1} = 0, \quad i = 1,\ldots,m+1.$$

3 Comparison

No transcoding P2P streaming system, as shown in Fig. 2, is composed of multiple independent subsystems. The overlay networks of these subsystems have their own multimedia coding rate r_i $(i = 1,\ldots,m)$ respectively, and no communication among them. For brevity, we also call the set of peers, whose coding rate is r_i , region i in no transcoding design. And other notations mean the same as TRANSCODING. Let $u_{s\min i}^{NT}$ $(i = 1,\ldots,m)$ be the minimal bandwidth demand of server for subsystem $s \cup \mathbf{P}_i$ $(i = 1,\ldots,m)$ running on US in no transcoding design. And let $u_{s\min}^{NT}$ be the minimal bandwidth demand of server for every subsystem running on US, which is equivalent to GUS in TRASCODING.

Base on [12,13], for the subsystem $s \cup \mathbf{P}_1$,

$$u_{s\min 1}^{NT} = \max(r_1,\ n_1(r_1 - \overline{u_1}))$$

for the subsystem $s \cup \mathbf{P}_2$

$$u_{s\min 2}^{NT} = \max(r_2,\ n_2(r_2 - \overline{u_2}))$$

and for the subsystem $s \cup \mathbf{P}_m$

$$u_{s\min m}^{NT} = \max(r_m,\ n_m(r_m - \overline{u_m}))$$

As the feature of no transcoding P2P streaming system, we have

$$u^{NT}_{s\min} = \sum_{i=1}^{m} u^{NT}_{s\min i} = \sum_{r_i \geq n_i(r_i - \overline{u_i})} r_i + \sum_{r_i < n_i(r_i - \overline{u_i})} n_i (r_i - \overline{u_i}) \quad (5)$$

(5) is a useful equation to figure $u^{NT}_{s\min}$ but not a clear way expressing the essence between no transcoding design and TRASCODING using the proposed algorithm. Let us turn back to the subsystem $s \cup \mathbf{P}_i$. Consider the situation of subsystem $s \cup \mathbf{P}_i$ running on US. First, $u^{NT}_{s\min i}$ need to be bigger than multimedia coding rate in region i. Second, the total upload bandwidth of subsystem $s \cup \mathbf{P}_i$ should be able to satisfy the total download consumption of $\mathbf{P}_i$. So we have the necessary conditions for subsystem $s \cup \mathbf{P}_i$ running on US, which is

$$\begin{cases} u^{NT}_{s\min i} \geq r_i \\ u^{NT}_{s\min i} + n_i \overline{u_i} \geq n_i r_i \end{cases} \quad (6)$$

Furthermore, if we need all subsystem in no transcoding design running on US, which is equivalent to GUS in TRANSCODING, the conditions (6) must be satisfied for all $i = 1, \ldots, m$ at the same time, which is

$$\begin{cases} u^{NT}_{s\min i} \geq r_i \\ u^{NT}_{s\min i} + n_i \overline{u_i} \geq n_i r_i \end{cases}, \quad for\ i = 1, \ldots, m.$$

Notice that $\sum_{i=1}^{m} u^{NT}_{s\min i} = u^{NT}_{s\min}$ and $\sum_{i=1}^{m} r_i \geq r_1$, as $u^{NT}_{s\min i} \geq r_i$, we have

$$u^{NT}_{s\min} \geq r_1 \quad (7)$$

And as $u^{NT}_{s\min i} + n_i \overline{u_i} \geq n_i r_i, \quad for\ i = 1, \ldots, m$, we have

$$\sum_{i=1}^{k} u^{NT}_{s\min i} + \sum_{i=1}^{k} n_i \overline{u_i} \geq \sum_{i=1}^{k} n_i r_i, \quad for\ k = 1, \ldots, m \quad (8)$$

