Stochastic Model for ISP-aware VoD Streaming

Jian Zhao, Chuan Wu
Department of Computer Science
The University of Hong Kong, Hong Kong
jzhao, cwu@cs.hku.hk

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

ABSTRACT

ISP-aware P2P applications are proposed to reduce the significant amount of costly cross-ISP traffic. The impact of ISP-awareness on P2P system performance is not well understood. There is still lack of theoretical study on the relationship between performance and the volume of cross-ISP traffic. In this paper, we focus on ISP-aware P2P Video-on-Demand streaming systems. We use the chunk request loss rate as the performance metrics and apply the loss network model to analyze the chunk request loss rateWe propose the optimal peer cache condition and optimal chunk request routing condition for achieving the minimum system chunk request loss rate. We analyze what is the minimum cross-ISP traffic to achieve the minimum system chunk request loss rate and how the limited cross-ISP traffic affects the minimum system chunk request loss rate. We also perform simulations to validate our theoretical study. Our theoretical study not only help us understand the impact of ISP-awareness on the P2P VoD system performance, but also offer fundamental insights to designing ISP-aware P2P VoD systems.

1. INTRODUCTION

Online videos are popular due to the proliferation of high-speed broadband services. The video traffic through the Internet and the server workloads increase largely [1]. As peer-to-peer technology is applied in video streaming systems, including Video-on-Demand systems (*e.g.*, PPLive [2], UUSee [3]), to alleviate the heavy workload of servers in data centers, the ISP-agnostic P2P connections bring about large volume of unnecessary inter-ISP traffic, which increases the cost of ISPs. This makes ISPs start to proactively detect and throttle P2P data packets, which definitely affects the streaming service quality.

The tussle between content distributors and ISPs attracts the attention of researchers. To solve the tussle, ISP-aware P2P applications are proposed. P4P [4] achieves ISP-friendly traffic control based on an architecture providing interfaces for networks to communicate with P2P applications. Huang *et al.* [5] design distributed peer selection algorithms that can effectively achieve desired performance and traffic locality tradeoff through multi-objective optimization. Fabio Picconi *et al.* [6] proposes a two-level overlay and a dynamic unchoke algorithm to reduce unnecessary inter-ISP traffic in P2P live streaming applications. Wang *et al.* [7] design an ISP-friendly rate allocation algorithm for peer-assisted VoD systems.

The common feature of the above ISP-aware P2P applications is that peers adaptively adjust the cross-ISP connections to reduce the unnecessary cross-ISP traffic and guarantee the system performance simultaneously. The remaining problem is how the limited cross-ISP traffic will affect the system performance. There is still lack of theoretical study on the impact of controlled peer selection on P2P system performance. This paper focuses on theoretical studying of ISP-aware peer-to-peer Video-on-Demand systems based on the loss network model. Our contributions are: (1) we derive the optimal cache condition and the optimal chunk request routing condition for the system to achieve the minimum chunk request loss rate; (2) we analyze the necessary minimum cross-ISP traffic to achieve the minimum chunk request loss rate and analyze how the limited cross-ISP traffic will affect the chunk request loss rate.

The remainder of the paper is organized as follows. We present our system model and notations in Sec. 2 We apply the stochastic loss network analysis and map solutions of the optimization problem derived from the loss network analysis into the corresponding maximum bipartite flow in Sec. 3. We state the optimal peer cache condition and the optimal chunk request routing condition in Sec. 4. We analyze the relationship between cross-ISP traffic and chunk request loss rate in Sec. 5. We perform performance evaluation in Sec. 6, and conclude the paper in Sec. 7.

2. SYSTEM MODEL AND NOTATIONS

We first introduce the P2P VoD system model.

