Noname manuscript No.
(will be inserted by the editor)

Locality-aware Streaming in Hybrid P2P-Cloud CDN Systems

Jian Zhao · Chuan Wu · Xiaojun Lin

the date of receipt and acceptance should be inserted later

Abstract The cloud CDN has been exploited as a cost-effective and elastic way for video streaming, where the content providers only pay for resources (*e.g.*, bandwidth, storage, Web service) that they use. As a server-client solution, the cost on the content providers, especially the bandwidth cost for delivering videos from the cloud to the users, is still high. To further mitigate the content distribution cost, it is promising to deploy the P2P streaming technology in conjunction with a cloud CDN, to construct a hybrid P2P-cloud CDN system. In this paper, we focus on optimal, locality-aware video-on-demand streaming solutions in a hybrid P2P-cloud CDN system, which achieve the best tradeoff between the costly bandwidth consumption on the cloud servers and the undesirable inter-ISP traffic incurred in the entire system. Especially, we characterize the demanded upload bandwidth in the cloud CDN and the incurred inter-ISP traffic through a number of stochastic and optimization models. We first apply a loss network model to derive the needed bandwidth capacity in the cloud CDN under any given chunk distribution pattern among the peer caches and any streaming request dispatching strategy among the ISPs, and derive the optimal peer caching and request dispatching strategies which minimize the bandwidth demand on the cloud CDN. We then investigate the necessary volume of inter-ISP traffic required to achieve the minimum cloud bandwidth, as well as the impact of limiting the inter-ISP traffic on the cloud bandwidth consumption. Based on the fundamental insights from our models and analytical results, we design a locality-aware, hybrid P2P-cloud streaming protocol, and validate its performance using extensive simulations under realistic settings.

Keywords Hybrid P2P-Cloud;Locality-awareness;Stochastic optimization;VoD streaming

Jian Zhao
Department of Computer Science, The University of Hong Kong
E-mail: jzhao@cs.hku.hk

Chuan Wu
Department of Computer Science, The University of Hong Kong
E-mail: cwu@cs.hku.hk

Xiaojun Lin
School of Electrical and Computer Engineering, Purdue University, West Lafayette
E-mail: linx@purdue.edu

1 Introduction

Owing to the cost effectiveness and elasticity of the cloud computing paradigm, more and more content providers start to exploit a cloud-based content distribution network (CDN) for content delivery, *e.g.*, Sohu Video [Soh(http://tv.sohu.com)], PPLive [PPL(http://www.pplive.tv)]. A cloud CDN, *e.g.*, Amazon CloudFront [Clo(http://aws.amazon.com/cloudfront/)], delivers contents using a global network of edge clouds, and automatically redirects requests for contents to the nearest edge cloud. Utilizing a cloud CDN is an easy way to distribute contents to end users with low latency and high data transfer speeds, while content providers pay only for what they use in terms of upload bandwidth, storage, and Web services over time.

Nevertheless, cloud-based content distribution is by nature a server-client solution, where the bandwidth consumption on the cloud servers increases significantly with the scale-up of the content distribution system. Based on the current pricing models of cloud systems [Clo(http://aws.amazon.com/cloudfront/#pricing)], the charges on incoming and outgoing traffic into and out of a cloud system are still quite high, leading to non-negligible, high bandwidth costs at the content providers.To further mitigate the content distribution cost, it has been promising to employ the peer-to-peer (P2P) technology in conjunction with a cloud CDN, to construct a hybrid P2P-cloud CDN system. Exploiting peers' mutual resource contribution, P2P content distribution enables significant reduction on bandwidth usage on the cloud servers. On the other hand, P2P content distribution can bring large volumes of inter-ISP traffic among regular network users, which significantly adds to the traffic relay cost of the Internet service providers (ISPs) [Inc.(2010)], risking ISPs' packet filtering and rate throttling. Hence, it is desirable to design a locality-aware P2P content distribution protocol in conjunction with a cloud CDN, where unnecessary inter-ISP traffic is reduced, while the bandwidth demand on the cloud servers is minimized.

A number of locality-aware P2P content distribution designs have been proposed, most of which advocate to connect peers to nearby neighbors in the same AS or ISP, in order to reduce inter-ISP traffic [Xie et al(2008)Xie, Yang, Krishnamurthy, Liu, and Silberschatz] [Picconi and Massoulie(2009)] [Magharei et al(2009)Magharei, Rejaie, Hilt, Rimac, and Hofmann]. Nonetheless, it is still left unknown what the relationship is between traffic localization and bandwidth consumption on the dedicated content distribution servers: How does the bandwidth demand on the servers change when the inter-ISP traffic is restricted to different levels? Given that content caching is a key component in P2P content distribution, what are the best caching strategies and request dispatching strategies, such that both the minimum inter-ISP traffic and the minimum server bandwidth consumption are achieved? The answers to these questions are critical for the design of a hybrid P2P-cloud content distribution protocol.

In this paper, we investigate optimal, locality-aware video-on-demand (VoD) streaming solutions in a hybrid P2P-cloud CDN system, which achieve the best tradeoff between the costly bandwidth consumption on the cloud servers and the undesirable inter-ISP traffic incurred in the entire system. Especially, we characterize the demanded upload bandwidth from the cloud CDN and the incurred inter-ISP traffic through a number of stochastic and optimization models. We first apply a loss network model to derive the needed bandwidth capacity in the cloud CDN under any given chunk distribution pattern among the peer caches and any streaming request dispatching strategy among the ISPs, and derive the optimal peer caching and request dispatching strategies which minimize the bandwidth demand from the cloud CDN. We then investigate the necessary volume of inter-ISP traffic required to achieve the minimum cloud bandwidth, as well as the impact of limiting the inter-ISP traffic on the cloud bandwidth consumption. Based on the fundamental insights from our models and analytical results, we design a locality-aware, hybrid P2P-cloud streaming protocol, and validate its performance using extensive simulations under realistic settings.

The remainder of the paper is organized as follows: We present the system model of hybrid P2P-cloud VoD streaming in Sec. 2. A loss network model is applied to derive the cloud bandwidth demand under any given chunk distribution and request dispatching strategy in Sec. 3, and the optimal conditions for peer caching and request dispatching to achieve the minimum cloud bandwidth are discussed in Sec. 4. We analyze the tradeoff between inter-ISP traffic and cloud bandwidth consumption in Sec. 5. We design a hybrid P2P-cloud VoD streaming protocol in Sec. 6, and perform simulation studies in Sec. 7. Related work are discussed in Sec. 8. Finally, Sec. 9 concludes the paper.

2 System Model and Notation

We consider a hybrid P2P-cloud VoD streaming system spanning M ISPs. The content provider stores its videos on the edge cloud in each of the ISPs, and uses the Web service of the edge cloud to serve videos to the requesting users (peers). P2P streaming technique is employed by each peer, which contributes upload bandwidth to serve other peers, and always tries to download the video chunks it needs from other peers first. When a peer can not obtain the requested chunks from other peers (due to lack of availability or bandwidth in the P2P overlay), it downloads them from the edge cloud in the same ISP. Fig. 1 illustrates the hybrid P2P-cloud VoD streaming system.

Multiple videos are provided in the VoD system. The videos are divided into chunks for storage and advertising to peer neighbors [Feng et al(2010)Feng, Li, and Liam]. We assume that the streaming playback rates of all videos are the same, and each chunk corresponds to one unit of playback time. Since a peer can watch any chunk in any video at a time, we consider the distribution of an aggregate collection of $J = |\mathcal{C}|$ chunks, $\mathcal{C} = \{c_1, c_2, ..., c_J\}$, regardless of which videos they belong to. Since the storage cost in a cloud platform is typically quite low [S3P(http://aws.amazon.com/s3/pricing/)], we assume that each edge cloud caches all the chunks, while the bandwidth usage at each edge cloud should be minimized.

Let F_m be the edge cloud that is located in ISP m, $1 \leq m \leq M$. Let Ψ denote the total upload bandwidth consumption in the cloud CDN, which is equivalent to the average number of chunks peers download from the cloud CDN in a unit time. Let N_m be the total number of peers in ISP m, which have installed the P2P VoD software. The average upload bandwidth among peers in ISP m is U_m. At

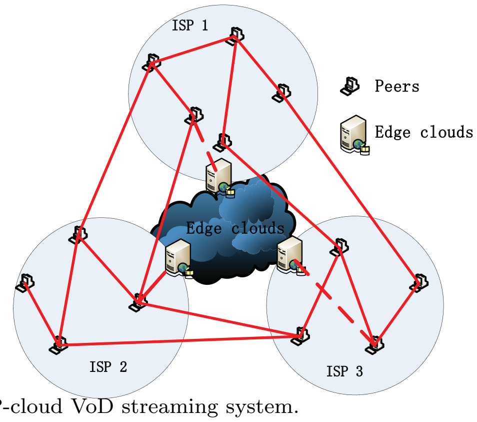

Fig. 1 A hybrid P2P-cloud VoD streaming system.

any time, a peer is either in the VoD streaming system or offline. We say that a peer is active when it participates in the streaming, and inactive when it is offline. Let π_0 denote the probability that a peer is inactive. The probability that a peer is active is $1-\pi_0$. The average numbers of active and inactive peers are $N_m(1-\pi_0)$ and $N_m\pi_0$, respectively.

2.1 Peer Cache States

Each peer has a buffer which can cache at most B chunks. There are in total $W = J!/(J-B)!$ possible cache states, each of which is a permutation of B chunks from the overall J chunks in the system. We associate each possible cache state with a base-$(J+1)$ number, *e.g.*, cache state $(c_J, c_{J-1}, \ldots, c_{J-B+2}, c_{J-B+1})$ is associated with the number $J(J+1)^{B-1}+(J-1)(J+1)^{B-2}+\ldots+(J-B+2)(J+1)+(J-B+1)$. We can hence order cache states in the ascending order according to these numbers.

Let s_i denote the ith cache state in this order, and $\mathcal{W}$ be the set of all different cache states, with $W = |\mathcal{W}|$. Let $N_m^{(i)}$ be the number of active peers with cache state s_i in ISP m, and γ_i be the stationary probability that a peer's cache is in state s_i. In the entire system, the proportion of peers caching chunk c_j is $\rho_j = \sum_{i:c_j\in s_i}\gamma_i$, and the number of peers caching chunk c_j in ISP m is $N_m\rho_j$.

2.2 Chunk Request

Without loss of generality, suppose the video playback rate is 1 chunk per unit time. Active peers send out requests for interested chunks at a rate equal to the video playback rate, and each chunk can be downloaded using 1 unit of upload bandwidth in a unit time. This assumption is reasonable when we practically focus on a system where the overall peer upload bandwidth is not enough to support the streaming at all the peers, and thus extra bandwidth provisioning from the edge clouds are needed. In this case, using a larger request rate than the playback rate and buffering chunks in advance may add to the bandwidth usage in edge clouds, leading to an increased operating cost, which is not necessary.

Suppose the number of peers N_m in each ISP m is much larger than 1. The request generation process at each active peer in the ISP can be considered as a general renewal process with relative small intensity. According to the Palm-Khintchine theorem [Heyman and Sobel(2004)], the aggregate requests for chunks generated by all peers in ISP m can be approximately as a Poisson Process, with a request rate $N_m(1-\pi_0)$.