In (8), when $k = m$, we have

$$u^{NT}_{s\min} \geq \sum_{i=1}^{m} n_i (r_i - \overline{u_i}) \quad (9)$$

Noctice $u^{NT}_{s\min k+1} \geq r_{k+1}, \quad for\ k = 1, \ldots, m-1$. So adding $u^{NT}_{s\min k+1} \geq r_{k+1}$ to (8), we have

$$\sum_{i=1}^{k+1} u^{NT}_{s\min i} + \sum_{i=1}^{k} n_i \overline{u_i} \geq \sum_{i=1}^{k} n_i r_i + r_{k+1}, \quad for\ k = 1, \ldots, m-1$$

i.e.

$$\sum_{i=1}^{k+1} u^{NT}_{s\min i} \geq r_{k+1} + \sum_{i=1}^{k} n_i (r_i - \overline{u_i}), \quad for\ k = 1, \ldots, m-1$$

As $u^{NT}_{s\min} = \sum_{i=1}^{m} u^{NT}_{s\min i} \geq \sum_{i=1}^{k+1} u^{NT}_{s\min i}, \quad for\ k = 1, \ldots, m-1$, we have

$$u^{NT}_{s\min} \geq r_{k+1} + \sum_{i=1}^{k} n_i (r_i - \overline{u_i}), \quad for\ k = 1, \ldots, m-1 \quad (10)$$

We rewrite (10) as

$$u^{NT}_{s\min} \geq r_i + \sum_{k=1}^{i-1} n_k (r_k - \overline{u_k}), \quad for\ i = 2, \ldots, m \quad (11)$$

Notice that (7), (9) and (11) all come from necessary conditions for whole no transcoding design running on US. Compared with (1) and (2) of TRNSCODING using the proposed algorithm, we conclude that, in the same situations, the conditions that provide all regions in no transcoding design running on US are the sufficient conditions for TRNSCODING using the proposed algorithm running on GUS, which means the bandwidth demand of server for whole no transcoding design running on US is always able to satisfy TRNSCODING using the proposed algorithm running on GUS. So we have our second theorem.

Theorem 2. Let $u^{NT}_{s\min}$ be the minimal demand of s upload bandwidth for all peers in no transcoding P2P streaming system receiving data above their multimedia coding rate. Let $u^{T}_{s\min}$ be the minimal demand of s upload bandwidth for all peers in TRASCODING using the proposed algorithm receiving data above their multimedia coding rate. In the same situations, we always have

$$u^{NT}_{s\min} \geq u^{T}_{s\min}$$

where

$$\begin{cases} u^{NT}_{s\min} = \sum_{i:r_i \geq n_i(r_i - \overline{u_i})} r_i + \sum_{i:r_i < n_i(r_i - \overline{u_i})} n_i (r_i - \overline{u_i}) \\ u^{T}_{s\min} = \max_{i} \left(r_i + \sum_{k=0}^{i-1} n_k (r_k - \overline{u_k}) \right) \end{cases}$$

4 Simulation

In our simulation experiments, we implement the algorithms of P2P streaming system with transcoding and no transcoding described in the Section 2 and 3. Our test networks are generated by the Georgia Tech's topology generator [19].

To conduct rigorous quantitative analysis of the systems under wide range of working conditions, we implement our experiments to emulate the characteristics of realistic systems with different parameters and a large number of test times. The parameters are chosen randomly. We set m, n_i, r_i and $\overline{u_i}$ all randomly, which chooses m from 2 to 10, n_i from 5 to 100, r_i from 100kbps to 1500kbps, and $\overline{u_i}$ from 20kbps to 1000kbps. We test the server load of transcoding and no transcoding designs 500 times respectively in the same conditions, and survey the difference $D = u^{NT}_{s\min} - u^{T}_{s\min}$. The result is shown in Fig. 3. We can see that the D value ranges from about 100kbps to about 10Mbps, that is to say, the server load in transcoding system using the proposed algorithm is less than no transcoding design, which meets the mathematical analysis.