We consider a VoD system involving M ISPs. ISP m has N_m participating peers in the VoD system. The average peer upload bandwidth in ISP m is U_m. The VoD system supplies multiple video channels. The videos are divided into chunks for storage and advertising to neighbors which parts of the video a peer caches [8]. As a peer can watch any chunk in any video at a time, we consider a collection of $J = |\mathcal{C}|$ chunks, $\mathcal{C} = \{c_1, c_2, ..., c_J\}$, regardless of which video they belong to, instead of the channels peers are watching. We define chunk j's popularity as the stationary probability that peers in the VoD system are playing chunk j. $(\pi_1, \pi_2, \pi_3, \ldots, \pi_J)$ denotes the chunk popularity distribution. In VoD systems, peers may backward and replay cached chunks. Let ϕ denote the probability that a peer replays cached chunks. The playback time for one chunk is one unit time. A peer can cache and serve chunks from different videos. Every peer can cache B chunks. $s_i = \{c_1^{(i)}, c_2^{(i)}, ..., c_B^{(i)}\}$ denotes the cache state i. Let Θ be the set of all different cache states of peers, $W = |\Theta|$. The servers have cached all chunks.

2.1 Peers' Cache State

In the VoD system, a peer contributes a storage of size B to cache chunks. The state of a peer's cache is $s_i = \{c_1^{(i)}, c_2^{(i)}, \ldots, c_B^{(i)}\}, 1 \leq i \leq W$. $N_m^{(i)}$ denotes the number of peers with cache state s_i in ISP m. Let γ_i be the stationary proportion of cache states s_i under a specific cache placement strategy. When the peer number in ISP m, N_m, is large enough, $N_m^{(i)} = \gamma_i \cdot N_m$. The proportion of peers caching chunk c_j is $\rho_j = \sum_{i:c_j \in s_i} \gamma_i$. The number of peers caching chunk c_j is $N_m \cdot \rho_j$.

2.2 Chunk Request

In the VoD system, users need to download a chunk before playing it. Different users may play different channels or different parts of videos at the same time. Hence, they may be downloading different video chunks. When peers replay the watched and cached parts, they will not need to download new chunks.

A request for a specific chunk is generated when a peer selects to download the chunk. As peers' playback rate is 1 chunk per unit time, the request rate is at least 1 request per unit time to catch up with the playback. We assume a peer's request rate is 1 request per unit time. The requests for chunks generated by peers in ISP m are the superposition of N_m peers' requests for chunks. As N_m is large, one peer's number of requests for chunks is a general renewal process with relative small intensity. According to Palm-Khintchine theorem [9], the requests for chunks generated by peers in ISP m can be modeled as a Poisson Process, with request rate $\lambda_m = N_m$. Given that when a peer requests for a cached chunk and replays it, the request can be served by its own cache, no downloading is necessary. Hence, with chunk j's popularity, the request rate for chunk j is $r_{m,j} = \lambda_m \cdot \pi_j(1 - \phi)$. The total request rate generated by peers in ISP m that needs downloading chunks is $r_m = \sum_{j=1}^{J} r_{m,j} = \lambda_m \sum_{j=1}^{J} \pi_j \cdot (1 - \phi) = (1 - \phi)\lambda_m$.

The generated chunk requests in ISP m may be served by peers in other ISPs. Let a_{ml} denote the proportion of chunk requests routed from ISP m to ISP l. The total requests for chunk j routed into ISP m is $\nu_{m,j} = \sum_{l=1}^{M} a_{lm} \cdot r_{l,j}$. The total chunk requests routed into ISP m is:

$$\nu_m = \sum_{j=1}^{J} \nu_{m,j} = \sum_{l=1}^{M} a_{lm} \cdot r_l.$$

2.3 Chunk Loss Rate

The VoD streaming is a delay-sensitive application. The requests that can not be served by peers' resource are redirected to servers. Let $L_{m,j}$ be the loss rate of requests for chunk j in ISP m, *i.e.*, the steady state probability that a request for chunk j routed to ISP m is dropped by peers and redirected to servers.