We assume peers have the same chunk preference distribution. Let $(\pi_1,\dots,\pi_J)$ denote the chunk preference distribution, π_j denotes the stationary probability that a peer in the VoD system requests chunk j, $j=1,\dots,J$. We assume that the probability that the requested chunk c_j is in the peer's cache (*i.e.*, in the case that a peer replays a previously downloaded chunk) equals to the proportion of peers caching chunk c_j, *i.e.*, ρ_j. When a chunk is not cached at a peer, the peer issues a chunk request to download it from others. The request rate for chunk j arising from peers in ISP m, which leads to the download traffic, is $r_{m,j} = N_m\pi_j(1-\rho_j) = N_m(\pi_j - \pi_j\rho_j)$. The overall request rate for all chunks generated by peers in ISP m, is $r_m = \sum_{j=1}^{J} r_{m,j} = N_m \sum_{j=1}^{J} \pi_j(1-\rho_j) = N_m(1-\pi_0-\sum_{j=1}^{M}\pi_j\rho_j)$.

The chunk requests from ISP m can be served by peers in the same ISP or other ISPs. Let a_{lm} be the proportion of chunk requests sent from ISP l to ISP m. The number of requests for chunk j dispatched into ISP m in one unit time (*i.e.*, the request rate) is $\sum_{l=1}^{M} a_{lm} r_{l,j}$. The overall request rate sent into ISP m, for all the chunks, is $\sum_{j=1}^{J}\sum_{l=1}^{M} a_{lm} r_{l,j} = \sum_{l=1}^{M} a_{lm} r_l$.

In our model, the distribution $(\pi_0,\pi_1,\dots,\pi_J)$ could be time varying. This could capture the peer churning in the real P2P system.

2.3 Cloud Bandwidth Usage

VoD streaming is delay-sensitive. When a chunk request dispatched into an ISP cannot be served by peers in that ISP, the requester will download the chunk from the edge cloud located in its own ISP. Let $L_{m,j}$ be the miss rate of requests for chunk j in ISP m, *i.e.*, the steady-state probability that a request for chunk j distributed to ISP m is resent to an edge cloud. The average miss rate of requests sent to ISP m is:

$$L_m = \frac{\sum_{j=1}^{J} L_{m,j} \sum_{l=1}^{M} a_{lm} r_{l,j}}{\sum_{l=1}^{M} a_{lm} r_l}. \tag{1}$$

The chunk requests resent to edge clouds incur upload bandwidth usage from the cloud CDN, to serve the requested chunks. We compute the overall bandwidth usage at the cloud CDN as the average number of chunk requests sent to the edge clouds per unit time:

$$\Psi = \sum_{l=1}^{M}\sum_{j=1}^{J}\sum_{m=1}^{M} r_{l,j} a_{lm} L_{m,j} = \sum_{m=1}^{M} L_m \sum_{l=1}^{M} a_{lm} r_l. \tag{2}$$

Table 1 Important Notations

N	total number of peers in the system.
M	number of ISPs.
N_m	number of peers in ISP m.
$N_m^{(i)}$	number of active peers in ISP m with cache state s_i.
F_m	the nearest edge cloud to peers in ISP m.
U_m	average peer upload bandwidth in ISP m.
$\mathcal{C}$	the set of all chunks shared in VoD system.
$\mathcal{A}$	any subset of $\mathcal{C}$.
J	the number of chunks shared in VoD.
B	one peer's cache size.
$\mathcal{W}$	the set of all possible cache states of peers.
W	the number of different cache states.
π_0	the probability that a peer is inactive.
π_j	the probability that a peer requests chunk j.
ρ_j	the proportion of peers that have cached chunk j.
$r_{m,j}$	request rate for chunk j generated by peers in ISP m.
a_{lm}	the fraction of requests routed from ISP l to ISP m.
r_l	the total rate of requests generated by peers in ISP l.
$L_{m,j}$	the miss rate for chunk j in ISP m.
L_m	the average chunk miss rate in ISP m.
Ψ	the total upload bandwidth consumption in the cloud CDN.
T_m	the inter-ISP traffic flowing out of ISP m.
T	the total inter-ISP traffic in the system.

2.4 Inter-ISP Traffic

When peers in one ISP serve chunk requests from other ISPs, inter-ISP traffic is incurred. The inter-ISP traffic per unit time for ISP m to send chunks to other ISPs is $T_m = \sum_{l=1,l\neq m}^{M} a_{lm} r_l (1 - L_m)$, *i.e.*, the number of requests from other ISPs served by ISP m. The overall amount of inter-ISP traffic in the entire system is $T = \sum_{m=1}^{M} T_m$.

For ease of reference, important notation in this paper is summarized in Table 1.

3 Relating Cloud Bandwidth Demand and Cache State Distribution: A Loss Network Approach

We next apply a loss network model to compute the steady-state chunk miss rate in any ISP, given the distribution of peer cache states and the number of requests distributed to the ISP. Based on the chunk miss rate in the P2P overlay, we can derive the total bandwidth demand from the cloud CDN. The model in this section inspires our optimal peer cache strategies in Sec. 4.

3.1 The Loss Network Model

A classical loss network is used to model service of calls in a telephone network: The telephone network consists of a set of links. Each link carries a number of circuits, which is the capacity of the link. When the telephone network receives

Table 2 Mapping Between Our VoD System and Classical Loss Network

Our VoD System	**Classical Loss Network**
Chunk requests	Call requesting
Chunk request rate	Call requesting rate
A subset of chunks $\mathcal{A} \subseteq \mathcal{C}$	A link
All subsets $\mathcal{A}$ containing a specific chunk j	A route
Total upload bandwidth of peers caching any chunk in $\mathcal{A}$	A link's capacity

a call, it selects some links and allocates the circuits on those links to connect both ends of the call. The set of selected links to connect a call is called a route. One or more circuits of some links are necessary to be connected to offer a route. If there are fewer circuits than necessary on a link for the route of a call, the call is blocked and lost [Kelly(1991)]. The loss network model provides the loss probability of calls on routes, and we apply it to derive the chunk miss rate in our VoD streaming system. The acceptance and rejection of chunk requests in our system are analogous to those of calls in a telephone network. The mapping between our streaming system and the loss network is summarized in Table 2. A subset of chunks $\mathcal{A} \subseteq \mathcal{C}$ is mapped to a link $\mathcal{A}$ in the loss network. The capacity of link $\mathcal{A}$ is the sum of the upload capacity of all peers whose chunk cache state has a non-empty intersection with $\mathcal{A}$. Under feasible allocation, the total upload bandwidth for requests of chunks in $\mathcal{A}$ should not exceed the capacity of link $\mathcal{A}$. The system can attempt to serve a new chunk request while ensuring the existing requests can obtain enough upload bandwidth, by reallocating the upload bandwidth in the P2P overlay. For each chunk j, the set of links $\mathcal{A}$ that contain j form a route P_j. A request for chunk j will consume one unit capacity from every link in this route P_j, *i.e.*, one unit capacity from any subset $\mathcal{A}$ that contains chunk j. If any link $\mathcal{A}$ containing chunk j does not have available capacity, the new arriving request for chunk j will be rejected.

Given the cache state distribution at peers, the chunk miss rate for chunk j in ISP m, $L_{m,j}$, can be calculated as follows [Jung et al(2008)Jung, Lu, Shah, Sharma, and Squillante]: The requests being serviced experience a delay of 1 time unit (service time). The missed requests experience a delay of 0. The average delay that chunk requests experience is $D_{m,j} = (1 - L_{m,j}) \cdot 1 + L_{m,j} \cdot 0 = 1 - L_{m,j}$. Let $n_{m,j}$ be the number of requests for chunk j being served concurrently in ISP m at any given time. Applying Little's law, we can obtain $\mathbf{E}[n_{m,j}] = D_{m,j} \sum_{l=1}^{M} a_{lm} r_{l,j}$, which yields $L_{m,j} = 1 - \mathbf{E}[n_{m,j}] / \sum_{l=1}^{M} a_{lm} r_{l,j}$.

Hence, the problem of deriving miss rate $L_{m,j}$ is equivalent to computing the expected value of the number of concurrently served requests, $\mathbf{E}[n_{m,j}]$. Define the vectors $\mathbf{N}_m = (N_m^{(1)}, \ldots, N_m^{(W)})$ and $\mathbf{n}_m = (n_{m,j}, 1 \le j \le J)$. $\mathbf{n}_m$ satisfies the condition $\forall \mathcal{A} \subseteq \mathcal{C}, \sum_{c_j \in \mathcal{A}} n_{m,j} \le U_m \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)}$. Let $\mathcal{N}_m$ be the set of all available vector $\mathbf{n}_m$. The detailed distribution of vector $\mathbf{n}_m$ is given as follows [Kelly(1991)]:

$$\pi(\mathbf{n}_m) = G(\mathcal{N}_m)^{-1} \prod_{j=1}^{J} \frac{(\sum_{l=1}^{M} a_{lm} r_{l,j})^{n_{m,j}}}{n_{m,j}!}$$

It has been pointed out that the computational complexity of the stationary distribution of the number of concurrently served requests is ♯P-complete in general [Jung et al(2008)Jung, Lu, Shah, Sharma, and Squillante]. One efficient and approximate approach for computing the stationary chunk miss rate is the 1-point approximate algorithm, in which the state with the maximum probability $n^*_{m,j}$ is taken as a surrogate of $\mathbf{E}[n_{m,j}]$. Under this approximation, the error of the loss rate $L_{m,j}$ diminishes to 0 as quickly as $\frac{1}{N_m}$ [Kelly(1991)]. We relax the integer variable $n_{m,j}$ to a real variable $x_{m,j}$. The following optimization problem is to solve for the relaxed value $x_{m,j}$ of state $n^*_{m,j}$ with the maximum probability in ISP m, $1 \le j \le J$ [Tan and Massoulié(2011)]:

$$\max \sum_{j=1}^{J} \left[x_{m,j} \log(\sum_{l=1}^{M} a_{lm} r_{l,j}) - x_{m,j} \log x_{m,j} + x_{m,j} \right]$$

$$\text{subject to: } \forall \mathcal{A} \subseteq \mathcal{C}, \sum_{c_j \in \mathcal{A}} x_{m,j} \le U_m \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)}.$$

$\mathcal{A}$ denotes any subset of chunk set $\mathcal{C}$. The objective function is to maximize the probability of state $x_{m,j}, 1 \le j \le J$. The constraints mean that the total served requests for a link $\mathcal{A}$ in the loss network should not exceed its capacity.

The derived value of $x_{m,j}$ is used to approximate the value of $\mathbf{E}[n_{m,j}]$, which is a standard technique to derive the loss rate in a loss network model [Kelly(1991)] [Tan and Massoulié(2011)]. When $N_m^{(i)} \to \infty$, the loss rate obtained from the approximation approaches that of the actual values. We can then calculate the average chunk miss rate in ISP m as

$$L_m = \frac{\sum_{j=1}^{J} (1 - x_{m,j} / \sum_{l=1}^{M} a_{lm} r_{l,j}) \sum_{l=1}^{M} a_{lm} r_{l,j}}{\sum_{l=1}^{M} a_{lm} r_l} = 1 - \frac{\sum_{j=1}^{J} x_{m,j}}{\sum_{l=1}^{M} a_{lm} r_l}, \tag{3}$$

and the total upload bandwidth consumption in the cloud CDN according to Eqn. (2).