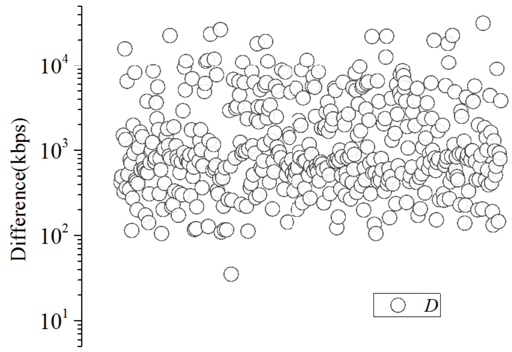

Fig. 3: Difference of server load in transcoding and no transcoding design.

Table 1: Notations used in this paper.

Symbol	Illustration
n	Total number of peers in the system.
n_i	Total number of peers in region i.
m	Total number of regions in the system.
s	Server.
p_{ij}	The jth peer in the region i.
$\mathbf{P}$	Set of all peers.
$\mathbf{P}_i$	Set of peers in region i.
r_i	Multimedia coding rate in region i.
$\mathbf{R}$	Set of coding rate in the system.
u_s	Upload bandwidth of server.
u_{ij}	Upload bandwidth of p_{ij}.
$\mathrm{u}(\cdot)$	Function of upload bandwidth Summation.
$\bar{u}$	Average upload bandwidth of all peers.
$\bar{u_i}$	Average upload bandwidth of peers in region i.
$u_{s\min}$	Minimal server load to provide GUS.
u_s^i	Rest of s upload bandwidth below the region i.
$u_{gh}^{(i)}$	Rest of p_{gh} upload bandwidth below region i.
$u_{\mathrm{all}}^{(i)}$	Total rest of upload bandwidth below region i.
s_{ij}^o	The rate of multimedia substream from s to p_{ij}.
s_{ij}^{gh}	The rate of multimedia substream from p_{gh} to p_{ij}.
tr_{ij}	The total rate of p_{ij} received from the system.

5 Conclusion

In this paper, we have mathematically studied the performance of P2P streaming system with transcoding. We have derived the performance bounds of server load for the existence of a fluid distribution scheme that achieves *GUS* in P2P streaming system with transcoding, and given an algorithm to achieve the minimal load in the proof. We have compared the minimal server load with typical traditional P2P streaming system with no transcoding. The results have shown that, in the same situations, the minimal server load of transcoding P2P streaming system using the proposed algorithm is less than the no transcoding design. And the result of simulation experiment also shows the same conclusion.

Acknowledgement

This work is partially supported by The National Natural Science Foundation of China (NSFC) (NO. 60972030 and 61001084) and Fundamental Research Funds for the Central Universities (ZYGX2010X001). The author is grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] M.-T. Lu, J.-C. Wu, K.-J. Peng, P. Huang, J. J. Yao, and H. H. Chen, Design and evaluation of a P2P IPTV system for heterogeneous networks, IEEE Trans. Multimedia, **9**, 1568-1579 (2007).
[2] T. Wolf, In-network service for customization in next generation networks, IEEE Network, **24**, 6-12 (2010).
[3] F. Chen, T. Repantis, and V. Kalogeraki, Coordinated media streaming and transcoding in peer-to-peer systems, in Proc. IEEE Int. Parallel Distributed Process. Symp., 56b (2005).
[4] D. Liu, E. Setton, B. Shen, and S. Chen, PAT: Peer-assisted transcoding for overlay streaming to heterogeneous devices, in Proc. Int. Workshop Netw. Oper. Syst. Support Dig. Audio Video, (2007).
[5] S. R. Narayanan, D. Braun, J. Buford, R. S. Fish, A. D. Gelman, A. Kaplan, R. Khandelwal, E. Shim, and H. Yu, Peer-to-peer streaming for networked consumer electronics, IEEE Commun. Mag., **45**, 124-131 (2007).
[6] Guntur Ravindra , Sumit Kumar , Suresh Chintada, Distributed media transcoding using a P2P network of set top boxes, Proceedings of the 6th IEEE Conference on Consumer Communications and Networking Conference,Las Vegas, NV, USA, 1014-1015 (2009).
[7] J. Noh, M. Makar, and B. Girod, Streaming to mobile users in a peer-to-peer network, In International Mobile Multimedia Communications Conference, London, England, (2009).
[8] Can Yang, Yuanqiong Chen, Yilong Shen, The Research on a P2P Transcoding System Based on Distributed Farming Computing Architecture, in Proc. of KESE '09, Pacific-Asia Conference on Knowledge Engineering and Software Engineering, 55-58 (2009).
[9] Atenas M, Canovas A, Garcia M, Lloret J, IPTV transcoding to avoid network congestion, The Sixth International Conference on Networking and Services, ICNS , Cancun, Mxico, **2010**, (2010).
[10] Zixia Huang, Chao Mei, Li Erran Li and Thomas Woo, CloudStream: delivering high-quality streaming video through a cloud-based H.264/SVC proxy, In Proceedings of IEEE International Conference on Computer Communications (INFOCOM) miniconference, (2011).
[11] J.-C. Wu, P. Huang, J. J. Yao, and H. H. Chen, A collaborative transcoding strategy for live broadcasting over peer-to-peer IPTV networks, IEEE Trans. Circuits Syst. Video Technol., **21**, 220-224 (2011).