The average loss probability of requests in ISP m is:

$$L_m = \frac{\sum_{j=1}^{J} L_{m,j}\nu_{m,j}}{\nu_m}.$$

The average loss probability in the VoD system is:

$$L = \frac{\sum_{m=1}^{M} L_m \cdot \nu_m}{\sum_{m=1}^{M} \nu_m}.$$

2.4 Cross-ISP Traffic

When serving a chunk request from other ISPs, cross-ISP traffic will be generated. The cross-ISP traffic from other ISPs to ISP m is:

$$T_m = \sum_{l=1, l\neq m}^{M} a_{ml} \cdot r_m \cdot (1 - L_l).$$

By summing up the cross-ISP traffic into all ISPs, we get the total cross-ISP traffic:

$$T = \sum_{m=1}^{M} T_m.$$

We summarize important notations in Table 1 for ease of reference.

3. A MODEL FRAMEWORK FOR CHUNK LOSS IN P2P VOD SYSTEMS

3.1 Loss Network Model

The acceptance and rejection of chunk requests in the P2P VoD system can be modeled as a loss network, which suits the characteristic of zero waiting time for requests in VoD streaming systems [10] [11]. Compared with the basic model of a loss network with terminology based on routes and links, the requests for different chunks correspond to the calls on different routers, the peers with different cache states correspond to the different links. Peers' upload bandwidth correspond to

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 peers in ISP m with cache state s_i.
U_m	average peer upload bandwidth in ISP m.
B	the cache size of a peer.
$\mathcal{C}$	the set of all chunks shared in VoD system.
J	the number of chunks shared in VoD.
Θ	the set of all possible cache states of peers.
W	the number of different cache states.
ρ_j	the proportion of peers that have cached chunk j.
$r_{m,j}$	the request rate for chunk j generated by peers in ISP m.
$\nu_{m,j}$	the request rate for chunk j routed into ISP m.
a_{lm}	the fraction of requests routed from ISP l to ISP m.
ϕ	the probability that a peer replays its cached chunks.
$L_{m,j}$	the loss rate for chunk j in ISP m.
T_m	the cross-ISP traffic flowing into ISP m.
T	the total cross-ISP traffic in VoD system.

the circuits of a link. A peer sending requests for a chunk can link to peers caching the chunk for service. The service time is one unit time. If peers caching the chunk have no enough upload bandwidth, the requests are rejected. We apply the loss network model [11] to calculate the chunk loss rate in the P2P VoD system.

Let $\mathbf{n_m} = \{n_{m,j}\}_{c_j \in \mathcal{C}}$ denote the vector of request numbers for different chunks being served concurrently in ISP m.

The chunk loss probability for chunk j in ISP m, $L_{m,j}$, can be calculated as follows [12]: The requests under service experience a delay of 1 unit time (service time). The loss requests experience a delay of 0. The average delay that chunk requests experience is $D_{m,j} = (1 - L_{m,j}) \cdot 1 + L_{m,j} \cdot 0 = (1 - L_{m,j})$. Upon applying Little's law to the VoD system in ISP m (with respect to chunk j), we obtain $\nu_{m,j} D_{m,j} = \mathbf{E}[n_{m,j}]$, which yields

$$L_{m,j} = 1 - \frac{\mathbf{E}[n_{m,j}]}{\nu_{m,j}}.$$

Hence, the problem of obtaining the loss probability $L_{m,j}$ becomes deriving $\mathbf{E}[n_{m,j}]$. We take the 1-point approximate algorithm, using $n^*_{m,j}$, which is the element of $\mathbf{n^*_m}$, the state having the maximum probability, as a surrogate of $\mathbf{E}[n_{m,j}]$ [12]. Relaxing integer vector $\mathbf{n_m}$ using a real vector $\mathbf{x_m}$. $\mathbf{n^*_m}$ satisfies the following optimization problem [11]:

$$\max \textstyle\sum_{j=1}^{J} x_{m,j} log \nu_{m,j} - x_{m,j} log x_{m,j} + x_{m,j}$$
$$\text{over } \forall \mathcal{A} \subseteq \mathcal{C}, \textstyle\sum_{c_j \in \mathcal{A}} x_{m,j} \leq U_m \cdot \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)}$$
$$\mathbf{x_m} \geq 0$$