3.2 Deriving Average Chunk Miss Rate

To derive a solution to the above convex optimization problem, we solve the set of KKT conditions of the problem. Let $\epsilon_{\mathcal{A}}$ be Lagrangian dual variables associated with the constraints. The KKT conditions are:

$$\forall \mathcal{A} \subseteq \mathcal{C}, \sum_{c_j \in \mathcal{A}} x_{m,j} \le U_m \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)}, \tag{4}$$

$$\forall \mathcal{A} \subseteq \mathcal{C}, \epsilon_{\mathcal{A}} \ge 0, \tag{5}$$

$$\forall \mathcal{A} \subseteq \mathcal{C}, \epsilon_{\mathcal{A}} (U_m \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)} - \sum_{c_j \in \mathcal{A}} x_{m,j}) = 0, \tag{6}$$

$$x_{m,j} = (\sum_{l=1}^{M} a_{lm} r_{l,j}) \exp(- \sum_{c_j \in \mathcal{A}, \mathcal{A} \subseteq \mathcal{C}} \epsilon_{\mathcal{A}}), 1 \le j \le J. \tag{7}$$

The number of KKT conditions grows exponentially with the number of chunks, which makes it computationally complex to solve the KKT conditions directly. Nevertheless, according to (3), in order to derive the average chunk miss rate in

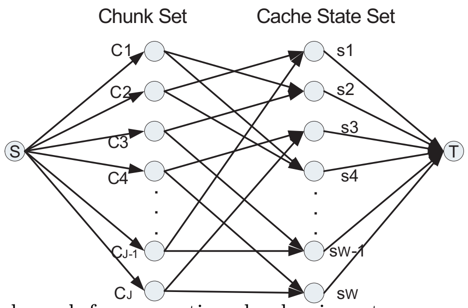

Fig. 2 The constructed graph for computing chunk miss rate.

ISP m, we only need to know the optimal value of the sum, $\sum_{j=1}^{J} x_{m,j}$, which is approximately the expected number of concurrently served chunk requests in the ISP at any time. We next show that this number can be derived by solving a maximum bipartite flow problem, for which an efficient push-relabel algorithm exists [Negruseri et al(2009)Negruseri, Pasoi, Stanley, and Stein].

We construct a bipartite graph, with one set of J nodes representing the set of chunks $c_1, c_2, ..., c_J$, and the other set of W nodes representing the set of peer cache states $s_1, s_2, ..., s_W$. We also add a source node $\mathcal{S}$, connecting to all nodes representing chunks, and a destination node $\mathcal{T}$, connecting to all nodes representing the cache states. An illustration of the graph is given in Figure 2. If a cache state contains a chunk, there is a directed edge connecting the chunk to the cache state. Let the capacity of edge $(\mathcal{S}, c_i)$ be the number of requests for chunk c_i dispatched to ISP m per unit time, which is $\sum_{l=1}^{M} a_{lm} r_{l,i}$, and the capacity of edge $(s_i, \mathcal{T})$ be the total upload bandwidth of active peers in cache state s_i in ISP m, which is $N_m^{(i)} U_m$. The capacity of an edge $(c_i, s_j), \forall c_i \in \mathcal{C}, \forall s_j \in \Theta$, is unlimited. We then derive the following result, with detailed proof given in Appendix A.

Theorem 1. *The sum of solutions of the KKT conditions (4) (5) (6) (7),* i.e., $\sum_{j=1}^{J} x_{m,j}$*, is equal to the maximum flow of the constructed graph in Figure 2.*

By solving the maximum flow problem [Negruseri et al(2009)Negruseri, Pasoi, Stanley, and Stein] and deriving $\sum_{j=1}^{J} x_{m,j}$, we can then derive the average chunk miss rate in ISP m, *i.e.*, L_m, according to (3). With L_m, we can derive the total bandwidth consumption on the cloud CDN using Eqn. (2).

4 Optimal Caching and Request Dispatching Conditions

The loss network framework enables us to calculate the bandwidth demand on the cloud CDN, under any given cache state distribution and chunk request rates into the ISPs. We are especially interested in the minimum cloud bandwidth needed to sustain the P2P streaming system, under the optimal peer caching and request distribution strategies.

4.1 Optimal Caching Condition

We define an ISP m's workload, η_m, as the ratio of the overall chunk request rate to the total peer upload bandwidth in the ISP, $\eta_m = \sum_{l=1}^{M} a_{lm} r_l / [N_m (1 -$

$\pi_0)U_m]$. Given the request rates for different chunks routed into the ISP (*i.e.*, $\sum_{l=1}^{M} a_{lm} r_{l,j}$, $\forall c_j \in \mathcal{C}$), the optimal cache state distribution should enable as many chunk requests as possible to be served using peer upload bandwidth in the ISP. The ideal minimum chunk miss rate in ISP m is:

$$L_m = \max\{0, 1 - \frac{1}{\eta_m}\} = \max\{0, \frac{\sum_{l=1}^{M} a_{lm} r_l - U_m N_m (1-\pi_0)}{\sum_{l=1}^{M} a_{lm} r_l}\}. \tag{8}$$

The following lemma gives the optimal cache condition in ISP m to achieve this minimum chunk miss rate, given the chunk request rates $\sum_{l=1}^{M} a_{lm} r_{l,j}$, $\forall c_j \in \mathcal{C}$.

Lemma 1. *If the numbers of active peers at cache state s_i in ISP m,* i.e., $N_m^{(i)}, 1 \leq i \leq W$*, satisfy the following inequalities*

$$\forall \mathcal{A} \subseteq \mathcal{C}, \sum_{c_j \in \mathcal{A}} \sum_{l=1}^{M} a_{lm} r_{l,j} \leq \eta_m U_m \sum_{i: \mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)}, \tag{9}$$

then the minimum chunk miss rate $L_m = \max\{0, 1 - \frac{1}{\eta_m}\}$ can be achieved in this ISP.

Lemma 1 shows that if the rate to each link $\mathcal{A}$ in the loss network is no greater than η_m times its capacity, then the minimum chunk miss rate can be achieved. The proof can be found in Appendix B. The following Lemma 2 states a concrete optimal cache placement strategy, which satisfies condition (9) in Lemma 1, with proof in Appendix C.

Lemma 2. *If chunk c_j is randomly cached in a proportion of $\rho_j = B\pi_j$ peers in ISP m, $\forall c_j \in \mathcal{C}$, then condition (9) is satisfied, and the minimum chunk miss rate $L_m = \max\{0, 1 - \frac{1}{\eta_m}\}$ can be achieved in this ISP.*

We also prove that the stationary cache state distribution under the Least Recently Used (LRU) cache replacement strategy satisfies the optimal cache distribution proposed in Lemma 2, and therefore, a LRU peer caching strategy can achieve the minimum chunk miss rate. This is consistent with the observations in Wu *et al.*'s work [Wu and Li(2009)], which show that LRU cache replacement performs as well as the optimal cache replacement using simulations, while we provide rigorous theoretical proof of the property (Lemma 3) in Appendix D.

Lemma 3. *If peers in ISP m apply the LRU cache replacement strategy, the minimum chunk miss rate $L_m = \max\{0, 1 - \frac{1}{\eta_m}\}$ can be achieved in this ISP in the steady state.*

Under the optimal peer caching strategies, the upload bandwidth consumption on the cloud CDN is given by

$$\Psi = \sum_{m=1}^{M} L_m \sum_{l=1}^{M} a_{lm} r_l = \sum_{m=1}^{M} \max\{\sum_{l=1}^{M} a_{lm} r_l - U_m N_m (1-\pi_0), 0\}. \tag{10}$$

4.2 Optimal Request Dispatching Condition

The optimal caching conditions above are derived given the request rates for different chunks dispatched into each ISP m, $\sum_{l=1}^{M} a_{lm} r_{l,j}$, $\forall c_j \in \mathcal{C}$. We next study the best request distribution strategies (a_{lm}'s, $l = 1, \ldots, M, m = 1, \ldots, M$), which minimizes (10) as follows:

$$\begin{aligned} \min \quad & (10) \\ \text{subject to:} \quad & \sum_{m=1}^{M} a_{lm} = 1, 1 \le l \le M, && (11) \\ & a_{lm} \ge 0, 1 \le l \le M, 1 \le m \le M. && (12) \end{aligned}$$

We derive the sufficient conditions that a_{lm}'s should satisfy, to achieve the minimum cloud bandwidth usage in (10), considering a system where the overall peer upload bandwidth is smaller than the bandwidth needed to serve all the chunk requests, *i.e.*, $\sum_{m=1}^{M} U_m N_m (1 - \pi_0) < \sum_{l=1}^{M} r_l$ (otherwise, there is no bandwidth consumed in the cloud CDN and the case is of less interest). We have

$$\begin{aligned} (10) &\ge \max\{\sum_{m=1}^{M} [\sum_{l=1}^{M} a_{lm} r_l - U_m N_m (1 - \pi_0)], 0\} \\ &= \sum_{m=1}^{M} [\sum_{l=1}^{M} a_{lm} r_l - U_m N_m (1 - \pi_0)]. \qquad (13) \end{aligned}$$

If we can find a_{lm}'s satisfying constraints (11), (12), as well as the following:

$$\sum_{l=1}^{M} a_{lm} r_l \ge U_m N_m (1 - \pi_0), 1 \le m \le M, \qquad (14)$$

i.e., the request rate distributed into each ISP m is no smaller than the overall peer bandwidth in the ISP, then we have

$$(10) = \sum_{m=1}^{M} [\sum_{l=1}^{M} a_{lm} r_l - U_m N_m (1 - \pi_0)],$$

which achieves the minimum cloud bandwidth consumption according to (13).

(14) together with (11) and (12) are referred to as the optimal request distribution conditions. An optimal request distributing strategy should generate a set of a_{lm}'s satisfying these conditions, and decide the request rate for each chunk j, to be dispatched into each ISP m, as $\sum_{l=1}^{M} a_{lm} r_{l,j}$. Then inside each ISP, if an optimal peer caching strategy is applied, the minimum overall cloud bandwidth consumption can be achieved.

5 Tradeoff between Inter-ISP Traffic and Cloud Bandwidth Consumption

We now address the following questions: What is the minimum volume of necessary inter-ISP traffic, in order to achieve the minimum cloud bandwidth consumption? What is the cloud bandwidth needed when there is no inter-ISP traffic allowed? How does the change of inter-ISP traffic volume influence the cloud bandwidth consumption? In our following analysis, we always assume that peers apply an optimal caching strategy inside each ISP, when the request rates into the ISP are given.

5.1 Minimum Inter-ISP Traffic for Minimum Cloud Bandwidth Usage

The minimum inter-ISP traffic needed to achieve the minimum bandwidth consumption at the cloud CDN can be derived from the following optimization problem:

$$\begin{aligned}&\min_{a_{ml}\text{'s}} \sum_{m=1}^{M}(1-L_m)\sum_{l=1,l\neq m}^{M} a_{lm}r_l\\ &\text{subject to: constraints (11)(12)(14).}\end{aligned} \tag{15}$$

The objective function corresponds to the overall inter-ISP chunk delivery traffic $T=\sum_{m=1}^{M}T_m$, where $a_{lm}r_l$ is the number of chunk requests distributed from ISP l to ISP m in one time unit, and $a_{lm}r_l(1-L_m)$ gives the actual number of chunks to be served from peers in ISP m to peers in ISP l in one time unit, where L_m is a function of a_{lm}'s given by (8). The constraints in (14) guarantee that a_{lm}'s achieve minimum cloud bandwidth consumption. We give the optimal solutions a^*_{lm}'s to this optimization problem in the following theorem.