[12] R. Kumar, Y. Liu, and K. W. Ross, Stochastic Fluid Theory for P2P Streaming Systems, in Proc. of IEEE INFOCOM, (2007).
[13] S. Liu, R. Z. Shen, W. Jiang, J. Rexford, and M. Chiang, Performance Bounds for Peer-Assisted Live Streaming, in Proc. of ACM SIGMETRICS, (2008).
[14] D. Wu, C. Liang, Y. Liu, and K. Ross, View-upload decoupling: A redesign of multi-channel p2p video systems, in IEEE INFOCOM, 2726 -2730 (2009).
[15] D.Wu, Y. Liu, and K. Ross, Queuing network models for multi-channel p2p live streaming systems, in IEEE INFOCOM, 73 -81 (2009).
[16] C. Zhao, X. Lin, and C Wu, The streaming capacity of sparsely-connected P2P systems with distributed control, in Proc. of IEEE INFOCOM, 1449-1457 (2011).
[17] S. Sengupta, S. Liu, M. Chen, M. Chiang, J. Li, and P.A. Chou, Peer-to-Peer Streaming Capacity, IEEE Transactions on Information Theory, **57**, 5072-5087 (2011).
[18] C. Feng and B. Li, On large-scale peer-to-peer streaming systems with network coding, in Proc. of the 16th ACM international conference on Multimedia, Vancouver, British Columbia, Canada, 269-278 (2008).
[19] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, How to model an internetwork, in Proc. of IEEE INFOCOM, 594-602 (1996).

Shuai Zeng was born in 1982. He received the B.S. and M.S. degrees from the Chongqing University of Posts and Telecommunications (CQUPT), Chongqing, in 2003 and 2007, both in communication and information engineering. He is currently pursuing the Ph.D. degree with the School of Communication and Information Engineering, UESTC. His current research interests include network optimization, P2P networks and wireless networks.

Lemin Li received the BS degree in electrical engineering from Shanghai Jiaotong University, China in 1952. From 1952 to 1956, he was with the Department of Electrical Communications at Shanghai Jiaotong University. Since 1956 he has been with the University of Electronic Science and Technology of China (UESTC). From 1980.8 to 1982.8, he was a Visiting Scholar in the Department of Electrical Engineering and Computer Science at the University of California at San Diego, USA. He is currently a Professor of the School of Communications and Information Engineering, UESTC. He is a member of the Chinese Academy of Engineering. His present research interests are in the areas of communication networks.

Dan Liao received the Ph.D. degree from University of Electronic Science and Technology of China in 2007. He has been an associate professor at UESTC since 2009. His current research interests include P2P networks, wireless networks and cloud computing.