The corresponding Lagrangian is:

$$\begin{aligned} L(\mathbf{x_m}, \epsilon) &= \sum_{j=1}^{J} (x_{m,j} log \nu_{m,j} - x_{m,j} log x_{m,j} + x_{m,j}) \\ &+ \sum_{\mathcal{A} \subseteq \mathcal{C}} \epsilon_{\mathcal{A}} \cdot (U_m \cdot \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)} - \sum_{j=1}^{M} x_{m,j}) \\ &= \sum_{j=1}^{J} x_{m,j} + \sum_{j=1}^{J} x_{m,j} (log \nu_{m,j} - log x_{m,j} \\ &- \sum_{c_j \in \mathcal{A}, \mathcal{A} \subseteq \mathcal{C}} \epsilon_{\mathcal{A}}) + \sum_{\mathcal{A} \subseteq \mathcal{C}} \epsilon_{\mathcal{A}} \cdot U_m \cdot \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)} \end{aligned}$$

The KKT conditions for this convex optimization problem are:

$$\forall \mathcal{A} \subseteq \mathcal{C}, \sum_{c_j \in \mathcal{A}} x_{m,j} \leq U_m \cdot \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)} \quad (1)$$

$$\epsilon_{\mathcal{A}} \geq 0 \quad (2)$$

$$\forall \mathcal{A} \subseteq \mathcal{C}, \epsilon_{\mathcal{A}} \cdot (U_m \cdot \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)} - \sum_{c_j \in \mathcal{A}} x_{m,j}) = 0 \quad (3)$$

$$x_{m,j} = \nu_{m,j} \cdot \exp(- \sum_{c_j \in \mathcal{A}, \mathcal{A} \subseteq \mathcal{C}} \epsilon_{\mathcal{A}}) \quad (4)$$

When we get $\mathbf{x_m}$ from the above KKT conditions, we can calculate the system average chunk loss rate as:

$$L_m = \frac{\sum_{j=1}^{J} (1 - \frac{x_{m,j}}{\nu_{m,j}}) \cdot \nu_{m,j}}{\nu_m} = 1 - \frac{\sum_{j=1}^{J} x_{m,j}}{\nu_m} \quad (5)$$

3.2 Maximum Bipartite Flow Map of the KKT Conditions

The total served chunk requests are $\sum_{j=1}^{J} x_{m,j}$, which is the sum of the solutions of the KKT conditions. The number of functions in KKT conditions grows exponentially with the number of chunks, which makes it computationally complex to solve the KKT conditions. We prove that the total served request rates got from solutions of the KKT conditions can be mapped into the maximum bipartite flow in the corresponding bipartite graph (Fig. 1).

We first present the corresponding bipartite graph (Fig. 1) with source node s and destination node t. The bipartite graph has two sets of nodes: chunk set $\mathcal{C}$ and peer cache state set Θ. The left set of nodes, $\mathcal{C}$, represent different chunks, the right set of nodes, Θ, represent different peer cache states. The edges directed from nodes in $\mathcal{C}$ to nodes in Θ represent that the cache state contains the chunk, and requests for a chunk can

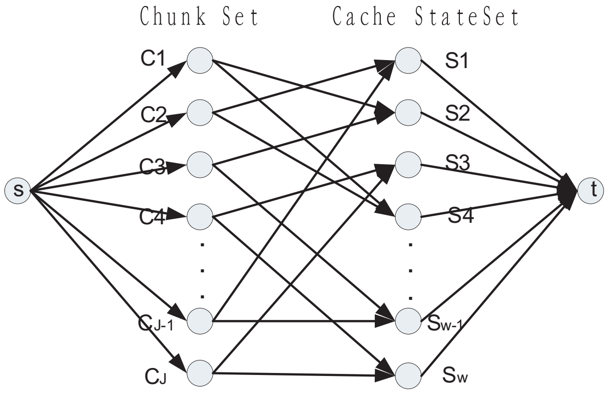

Figure 1: Corresponding Bipartite Graph.