Theorem 2. *The minimum inter-ISP traffic to achieve the minimum cloud bandwidth consumption is* $T^*=\sum_{m=1}^{M}T^*_m=\sum_{m=1}^{M}\sum_{l\neq m}a^*_{lm}r_l(1-L^*_m)$, *where*

$$L^*_m=\max\{0,\frac{\sum_{l=1}^{M}a^*_{lm}r_l-U_mN_m(1-\pi_0)}{\sum_{l=1}^{M}a^*_{lm}r_l}\}$$

and a^*_{lm}*'s are given as follows:*
(i) For ISPs with sufficient upload bandwidth, i.e., $I_m=U_mN_m(1-\pi_0)-r_m\geq 0$,

$$a^*_{lm}=\begin{cases}1: & l=m,\\ 0: & l\neq m, I_l\geq 0,\\ \min\{\frac{I_m}{-\sum_{s,I_s<0}I_s},\frac{I_m}{\sum_{t,I_t>0}I_t}\}(\frac{-I_l}{r_l}): & l\neq m, I_l\leq 0.\end{cases}$$

(ii) For ISPs with insufficient upload bandwidth, i.e., $I_m=U_mN_m(1-\pi_0)-r_m<0$,

$$a^*_{lm}=\begin{cases}\max\{(1-\frac{\sum_{t,I_t>0}I_t}{-\sum_{s,I_s<0}I_s})(\frac{-I_m}{r_m})+\frac{U_mN_m(1-\pi_0)}{r_m},\\ \qquad\frac{U_mN_m(1-\pi_0)}{r_m}\}: & l=m;\\ 0: & l\neq m, 1\leq l\leq M.\end{cases}$$

Detailed proof is given in Appendix E. The implications of a^*_{lm}'s values are as follows: When an ISP m has sufficient peer upload bandwidth ($I_m=U_mN_m(1-\pi_0)-r_m\geq 0$), its generated requests are served by its own upload bandwidth, and no requests are distributed outside the ISP. The requests distributed into the ISP do not exceed the upload bandwidth excess. $L^*_m=0,\forall I_m\geq 0$. When an ISP m's total peer upload bandwidth is not sufficient ($I_m=U_mN_m(1-\pi_0)-r_m<0$), it distributes part of requests to other ISPs with sufficient bandwidth in proportion to their extent of bandwidth excess, *i.e.*, a fraction of $I_l\cdot\min\{\frac{1}{-\sum_{s,I_s<0}I_s},\frac{1}{\sum_{t,I_t>0}I_t}\}$ to ISP l. If the overall peer upload bandwidth exceeds or equals to the bandwidth demand for chunk requests, $L^*_m=0,\Psi_m=0,\forall I_m<0$; if the overall peer upload bandwidth is smaller than the bandwidth demand for chunk requests, $L^*_m>0,\Psi_m>0,\forall I_m<0$.

We can get the following conclusions from Theorem 2:

- Under the following two cases, the minimum inter-ISP traffic can be 0 when the minimum cloud bandwidth consumption is achieved, *i.e.*, a win-win situation for minimizing cloud bandwidth usage in edge clouds and minimizing inter-ISP traffic among ISPs: (i) all ISPs have sufficient peer upload bandwidth, i.e., $\forall m, I_m = U_m N_m(1-\pi_0) - r_m \geq 0$; (ii) all ISPs have insufficient peer upload bandwidth, i.e., $\forall m, I_m = U_m N_m(1-\pi_0) - r_m < 0$.
- A more general case is that there exist both ISPs with sufficient peer upload bandwidth ($I_m \geq 0$), and those with insufficient peer upload bandwidth ($I_m < 0$). For ISP m with insufficient upload bandwidth, when the minimum cloud bandwidth usage is achieved, the minimum inter-ISP traffic flowing out can be 0, *i.e.*, $T^*_m = 0, \forall I_m < 0$. For ISP m with sufficient upload bandwidth, the minimum cloud bandwidth usage at its edge cloud is $\Psi_m = 0$, the minimum inter-ISP traffic flowing out of it is $\min\{1, -\sum_{l,I_l<0} I_l / \sum_{t,I_t>0} I_t\} I_m$.
- The traffic flowing out of one ISP is proportional to $I_m = U_m N_m(1-\pi_0) - N_m(1-\pi_0-\sum_{j=1}^{M}\pi_j\rho_j) = N_m[(U_m-1)(1-\pi_0)+\sum_{j=1}^{M}\pi_j\rho_j]$. Hence, the inter-ISP traffic flowing out of ISP m is larger when more peers are active (π_0 is smaller), the average peer upload bandwidth is larger..

5.2 Cloud Bandwidth Usage Under No Inter-ISP Traffic Allowed

When there is no inter-ISP traffic allowed, no chunk requests are distributed out of ISPs, *i.e.*, $a_{mm} = 1, 1 \leq m \leq M$, $a_{lm} = 0, l \neq m$, the minimum total cloud bandwidth usage in the entire system is,

$$\Psi = \sum_{m=1}^{M} \max\{0, r_m - U_m N_m(1-\pi_0)\}.$$

5.3 Impact of Inter-ISP Traffic on Cloud Bandwidth Usage

We next investigate how the volume of inter-ISP traffic affects the cloud bandwidth usage in cloud CDN. Suppose the maximally allowed inter-ISP traffic in the system is T^c. If $T^c \geq T^*$ (given in Theorem 2), the ISP-aware chunk request distributing algorithm above can be applied to achieve the minimum cloud bandwidth usage in the cloud CDN with an inter-ISP traffic volume at $T^* \leq T^c$. If $T^c < T^*$, the minimum cloud bandwidth usage achievable can be derived by the following optimization problem:

$$\begin{aligned} &\min (10) \\ \text{subject to: } &\sum_{m=1}^{M}(1-L_m)\sum_{l\neq m} a_{lm} r_l = T^c, \\ &\text{constraints (11)(12).} \end{aligned}$$

Let a^c_{lm}'s denote the optimal solutions to the above optimization problem.

When $0 < T^c < T^*$, for ISPs with insufficient bandwidth to serve its own requests ($I_m < 0$), fewer requests are distributed to other ISPs in case that the allowed inter-ISP traffic is T^c, than those in case that the allowed inter-ISP traffic is T^*, *i.e.*, $a^*_{mm} < a^c_{mm} < 1$; for ISPs with sufficient bandwidth ($I_m \geq 0$), all requests are served in the ISP, *i.e.*, $a^c_{mm} = 1$. Hence, the rates of requests distributed into ISPs with insufficient bandwidth ($I_m < 0$) exceed their respective upload bandwidth, *i.e.*, $a^c_{mm} r_m > a^*_{mm} r_m \geq U_m N_m (1 - \pi_0)$, $L_m > 0$. the rates of requests distributed into ISPs with sufficient bandwidth ($I_m \geq 0$) are smaller than their respective upload bandwidth, *i.e.*, $\sum_{l=1}^{M} a^c_{lm} r_l \leq \sum_{l=1}^{M} a^*_{lm} r_l \leq U_m N_m (1 - \pi_0)$, $L_m = 0$. The minimum cloud bandwidth usage is:

$$
\begin{aligned}
\Psi &= \sum_{m, I_m < 0} [\sum_{l=1}^{M} a^c_{lm} r_l - U_m N_m (1 - \pi_0)] \\
&= - \sum_{m, I_m < 0} I_m - T^c.
\end{aligned}
$$

We can see that when the inter-ISP traffic is constrained below T^*, the system-wide chunk miss rate increases linearly with the decrease of the allowed inter-ISP traffic volume.

6 A Hybrid P2P-Cloud Streaming Protocol

We next design a locality-aware VoD streaming protocol for hybrid P2P-cloud CDN systems, which implements peer caching and request routing strategies that fulfill the optimal conditions we derived in Sec. 4, and achieves minimum inter-ISP traffic as well as minimum cloud bandwidth usage in the entire system.

The locality-aware P2P VoD protocol is based on a typical BitTorrent-like P2P VoD streaming protocol (*e.g.*, [Yang et al(2010)Yang, Chow, Golubchik, and Bragg]), with chunk requests sent from peers to neighbors in different ISPs with different probabilities, guided by the optimal request distribution strategy in Theorem 2. Like in most other P2P systems, we assume that there is one tracker server provided by the content provider (which can be implemented by server(s) or virtual machine(s) in the cloud as well), which provides to each peer IP addresses of other peers and the entry point to the edge cloud in the same ISP.

Tracker Protocol. The tracker maintains information of the active peers in the system in M lists according to the ISPs they belong to (derived based on their IP addresses), $ActivePeerSet_1$, $\ldots$, $ActivePeerSet_M$, and records the number of active peers in each ISP, using $N_a(m), 1 \leq m \leq M$. Peers within each active peer set are sorted according to their arrival times.

In addition, the tracker uses vector $\mathbf{r}$ and $\mathbf{I}$ to record chunk request rate information and excess peer upload bandwidth information of the ISPs. It sets initial values for chunk popularity distribution $\pi_j, 0 \leq j \leq J$ based on statistical results from the past video viewing history, and updates the chunk popularity distribution periodically as follows: Each peer records the chunks it requested in a frame of t_0 time slots before the current time slot (the length of a time slot is 1 unit time), using vector ($playedchunk(j)$, $1 \leq j \leq J$), and sends the vector to the tracker every t_0 time slots, where $t_0 >> 1$ is a protocol parameter. Based on the vectors received, the tracker counts the proportion of chunk j being requested as the new value of π_j. Then the tracker updates each ISP's chunk request rate information $r(m) = N_a(m)[1 - \sum_{j=1}^{J} \pi_j \rho_j]$, and each ISP's excess bandwidth information $I(m) = N_a(m)U_m - r(m)$. Here ρ_j is decided by the peer caching strategy employed, *e.g.*, $\rho_j = B\pi_j$ when peers apply the LRU caching strategy.

When a peer p asks the tracker for a list of neighbors, the tracker sends M sets of neighbors back, $PeerSet_1, \ldots, PeerSet_M$, where $PeerSet_i$ contains a list of peers from $ActivePeerSet_i$ with the closest arrival times to peer p's. The tracker also sends information on the entry point to the edge cloud located in the same ISP as peer p, the chunk request information $\mathbf{r}$ and the excess bandwidth information $\mathbf{I}$ with a time stamp to the peer.

Peer protocol. A peer divides its neighbors into M sets, $NeighborSet_m, 1 \leq m \leq M$, according their ISPs. Each peer maintains a queue of received chunk requests. Peers exchange information among each other on chunk availability and current request queue sizes using a set of protocol messages similar to those in the BitTorrent protocol, *e.g., Have*.

A peer in an ISP m where the total peer upload bandwidth is sufficient to serve chunk requests from the same ISP ($I(m) \geq 0$), sends out its chunk request to a neighbor in the same ISP, which caches the chunk and currently has the shortest queue of requests received, among all those caching the needed chunk. If no neighbors in the same ISP have the chunk, the peer sends the request to the edge cloud in its ISP.

A peer in an ISP m where the total peer upload bandwidth is insufficient to serve chunk requests from the same ISP, may send the request to the same ISP with probability $\max\{(1 - \frac{\sum_{t,I(t)>0} I(t)}{-\sum_{s,I(s)<0} I(s)})\ (\frac{-I(m)}{r(m)}) + \frac{I(m)+r(m)}{r(m)}, \frac{I(m)+r(m)}{r(m)}\}$, or to an ISP l with sufficient peer upload bandwidth ($I(l) > 0$) with probability $\min\{\frac{I(l)}{-\sum_{s,I(s)<0} I(s)}, \frac{I(l)}{\sum_{t,I(t)>0} I(t)}\}(\frac{-I(m)}{r(m)})$. When the ISP to which the chunk request should be sent is determined, the peer chooses among its neighbors in that ISP one which caches the chunk and has the shorted request queue, and sends the request to that specific neighbor. If no neighbors in the selected ISP have the requested chunk, the peer downloads the chunk from the edge cloud in its own ISP.

A deadline of service is stamped to each request sent out from a peer, indicating the playback deadline of the requested chunk at the peer. Each peer sorts its queue of received chunk requests by the deadlines and serves the requests based on their urgency. The requests which can not be served before their deadlines are dropped, and a requester will resend its chunk request to the edge cloud if no response is received before the deadline.

Bandwidth usage in the cloud CDN. The VoD system exploits bandwidth in the cloud CDN by a combination of reservation and on-demand usage. Every t_0 time slots, the content provider calculates the average cloud bandwidth demand per unit time at each edge cloud as follows: In an ISP m with sufficient peer upload bandwidth, the average upload bandwidth to reserve in its edge cloud F_m is zero; in an ISP with insufficient peer upload bandwidth, the cloud bandwidth to reserve is $\max\{0, -\sum_{m=1}^{M} I(m)\} \frac{I(m)}{\sum_{t,I(t)<0} I(t)}$. Such reserved bandwidth is based on the average cloud bandwidth demand computed using our analytical models, to guarantee basic server bandwidth provisioning in the VoD system. Extra bandwidth demand arising over time in the dynamic streaming system can be flexibly served by on-demand bandwidth consumption from the respective edge clouds.

7 Performance evaluation

We implement our locality-aware P2P-cloud streaming protocol using an event-driven simulator. Three different peer selection strategies are evaluated, to compare their cloud bandwidth usage and inter-ISP traffic incurred: (1) ISP-aware downloading peer selection as proposed in Sec. 6; (2) An uploading peer selection strategy in which peers with the shortest service queue are selected as proposed in [Yang et al(2010)Yang, Chow, Golubchik, and Bragg]; (3) A simple locality-based strategy where with a fixed, large probability peers with the shortest service queue from the same ISP are selected as uploading peers and with a small probability peers with the shortest service queue from other ISPs are selected as uploading peers [158(2009)]. We also verify our analytical results by comparing them with the simulation results.

The simulation settings are as follows: $N = 10000$ peers are distributed among $M = 10$ ISPs. We model the peer distribution in different ISPs using the Zipf-Mandelbrot distribution function $p_m = \frac{(M-m+1)^\beta}{\sum_{m=1}^{M}(M-m+1)^\beta}$ as in [Dai et al(2011)Dai, Li, Liu, Li, and Jin], where the larger the absolute value of parameter β is, the more skewed the distribution is among different ISPs. The peer number inside ISP m is $N_m = p_m N$. There are 50 videos with equal size in the system. Each of these videos is divided into 100 chunks. The total number of chunks in the system is $J = 5000$. The stationary probability that a peer requests a chunk j can also be modeled by using the Zipf-Mandelbrot model [Dai et al(2011)Dai, Li, Liu, Li, and Jin], with $\pi_j = \frac{\frac{1}{(j+q)^\alpha}}{\sum_{j=1}^{J}\frac{1}{(j+q)^\alpha}}$, $\alpha = 0.78, q = 4$. Each peer can cache at most $B = 200$ chunks. At the beginning of each experiment, peers' caches are initialized by placing chunks in them according to the probability in Lemma 2. The LRU caching strategy is applied at the peers. The video rate is one chunk per second. The average upload bandwidth relative to the video rate in each ISP m is $U_m = 1 + \frac{\gamma-m}{10}, 1 \leq \gamma \leq 10$, where γ is a parameter. The startup delay of playing a video is 10 seconds. We use a daily periodicity user watching behavior for the number of active peers among one day [Huang et al(2008)Huang, Fu, Chiu, Lui, and Huang].When a peer is not online, the chunks stored in its cache are not refreshed, and they can be used to serve other peers when the peer rejoins the system.

7.1 Comparison Among Different Peer Selection Strategies

We first compare the cloud bandwidth usage and the inter-ISP traffic incurred using our proposed ISP-aware peer selection protocol in Sec. 6, the peer selection strategy of selecting peers with the shortest service queue, the simple locality-based strategy and our analytical results. In the simple locality-based strategy, peers select 80% of its neighbors in the same ISP, and uniformly randomly select the rest 20% from other ISPs. The inter-ISP traffic is measured as the average volume of traffic in one second, relative to the video playback rate. The cloud bandwidth usage is evaluated as the average proportion of bandwidth from the cloud CD-N in the overall chunk download bandwidth required. The analytical results are the computed required inter-ISP traffic and the minimum cloud bandwidth usage based on Theorem 2.

Fig. 3(a) and 3(b) show the inter-ISP traffic incurred and the cloud bandwidth usage under different average upload bandwidths in different ISPs, with peer distribution parameter $\beta = 0$, *i.e.*, peers are evenly distributed among different ISPs, and a proportion of 10% peers inactive. Fig. 4(a) and 4(b) show the inter-ISP traffic incurred and the cloud bandwidth usage under different peer distributions in different ISPs, with system-wide average peer upload bandwidth of 0.7, and a proportion of 10% peers inactive.

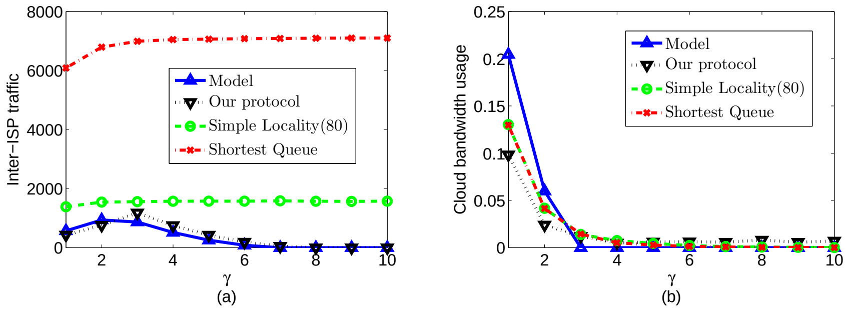

Fig. 3 (a) Inter-ISP traffic under different peer upload bandwidth distributions. (b) Cloud bandwidth usage under different peer upload bandwidth distributions.

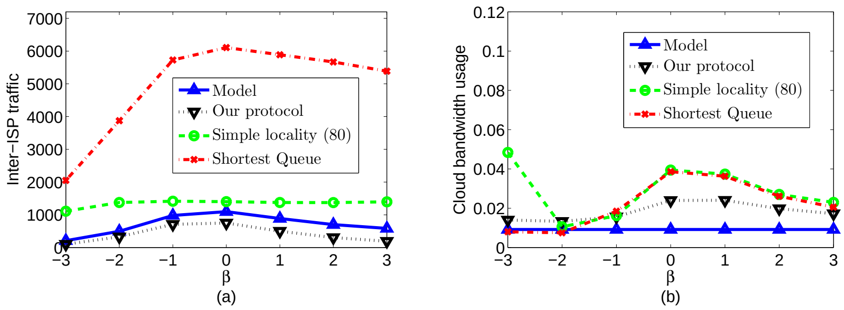

Fig. 4 (a) Inter-ISP traffic under different peer distributions. (b) Cloud bandwidth usage under different peer distributions.

In Fig. 3(a) and Fig. 4(a), we see that both our protocol and the simple locality-based strategy achieve much less inter-ISP traffic, as compared to the ISP-unaware (Shortest Queue) peer selection strategy. We also find that when peers' upload bandwidths are upgraded to a higher level, our protocol can reduce more inter-ISP traffic with little impact on the cloud bandwidth usage, while the volume of inter-ISP traffic under the random peer selection strategy and the simple locality-based strategy do not change much. In Fig. 3(b), when the average peer upload bandwidth is small, our analytical result indicates larger cloud bandwidth usage than the simulation results. This shows that a chunk is cached by its requested peers with a larger probability in the simulations, than the theoretical one. The gap could be due to the reason that the theoretical probability that a requested chunk is in the peer's cache is derived with the assumption that chunk requesting and chunk caching are independent (*i.e.*, the probability that the requested chunk c_j is in the peer's cache equals to the proportion of peers caching chunk c_j). Fig. 4(b) shows that when the numbers of peers are unbalanced in different ISPs and more peers are in ISPs with smaller bandwidth ($\beta = -3$), the cloud bandwidth usage is larger with the simple locality-based strategy. In all cases, our protocol achieves the cloud bandwidth consumption, close to that computed using our analytical results.

Fig. 5(a) and Fig. 5(b) further plot the cloud bandwidth usage and inter-ISP traffic in two day's time, under the three peer selection strategies, using $\gamma = 1$ and $\beta = 0$. Fig. 5(a) shows that the cloud bandwidth usage under our protocol is always the smallest among all three protocols. Fig. 5(b) shows that the inter-ISP traffic under our protocol is less than that under the other two strategies, which means unnecessary inter-ISP traffic is minimized by our protocol.

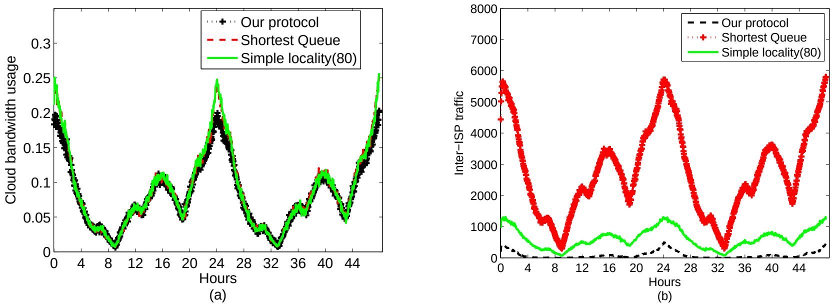

Fig. 5 (a) Cloud bandwidth usage in one day under different peer selection strategies. (b) Inter-ISP traffic in one day under different peer selection strategies.

7.2 Inter-ISP Traffic vs. Cloud Bandwidth Usage

In Fig. 6, we evaluate the relationship between the minimum cloud bandwidth usage and the maximally allowed overall inter-ISP traffic (T^c in Sec. 5.3) using our protocol, under different peer distribution and upload bandwidth levels in the system. We observe that in general, the cloud bandwidth usage is smaller when more inter-ISP traffic is allowed. The cloud bandwidth usage in the two cases of (1) $\beta = 0.5, \gamma = 3$ and (2) $\beta = 1, \gamma = 3$ is larger than that in the two cases of (3) $\beta = 0.5, \gamma = 6$ and (4) $\beta = 1, \gamma = 6$, since the upload bandwidth in the system is higher in the latter two cases. The cloud bandwidth consumption decreases to 0 as inter-ISP traffic increases above a specific volume, in systems of sufficient peer upload bandwidth.

8 Related Work

Hybrid P2P-cloud architectures have been proposed for content distribution in recent years, to exploit both the scalability and cost effectiveness of a P2P solution, and the reliability and performance guarantee delivered by a cloud infrastructure.

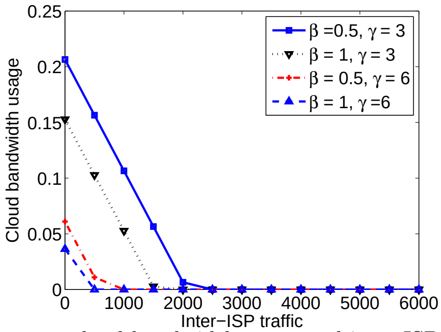

Fig. 6 Relationship between cloud bandwidth usage and inter-ISP traffic.