be sent to those states caching the chunk for downloading service. The edges from source s to any node in set $\mathcal{C}$ represent the requests for different chunks in ISP m. The corresponding edge capacity from source s to node C_j is the chunk request rate for chunk j in ISP m, $\nu_{m,j}$. The edges from $c_j \in \mathcal{C}$ to $s_i \in \Theta, c_j \in s_i$ have the capacity, $\nu_{m,j}$, which can not exceed the total request rate served by ISP m. The edges from any node in Θ to the destination t represent cache states serving the requests. The edge capacity is the total upload bandwidth of a cache state, which is $N_m^{(i)} \cdot U_m$.

Theorem 1. *The total served request rate, $\sum_{j=1}^{J} x_{m,j}$, obtained from the KKT conditions is the maximum bipartite flow of the corresponding bipartite graph (Fig. 1).*
Proof: We first construct a graph cut whose cutset equals to the total served request rate obtained from the KKT conditions. We prove that this cutset is the minimum cut. Applying the min-cut-max-flow theorem, we can show that the total served request rate is the maximum bipartite flow.

Let $x_{m,j}, 1 \leq j \leq J$ denote the solutions of the KKT conditions for the serving request numbers. We can divide the $x_{m,j}$'s into two classes according whether $x_{m,j}$ is equal to $\nu_{m,j}$ or smaller than $\nu_{m,j}$. $\mathcal{C}_1 = \{c_j | x_{m,j} = \nu_{m,j}, 1 \leq j \leq J\}$, $\mathcal{C}_2 = \{c_j | x_{m,j} < \nu_{m,j}, 1 \leq j \leq J\}$. The peers' upload bandwidth for the chunks in set $\mathcal{C}_2$ is not enough. From the KKT conditions, for all $\mathcal{A}$, when $\mathcal{A}$ includes $c_j \in \mathcal{C}_1$, $\epsilon_{\mathcal{A}} = 0$; for $\mathcal{A} = \mathcal{C}_2$, as the peers' total upload bandwidth is not enough for chunks in $\mathcal{C}_2$, 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 (3), we have $\epsilon_{\mathcal{C}_2} > 0$. Hence, it is easy to verify that the cutset from the set including source node s, all nodes in set $\mathcal{C}_2$, nodes in set Θ that have connections with nodes in set $\mathcal{C}_2$ to the residual set of the graph is $\sum_{c_j \in \mathcal{C}_1} \nu_{m,j} + U_m \cdot \sum_{i: s_i \cap \mathcal{C}_2 \neq \emptyset} N_m^{(i)}$, equal to $\sum_{j=1}^{J} x_{m,j}$. In the following step, we prove that this cutset is the minimum cut.

On one hand, when the set including source node s contains a node c_j from $\mathcal{C}_1$, as the capacity of edges from node c_j to nodes in Θ is equal to that of the edge from source s to node c_j, the resulted cutset is no smaller than that of excluding the node from $\mathcal{C}_1$. When all nodes in Θ having connections with c_j are also included in the set. As a result, the total capacity of cutset is increased by $U_m \cdot \sum_{s_i : c_j \in s_i} N_m^{(i)} - \nu_{m,j} \geq 0, c_j \in \mathcal{C}_1$ according to KKT condition (1). On the other hand, when the set including source node s excludes a node c_k from $\mathcal{C}_2$, the cutset is increased by $\nu_{m,k}$. As a node s_i in Θ having connections with c_k is also excluded from the set including source node s, the cutset will be changed by $\sum_{j: j \in s_i} \nu_{m,j} - U_m N_m^{(i)}$, which is larger than 0 as the peer upload bandwidth for requests of chunks in $\mathcal{C}_2$ is not enough. Hence, the total capacity will be increased.

The cutset 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 total served request rate is the maximum bipartite flow.