Wu *et al.* [Wu et al(2011)Wu, Wu, Li, Qiu, and Lau] analyze the equilibrium server capacity demand from the cloud under a client-server VoD model, and propose dynamic cloud storage and VM provision algorithms. Li *et al.* [Li et al(2012)Li, Zhang, Huang, Zhang, and Dai] propose a cloud bandwidth allocation algorithm among all peer swarms to maximize the aggregate downloading bandwidth at peers, over a small amount of cloud bandwidth. Trajkovska *et al.* [Trajkovska et al(2010)Trajkovska, Salvachua, and Velasco] present functions to calculate the quality of service in hybrid P2P-cloud streaming, for the providers to better monitor the dynamic QoS changes. Payberah *et al.* [Payberah et al(2012)Payberah, Kavalionak, Kumaresan, Montresor, and Haridi] propose to assist live streaming through a cloud, in order to guarantee the QoS of live streaming with the minimal cloud usage cost. Cervino *et al.* [Cervino et al(2011)Cervino, Rodriguez, Trajkovska, Velasco, and Salvachua] prove the benefits of deploying virtual machines in clouds to aid P2P streaming, by measuring the QoS improvement using testing experiments. These work do not address traffic localization in a hybrid P2P-cloud content distribution system. Our work aims to derive the optimal peer caching and optimal request distribution strategies in an ISP-aware P2P protocol, to achieve the minimal bandwidth consumption in the cloud CDN, and to incur the minimum inter-ISP traffic.

ISP-aware P2P protocol design is one of the solutions to resolve the tussles between P2P applications and ISPs. Several work have designed ISP-aware P2P protocols [Wang et al(2008)Wang, Huang, and Li] [Dai et al(2011)Dai, Li, Liu, Li, and Jin]. Wang *et al.* [Wang et al(2008)Wang, Huang, and Li] formulate the optimal ISP-aware rate allocation problem in peer-assisted VoD streaming as a 3-stage optimization problem, to minimize server load, to minimize ISP-unfriendly traffic, and to maximize peer prefetching rates, respectively. The optimization formulation allows different tradeoffs between the server bandwidth and the ISP-unfriendly traffic. Dai *et al.* [Dai et al(2011)Dai, Li, Liu, Li, and Jin] study collaborative ISP caching strategies for the reduction of inter-ISP traffic. The resource allocation mechanism, *e.g.*, the allocation of storage and upload bandwidth of the cache servers, is designed with awareness of inter-ISP traffic and ISP policies. However, none of these work quantifies the tradeoff among traffic locality and server capacity usage. In this paper, we study the fundamental relationship between the server bandwidth and inter-ISP traffic in a hybrid P2P-cloud system, which was not addressed by any of the previous work.

The caching strategies in VoD streaming have significant impact on the streaming performance, which have been studied in the literature [Zhou et al(2011)Zhou, Fu, and Chiu] [Wu and Lui(2011)] [Wu and Li(2009)] [Tan and Massoulié(2011)]. Zhou *et al.* [Zhou et al(2011)Zhou, Fu, and Chiu] assume that a peer downloading a movie issues a downloading request to all peers caching the movie. A peer evenly divides its upload bandwidth to serve all the outstanding requests. A random load balancing algorithm is proposed. The upper and lower bounds of the required server capacity is derived under random downloading request arrivals. Wu *et al.* [Wu and Lui(2011)] analyze the optimal replication ratio in a multi-video P2P VoD system. Peers arriving earlier can upload contents to later peers watching the same video. The deficit upload bandwidth that can not be supplied by peers watching the same video is supplemented by peers who have cached the video earlier. They derive that the optimal cache replication ratio is proportional to the deficit bandwidth. Wu *et al.* [Wu and Li(2009)] use optimal control theory and dynamic programming to construct the optimal cache replacement strategy. Their simulations show that the simplest algorithms such as LRU have a close-to-optimal performance compared to the optimal strategy. In this paper, we follow the loss network model in [Tan and Massoulié(2011)], but differences are apparent: (i) we use it to model P2P VoD streaming, while [Tan and Massoulié(2011)] applies the loss network to model a distributed server network; (ii) we propose a general optimal caching condition based on the loss network model, while [Tan and Massoulié(2011)] has proven the optimality of a proportional-to-product placement strategy.

9 Conclusions

This paper targets an in-depth theoretical study of the relationship between inter-ISP traffic and cloud bandwidth usage in hybrid P2P-Cloud VoD streaming systems, as well as practical locality-aware, hybrid P2P-cloud streaming protocol design based on the theoretical insights. We propose optimal peer caching and request distribution conditions for achieving the minimum cloud bandwidth usage in the cloud CDN. We analyze the minimum volume of inter-ISP traffic required to achieve the minimum cloud bandwidth, as well as how different levels of restricted inter-ISP traffic influence the cloud bandwidth usage. We show that when the allowed inter-ISP traffic is restricted below the minimum volume to achieve the minimum cloud bandwidth, the cloud bandwidth usage increases linearly with the decrease of the allowed inter-ISP traffic. We also design a practical, locality-aware P2P streaming protocol, to achieve the minimum cloud bandwidth usage and the minimum inter-ISP traffic in a hybrid P2P-cloud system. Simulations under realistic settings verify the performance of our protocol.

A Proof of Theorem 1

Proof: Denote the graph as $G = (V, E)$. We first construct an S-T cut of the graph $C_{st} = (\mathcal{F}, \mathcal{L})$, a partition of nodes V with source node $\mathcal{S} \in \mathcal{F}$ and destination node $\mathcal{T} \in \mathcal{L}$. The cut set of C_{st} is the set $\{(u, v) \in E | u \in \mathcal{F}, v \in \mathcal{L}\}$. We construct a cut set as follows: We can divide $x_{m,j}$'s, $1 \leq j \leq J$, into two classes according to whether $x_{m,j}$ is equal to $\sum_{l=1}^{M} a_{lm} r_{l,j}$ or smaller than $\sum_{l=1}^{M} a_{lm} r_{l,j}$: $\mathcal{C}_1 = \{c_j | x_{m,j} = \sum_{l=1}^{M} a_{lm} r_{l,j}, 1 \leq j \leq J\}$, and

$\mathcal{C}_2 = \{c_j | x_{m,j} < \sum_{l=1}^{M} a_{lm} r_{l,j}, 1 \leq j \leq J\}$. There is not enough peer upload bandwidth in ISP m to serve chunks in set $\mathcal{C}_2$. From the KKT conditions, we can see that for all $\mathcal{A}$ which includes $c_j \in \mathcal{C}_1$, we have $\epsilon_{\mathcal{A}} = 0$ according to KKT condition (7); for $\mathcal{A} = \mathcal{C}_2$, we have $\sum_{c_j \in \mathcal{C}_2} x_{m,j} = U_m \cdot \sum_{i: s_i \cap \mathcal{C}_2 \neq \emptyset} N_m^{(i)}$ according to KKT condition (6). According to KKT condition (6), we have $\epsilon_{\mathcal{C}_2} > 0$. The nodes are partitioned into two sets. The set including source node is $\mathcal{F} = \{$source node $\mathcal{S}$, all nodes in $\mathcal{C}_2$, and nodes in Θ that are connected to nodes in $\mathcal{C}_2\}$, the set including destination node is $\mathcal{L} = \{$all other nodes in the graph not in $\mathcal{F}$ $\}$.

Now let us see the capacity of the constructed cut set. The cut set includes all the edges connecting $\mathcal{S}$ with all nodes in $\mathcal{C}_1$, and all the edges connecting $\mathcal{T}$ with the nodes in Θ which are connected to nodes in $\mathcal{C}_2$. The capacity of all the edges connecting $\mathcal{S}$ with all nodes in $\mathcal{C}_1$ is $\sum_{c_j \in \mathcal{C}_1} \sum_{l=1}^{M} a_{lm} r_{l,j}$, the capacity of all the edges connecting $\mathcal{T}$ with the nodes in Θ which are connected to nodes in $\mathcal{C}_2$ is $U_m \cdot \sum_{i: s_i \cap \mathcal{C}_2 \neq \emptyset} N_m^{(i)}$. As we have $U_m \cdot \sum_{i: s_i \cap \mathcal{C}_2 \neq \emptyset} N_m^{(i)} = \sum_{c_j \in \mathcal{C}_2} x_{m,j}$, $x_{m,j} = \sum_{l=1}^{M} a_{lm} r_{l,j}, \forall c_j \in \mathcal{C}_1$, the total capacity of the cut set is equal to $\sum_{j=1}^{J} x_{m,j}$.

The capacity of this cut set equals to the number of total concurrently served requests obtained by solving the KKT conditions. We prove that this cut is the minimum cut in the graph. Applying the min-cut max-flow theorem, we can then show that the number of total concurrently served requests is the maximum flow.

On one hand, if $\mathcal{F}$, the set including $\mathcal{S}$, contains a node c_j from $\mathcal{C}_1$, since the overall amount of flow from c_j to nodes in Θ is equal to the capacity of the edge from $\mathcal{S}$ to c_j, the capacity of the new cut set is no smaller than that of the original cut set with $\mathcal{F}$ excluding nodes from $\mathcal{C}_1$. When all nodes in Θ connecting to c_j are included together with c_j in the set $\mathcal{F}$, the capacity of the new cut is increased by $U_m \cdot \sum_{s_i : c_j \in s_i} N_m^{(i)} - \sum_{l=1}^{M} a_{lm} r_{l,j} \geq 0, c_j \in \mathcal{C}_1$, according to KKT condition (4).

On the other hand, if the set including source node $\mathcal{S}$ excludes a node c_k in $\mathcal{C}_2$, the edge ($\mathcal{F}$, c_k) will be added into the new cut set, the capacity of the new cut set is increased by $\sum_{l=1}^{M} a_{lm} r_{l,j}$. If a node s_i in Θ connecting to c_k is excluded with c_k together from the set including source node $\mathcal{S}$, the edges ($\mathcal{F}$, c_j) with $c_j \in s_i$ will be added into the new cut set, the edge (s_i, $\mathcal{L}$) will be deleted from the new cut set. Hence, the capacity of new cut set will be increased by $\sum_{j: c_j \in s_i} \sum_{l=1}^{M} a_{lm} r_{l,j} - U_m N_m^{(i)}$, which is larger than 0 since chunks in $\mathcal{C}_2$ do not receive enough peer upload bandwidth to be served. Hence, the capacity of new cut set increases.

Therefore, the cut we construct with capacity $\sum_{j=1}^{J} x_{m,j}$ is the minimum cut. Applying the min-cut max-flow theorem, we have proved that the number of total concurrently served requests is equal to the maximum bipartite flow. □

B Proof of Lemma 1

Proof: When $N_m^{(i)}$'s satisfy (9), let $\epsilon_{\mathcal{A}} = 0$ for all $\mathcal{A} \subset \mathcal{C}$, and $\epsilon_{\mathcal{A}} = \max\{0, \ln \eta_m\}$ for $\mathcal{A} = \mathcal{C}$. It is easy to verify that $x_{m,j} = \min\{\sum_{l=1}^{M} a_{lm} r_{l,j}, \frac{\sum_{l=1}^{M} a_{lm} r_{l,j}}{\eta_m}\}, 1 \leq j \leq J$, are solutions to the KKT conditions in Sec. 3.2. Hence, the average chunk miss rate is $L_m = 1 - \frac{\sum_{j=1}^{J} x_{m,j}}{\sum_{l=1}^{M} a_{lm} r_l} = 1 - \frac{\min\{1, \frac{1}{\eta_m}\} \sum_{j=1}^{J} \sum_{l=1}^{M} a_{lm} r_{l,j}}{\sum_{l=1}^{M} a_{lm} r_l} = 1 - \min\{1, \frac{1}{\eta_m}\} = \max\{0, 1 - \frac{1}{\eta_m}\}$. □

C Proof of Lemma 2

Proof: We prove the caching strategy in Lemma 2 satisfies the optimal cache condition (9) in Lemma 1, by induction.

(i) When $|\mathcal{A}| > J - B$, the intersection between any cache state $s_i, 1 \leq i \leq W$, and $\mathcal{A}$ is not empty. Hence, the L.H.S of (9) equals to $\sum_{c_j \in \mathcal{A}} \sum_{l=1}^{M} a_{lm} r_{l,j}$, which is smaller than the

R.H.S of (9), $\eta_m \cdot U_m \cdot N_m = \sum_{l=1}^{M} a_{lm} r_l$.
(ii) Let $\mathcal{A}_k$ denote the set $|\mathcal{A}| = k$. When $|\mathcal{A}| = J - B$, we use $\mathcal{A}_{J-B}$ to denote $\mathcal{A}$. Let $\bar{\mathcal{A}}_{J-B}$ be the complementary set of $\mathcal{A}_{J-B}$. $|\bar{\mathcal{A}}_{J-B}| = B$. The intersection between the cache state $s_i = \bar{\mathcal{A}}_{J-B}$ and $\mathcal{A}_{J-B}$ is empty.

$$\text{R.H.S of (9)} = \eta_m \cdot U_m (N_m - \sum_{i:s_i = \bar{\mathcal{A}}_{J-B}} N_m^{(i)})$$
$$= \eta_m \cdot U_m (N_m - N_m \cdot \prod_{c_j \in \bar{\mathcal{A}}_{J-B}} \rho_j)$$
$$\geq \eta_m \cdot U_m (N_m - N_m \cdot (\frac{B}{\sum_{l=1}^{M} a_{lm} r_l})^B \prod_{c_j \in \bar{\mathcal{A}}_{J-B}} \sum_{l=1}^{M} a_{lm} r_{l,j})$$
$$\geq \eta_m \cdot U_m (N_m - N_m \cdot (\frac{B}{\sum_{l=1}^{M} a_{lm} r_l})^B \cdot (\frac{\sum_{c_j \in \bar{\mathcal{A}}_{J-B}} \sum_{l=1}^{M} a_{lm} r_{l,j}}{B})^B)$$
$$\geq \sum_{l=1}^{M} a_{lm} r_l - (\sum_{c_j \in \bar{\mathcal{A}}_{J-B}} \sum_{l=1}^{M} a_{lm} r_{l,j}) \cdot (\frac{\sum_{c_j \in \bar{\mathcal{A}}_{J-B}} \sum_{l=1}^{M} a_{lm} r_{l,j}}{\sum_{l=1}^{M} a_{lm} r_l})^{B-1}$$
$$\geq \sum_{c_j \in \mathcal{A}_{J-B}} \sum_{l=1}^{M} a_{lm} r_{l,j} = \text{L.H.S of (9)}$$

(iii) For $k \leq J - B$, suppose all $\mathcal{A}_k$'s satisfy (9). Consider the case for any $\mathcal{A}_{k-1}$. Let us consider a specific $\mathcal{A}_{k-1}$, there are $(J - k + 1)$ chunks $c_j \notin \mathcal{A}_{k-1}$. For any $c_j \notin \mathcal{A}_{k-1}$, we can construct an $\mathcal{A}_k = c_j \cup \mathcal{A}_{k-1}$. Hence, we get $(J-k+1)$ sets of $\mathcal{A}_k$'s. Let $\mathcal{A}_k^1, \mathcal{A}_k^2, \ldots, \mathcal{A}_k^{J-k+1}$ denote them. For each $\mathcal{A}_k$, we apply (9),

$$\sum_{c_j \in \mathcal{A}_k^t} \sum_{l=1}^{M} a_{lm} r_{l,j} \leq \eta_m \cdot U_m \cdot \sum_{i:\mathcal{A}_k^t \cap s_i \neq \emptyset} N_m^{(i)}, 1 \leq t \leq J - k + 1.$$

Sum up all the $(J - k + 1)$ inequalities, we have,

$$\sum_{t=1}^{J-k+1} \sum_{c_j \in \mathcal{A}_k^t} \sum_{l=1}^{M} a_{lm} r_{l,j}$$
$$= (J - k + 1) \sum_{c_j \in \mathcal{A}_{k-1}} \sum_{l=1}^{M} a_{lm} r_{l,j} + \sum_{c_j \notin \mathcal{A}_{k-1}} \sum_{l=1}^{M} a_{lm} r_{l,j}$$
$$\leq \sum_{t=1}^{J-k+1} \eta_m \cdot U_m \cdot \sum_{i:\mathcal{A}_k^t \cap s_i \neq \emptyset} N_m^{(i)}$$
$$= (J - k + 1)\eta_m \cdot U_m \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)}$$
$$+ \eta_m \cdot U_m \sum_{i:s_i \cap \mathcal{A}_{k-1} = \emptyset} N_m^{(i)}.$$

Hence,

$$(J - k + 1) \sum_{c_j \in \mathcal{A}_{k-1}} \sum_{l=1}^{M} a_{lm} r_{l,j} + \sum_{l=1}^{M} a_{lm} r_l - \sum_{c_j \in \mathcal{A}_{k-1}} \sum_{l=1}^{M} a_{lm} r_{l,j}$$
$$\leq (J - k + 1)\eta_m \cdot U_m \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)}$$
$$+ \eta_m \cdot U_m (N_m - \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)}).$$

We have

$$\sum_{c_j \in \mathcal{A}_{k-1}} \sum_{l=1}^{M} a_{lm} r_{l,j} \leq \eta_m \cdot U_m \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)}.$$

Hence, Lemma 2 is proven. □

D Proof of Lemma 3

Proof: Suppose each peer's cache contains B ordered positions, to cache B chunks. Consider chunk j's position in a peer's cache n time units after the peer starts playing videos, under the LRU algorithm, which is denoted by s_n^j. We first derive the position transition probabilities, and then the stationary-state probability of a peer caching chunk c_j.

Recall that a peer has a probability $\sum_{j=1}^{J} \pi_j \rho_j$ to replay a cached chunk. The probability of downloading and playing a new chunk is $1 - \pi_0 - \sum_{j=1}^{J} \pi_j \rho_j$. We assume a chunk being played (no matter newly downloaded or already cached) is put into position 1 in the cache. In case of a new download, the chunk in the last position of the cache, position B, will be evicted, and positions of all other cached chunks will be increased by 1 (LRU algorithm). In case of replaying a cached chunk, positions of chunks cached before this one, will be increased by 1. Given s_n^j, we can derive the probability distribution of chunk j's position at time $n+1$, *i.e.*, s_{n+1}^j. For $2 \leq b \leq B$,

$$\begin{aligned} & Pr[s_{n+1}^j = b | s_n^j] = \\ & Pr[c_j\text{'s position increases by } 1 | s_n^j = b-1] \cdot Pr[s_n^j = b-1] \\ & \quad + Pr[c_j\text{'s position does not change} | s_n^j = b] \cdot Pr[s_n^j = b]. \end{aligned}$$

The event that chunk j's position increases by 1 when $s_n^j = b - 1$ can be divided into two disjoint cases: (i) when the peer plays a new chunk, and (ii) when the peer replays a chunk cached at a position behind $b - 1$:

$$\begin{aligned} & Pr[c_j\text{'s position increases by } 1 | s_n^j = b-1] = \\ & (1 - \pi_0 - \sum_{i \neq j} \pi_i \rho_i) + \sum_{i \neq j} \pi_i \cdot Pr[s_n^i > b-1]. \end{aligned}$$

The event that chunk j's position does not change when $s_n^j = b$ happens when the peer replays a chunk cached at a position ahead of b:

$$Pr[c_j\text{'s position does not change} | s_n^j = b] = \pi_0 + \sum_{i \neq j} \pi_i Pr[s_n^i < b].$$

Hence,

$$\begin{aligned} Pr[s_{n+1}^j = b | s_n^j] = & \Big\{ (1 - \pi_0 - \textstyle\sum_{i \neq j} \pi_i \rho_i) \\ & + \textstyle\sum_{i \neq j} \pi_i \cdot Pr[s_n^i > b-1] \Big\} \cdot Pr[s_n^j = b-1] \\ & + \{ \pi_0 + \textstyle\sum_{i \neq j} \pi_i Pr[s_n^i < b] \} \cdot Pr[s_n^j = b]. \end{aligned} \tag{16}$$

Equation (16) shows that the next position of chunk j is only related to the previous position of chunk j. We can use a Markov chain to model the change of chunk j's positions: state $b, 1 \leq b \leq B$, represents that chunk j is at position b in the peer's cache; state 0 denotes that chunk j is not in the peer's cache; $Pr[s_{n+1}^j = b | s_n^j = b-1]$ is the transition probability from state $b - 1$ to state b for chunk j. Using this Markov chain model, we can analyze the stationary-state distribution of chunk j's position in a peer's cache.

Let s^j denote the stationary-state position of chunk j when n goes to infinity. Based on equation (16), for $2 \leq b \leq B$, we have,

$$\begin{aligned} & Pr[s^j = b] = (1 - \pi_0 - \textstyle\sum_{i \neq j} \pi_i \rho_i + \sum_{i \neq j} \pi_i \sum_{p=b}^{B} Pr[s_n^i = p]) \\ & \cdot Pr[s^j = b-1] + (\pi_0 + \textstyle\sum_{i \neq j} \pi_i \sum_{p=1}^{b-1} Pr[s_n^i = p]) \cdot Pr[s^j = b]. \end{aligned}$$

As $\pi_i \rho_i = \pi_i \sum_{p=1}^{B} Pr[s_n^i = p]$, hence,

$$Pr[s^j = 1] = Pr[s^j = 2] = \ldots = Pr[s^j = B]. \tag{17}$$

This implies that a chunk is cached in different positions with an equal probability in the steady state.

For $b = 1$, $Pr[s^j = 1]$ is the stationary probability that chunk j is cached at position 1 in the peer's cache, which equals to the stationary probability that a peer in the VoD system is requesting and playing chunk j that we have defined earlier, *i.e.*, π_j.

Therefore, under the LRU algorithm, the probability that a peer caches chunk j is $B \cdot Pr[s^j = 1] = B \cdot \pi_j$, the proportion of peers caching chunk j is $\rho_j = B \cdot \pi_j$ as well. This is the optimal cache distribution proposed in Lemma 2. □

E Proof of Theorem 2

Proof: Consider two cases:

1) The overall upload bandwidth of peers in the system is no smaller than the bandwidth demand for serving all the chunk requests, *i.e.*, $\sum_{m=1}^{M} U_m N_m (1-\pi_0) \geq \sum_{l=1}^{M} r_l$. In this case, $\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} \geq 1$. Hence, for ISPs with $I_m > 0$,

$$a_{lm}^* = \begin{cases} 1: & l = m, \\ 0: & l \neq m, I_l \geq 0, \\ \frac{I_m}{\sum_{t,I_t>0} I_t}(\frac{-I_l}{r_l}): & l \neq m, I_l \leq 0. \end{cases}$$

for ISPs with $I_m < 0$,

$$a_{lm}^* = \begin{cases} \frac{U_m N_m (1-\pi_0)}{r_m}: & l = m; \\ 0: & l \neq m, 1 \leq l \leq M. \end{cases}$$

First, we can verify that a_{ml}^*'s satisfy constraints (**??**): for ISPs with $I_m < 0$, $\sum_{l=1}^{M} a_{lm}^* r_l = a_{mm}^* r_m = U_m N_m (1-\pi_0)$; for ISPs with $I_m \geq 0$, $\sum_{l=1}^{M} a_{lm}^* r_l = r_m + \frac{-\sum_{l,I_l<0} I_l}{\sum_{t,I_t>0} I_t} I_m \leq U_m N_m (1-\pi_0)$. Hence, in any ISP, the total number of requests is fewer than or equal to the available peer upload bandwidth. $L_m^* = 0$ for $1 \leq m \leq M$, which means a_{ml}^*'s achieve the minimum system-wide chunk miss rate.

Second, we show that $T^* = \sum_{m=1}^{M} \sum_{l \neq m} a_{lm}^* \cdot r_l \cdot (1 - L_m^*) = \sum_{m=1}^{M} \sum_{l \neq m} a_{lm}^* \cdot r_l = \sum_{m=1}^{M} (1 - a_{mm}^*) \cdot r_m = \sum_{m,I_m<0} [r_m - U_m N_m (1-\pi_0)]$ is the minimum inter-ISP traffic to achieve the minimum chunk miss rate, by contradiction. Suppose there are a set of a_{lm}''s, different from a_{lm}^*'s, satisfying the constraints and inducing the volume of inter-ISP traffic $T' < T^*$, and still achieving the minimum chunk miss rate, $L_m' = L_m^* = 0$ for $1 \leq m \leq M$. $T' = \sum_{m=1}^{M} \sum_{l \neq m} a_{lm}' \cdot r_l \cdot (1 - L_l') = \sum_{m=1}^{M} (1 - a_{mm}') \cdot r_m < T^*$. Hence, there exists $a_{mm}' > a_{mm}^*$. If $I_m \geq 0$, then $a_{mm}' > a_{mm}^* = 1$, this contradicts with $\sum_{l=1}^{M} a_{lm}' = 1$; if $I_m < 0$, then $a_{mm}' > a_{mm}^* = \frac{N_m U_m}{r_m}$, $\sum_{l=1}^{M} a_{lm}' r_m \geq a_{mm}' \cdot r_m > N_m U_m$, this contradicts with $L_m' = 0$. Hence, such a_{lm}''s do not exist. a_{lm}^*'s are the optimal solutions. $T^* = \sum_{m,I_m<0} (r_m - U_m N_m)$ is the minimum volume of inter-ISP traffic.

2) The overall upload bandwidth of peers in the system is smaller than the bandwidth demand for serving all the chunk requests, *i.e.*, $\sum_{m=1}^{M} U_m N_m (1-\pi_0) < \sum_{l=1}^{M} r_l$. In this case, $\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} < 1$. Hence, for ISPs with $I_m > 0$,

$$a_{lm}^* = \begin{cases} 1: & l = m, \\ 0: & l \neq m, I_l \geq 0, \\ \frac{I_m}{-\sum_{-s,I_s<0} I_s}(\frac{-I_l}{r_l}): & l \neq m, I_l \leq 0. \end{cases}$$

for ISPs with $I_m < 0$,

$$a_{lm}^* = \begin{cases} (1 - \frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s})(\frac{-I_m}{r_m}) + \frac{U_m N_m (1-\pi_0)}{r_m}, \\ \quad : \quad l = m; \\ 0: \quad l \neq m, 1 \leq l \leq M. \end{cases}$$

First, for ISPs with $I_m < 0$, $\sum_{l=1}^{M} a^*_{lm} r_l = r_m + I_m \cdot \frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} = I_m(\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} - 1) + U_m N_m(1-\pi_0) > U_m N_m(1-\pi_0)$. For ISPs with $I_m \geq 0$, $\sum_{l=1}^{M} a^*_{lm} r_l = r_m + \sum_{l,I_l<0} a^*_{lm} r_l = r_m + I_m = U_m N_m(1-\pi_0)$. Hence, a^*_{lm}'s satisfy condition (14). For ISPs with $I_m < 0$, $L^*_m > 0$; for ISPs with $I_m \geq 0$, $L^*_m = 0$.

Second, we show $T^* = \sum_{m=1}^{M} \sum_{l \neq m} a^*_{lm} \cdot r_l \cdot (1 - L^*_m) = \sum_{m,I_m \geq 0} \sum_{l \neq m} a^*_{lm} r_l = \sum_{m,I_m \geq 0} \sum_{l,I_l<0} \frac{I_m}{-\sum_{s,I_s<0} I_s}(-I_l) = \sum_{m,I_m>0} I_m$ is the minimum inter-ISP traffic. Suppose there are a set of a'_{lm}'s, different from a^*_{lm}'s, satisfying the optimal chunk request routing constraints in (14). Hence, for ISPs with $I_m \geq 0$, $\sum_{l=1}^{M} a'_{lm} r_l \geq \sum_{l=1}^{M} a^*_{lm} r_l = U_m N_m(1-\pi_0)$. We have $T' = \sum_{m=1}^{M} \sum_{l \neq m} a'_{lm} \cdot r_l \cdot (1 - L_m) = \sum_{m=1}^{M} (\sum_{l=1}^{M} a'_{lm} r_l - a'_{mm} r_m) \cdot \frac{U_m N_m(1-\pi_0)}{\sum_{l=1}^{M} a'_{lm} \cdot r_l} \geq \sum_{m,I_m>0}$
$(\sum_{l=1}^{M} a'_{lm} r_l - a'_{mm} r_m) \cdot \frac{U_m N_m(1-\pi_0)}{\sum_{l=1}^{M} a'_{lm} \cdot r_l} \geq \sum_{m,I_m>0} [U_m N_m(1-\pi_0) - a'_{mm} r_m \frac{U_m N_m(1-\pi_0)}{\sum_{l=1}^{M} a'_{lm} \cdot r_l}] \geq \sum_{m,I_m>0} [U_m N_m(1-\pi_0) - r_m] = \sum_{m,I_m>0} I_m = T^*$, as $a'_{mm} \leq 1$ and $\sum_{l=1}^{M} a'_{lm} r_l \geq U_m N_m(1-\pi_0)$. So, T^* is the minimum inter-ISP traffic. □

References

[158(2009)] (2009) A Case Study of Traffic Locality in Internet P2P Live Streaming Systems

[Clo(http://aws.amazon.com/cloudfront/)] (http://aws.amazon.com/cloudfront/) Amazon CloudFront

[Clo(http://aws.amazon.com/cloudfront/#pricing)] (http://aws.amazon.com/cloudfront/#pricing) CloudFront Pricing

[S3P(http://aws.amazon.com/s3/pricing/)] (http://aws.amazon.com/s3/pricing/) Amazon S3 Pricing

[Soh(http://tv.sohu.com)] (http://tv.sohu.com) Sohu Video

[PPL(http://www.pplive.tv)] (http://www.pplive.tv) PPLive

[Cervino et al(2011)Cervino, Rodriguez, Trajkovska, Velasco, and Salvachua] Cervino J, Rodriguez P, Trajkovska I, Velasco AM, Salvachua J (2011) Testing a Cloud Provider Network for Hybrid P2P and Cloud Streaming Architectures. In: Proc. of CLOUD

[Dai et al(2011)Dai, Li, Liu, Li, and Jin] Dai J, Li B, Liu FM, Li BC, Jin H (2011) On the Efficiency of Collaborative Caching in ISP-aware P2P Networks. In: Proc. of IEEE INFOCOM

[Feng et al(2010)Feng, Li, and Liam] Feng Y, Li BC, Liam B (2010) Peer-assisted VoD Prefetching in Double Auction Markets. In: Proc. of ICNP

[Heyman and Sobel(2004)] Heyman DP, Sobel MJ (2004) Stochastic Models in Operations Research: Stochastic Processes and Operating Characteristics. Courier Dover

[Huang et al(2008)Huang, Fu, Chiu, Lui, and Huang] Huang Y, Fu TZJ, Chiu DM, Lui JCS, Huang C (2008) Challenges, Design and Analysis of a Large-Scale P2P-VoD System. In: Proc. of SIGCOMM

[Inc.(2010)] Inc CS (2010) Cisco Visual Networking Index: Forecast and Methodology, 2009-2014. White paper

[Jung et al(2008)Jung, Lu, Shah, Sharma, and Squillante] Jung K, Lu YD, Shah D, Sharma M, Squillante MS (2008) Revisiting Stochastic Loss Networks: Structures and Algorithms. In: Proc. of SIGMETRICS

[Kelly(1991)] Kelly F (1991) Loss networks. The Annals of Applied Probability 1(3):319–378

[Li et al(2012)Li, Zhang, Huang, Zhang, and Dai] Li Z, Zhang T, Huang Y, Zhang ZL, Dai Y (2012) Maximizing the Bandwidth Multiplier Effect for Hybrid Cloud-P2P Content Distribution. In: Proc. of IWQoS

[Magharei et al(2009)Magharei, Rejaie, Hilt, Rimac, and Hofmann] Magharei N, Rejaie R, Hilt V, Rimac I, Hofmann M (2009) ISP-Friendly Live P2P Streaming. In: Poster. of ACM SIGCOMM

[Negruseri et al(2009)Negruseri, Pasoi, Stanley, and Stein] Negruseri C, Pasoi M, Stanley B, Stein C (2009) Solving Maximum Flow Problems on Real World Bipartite Graphs. In: Proc. of 11th Workshop on Algo. Engineering and Experiments, pp 14–28

[Payberah et al(2012)Payberah, Kavalionak, Kumaresan, Montresor, and Haridi] Payberah AH, Kavalionak H, Kumaresan V, Montresor A, Haridi S (2012) CLIVE: Cloud-Assisted P2P Live Streaming. In: Proc. of P2P Computing

[Picconi and Massoulie(2009)] Picconi F, Massoulie L (2009) ISP Friend or Foe? Making P2P Live Streaming ISP-Aware. In: Proc. of IEEE ICDCS

[Tan and Massoulié(2011)] Tan BR, Massoulié L (2011) Optimal Content Placement for Peer-to-Peer Video-on-Demand Systems. In: Proc. of IEEE INFOCOM

[Trajkovska et al(2010)Trajkovska, Salvachua, and Velasco] Trajkovska I, Salvachua J, Velasco AM (2010) A Novel P2P and Cloud Computing Hybrid Architecture for Multimedia Streaming with QoS Cost Functions. In: Proc. of MM

[Wang et al(2008)Wang, Huang, and Li] Wang J, Huang C, Li J (2008) On ISP-friendly Rate Allocation for Peer-assisted VoD. In: Proc. of ACM Multimedia 2008, DOI http://doi.acm.org/10.1145/1459359.1459397

[Wu and Li(2009)] Wu JH, Li BC (2009) Keep Cache Replacement Simple in Peer-Assisted VoD Systems. In: Proc. of INFOCOM

[Wu and Lui(2011)] Wu WJ, Lui JCS (2011) Exploring the Optimal Replication Strategy in P2P-VoD Systems: Characterization and Evaluation. In: Proc. of IEEE INFOCOM

[Wu et al(2011)Wu, Wu, Li, Qiu, and Lau] Wu Y, Wu C, Li B, Qiu X, Lau FC (2011) CloudMedia: When Cloud on Demand Meets Video on Demand. In: Proc. of ICDCS

[Xie et al(2008)Xie, Yang, Krishnamurthy, Liu, and Silberschatz] Xie H, Yang Y, Krishnamurthy A, Liu Y, Silberschatz A (2008) P4P: Provider Portal for Applications. In: Proc. of ACM SIGCOMM

[Yang et al(2010)Yang, Chow, Golubchik, and Bragg] Yang Y, Chow AL, Golubchik L, Bragg D (2010) Improving QoS in BitTorrent-like VoD Systems. In: Proc. of IEEE INFOCOM 2010

[Zhou et al(2011)Zhou, Fu, and Chiu] Zhou Y, Fu TZ, Chiu D (2011) Statistical Modeling and Analysis of P2P Replication to Support VoD Service. In: Proc. of IEEE INFOCOM