4. OPTIMAL CACHE CONDITION AND OPTIMAL CHUNK REQUEST ROUTING

The model framework can use the maximum bipartite flow algorithm [13] to calculate the average chunk request loss rate under a specific cache state distribution and a specific chunk request rate distribution. We are especially interested in the chunk request loss rate under the optimal peer cache distribution. Hence, we first state the optimal peer cache condition. We propose a concrete optimal cache placement strategy and prove it satisfies the optimal cache condition. We also analyze the Least Recently Used (LRU) cache replacement strategy and show that it achieves the optimal cache distribution in its stationary states. With the optimal peer cache, we propose the optimal chunk request routing among different ISPs.

4.1 Optimal Cache Condition in P2P VoD System

We define that the ISP m's work load, η_m, is the ratio of the total number of chunk requests to the peers' total upload bandwidth:

$$\eta_m = \frac{\nu_m}{N_m U_m}.$$

The optimal cache distribution in P2P VoD system should make the chunk requests can be served when peers' upload bandwidth is available. Hence, under the optimal cache, the chunk request loss rate is related to the ISP's work load:

$$L_m = max\{0, 1 - \frac{1}{\eta_m}\}.$$

We have the following lemma for the optimal cache condition.

Lemma 1. When the number of peers with cache state s_i in ISP m, $N_m^{(i)}, 1 \leq i \leq W$, satisfies the following inequalities:

$$\forall \mathcal{A} \subseteq \mathcal{C}, \sum_{c_j \in \mathcal{A}} \nu_{m,j} \leq \eta_m \cdot U_m \cdot \sum_{i:\mathcal{A} \cap s_i \neq \emptyset} N_m^{(i)} \qquad (6)$$

The system can achieve the chunk request loss rate $L_m = max\{0, 1 - \frac{1}{\eta_m}\}$.

proof: When $N_m^{(i)}$'s satisfy (6), let $\epsilon_{\mathcal{A}} = 0$ for all $\mathcal{A} \subset \mathcal{C}$, $\epsilon_{\mathcal{A}} = max\{0, \ln \eta_m\}$ for $\mathcal{A} = \mathcal{C}$. It is easy to verify that $x_{m,j} = min\{\nu_{m,j}, \frac{\nu_{m,j}}{\eta_m}\}, 1 \leq j \leq J$ are the solutions of the KKT conditions. Hence, the average chunk request loss rate is $L_m = max\{0, 1 - \frac{1}{\eta_m}\}$.

Lemma 2 states a concrete optimal cache placement strategy.

Lemma 2. *One optimal cache placement strategy is to let chunk c_j be randomly cached in a proportion of $\rho_j = B \cdot \frac{\nu_{m,j}}{\nu_m} = B \cdot \pi_j$ peers.*

proof: We prove it by showing that the optimal strategy in Lemma 2 satisfies the optimal cache condition in Lemma 1. We use the induction method.

(i) When $|\mathcal{A}| > J - B$, the intersection between any cache state $s_i, 1 \leq i \leq W$ and $\mathcal{A}$ will not be empty. Hence, the L.H.S of (6) equals to $\sum_{c_j \in \mathcal{A}} \nu_{m,j}$, which is smaller than the R.H.S of (6), $\eta_m \cdot U_m \cdot N_m = \nu_m$.

(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.

$$\begin{aligned}
&\text{R.H.S of (6)} = \eta_m \cdot U_m (N_m - \sum_{i:s_i = \bar{\mathcal{A}}_{J-B}} N_m^{(i)}) \\
&= \eta_m \cdot U_m (N_m - N_m \cdot \prod_{c_j \in \bar{\mathcal{A}}_{J-B}} \rho_j) \\
&\geq \eta_m \cdot U_m (N_m - N_m \cdot (\frac{B}{\nu_m})^B \prod_{c_j \in \bar{\mathcal{A}}_{J-B}} \nu_{m,j}) \\
&\geq \eta_m \cdot U_m (N_m - N_m \cdot (\frac{B}{\nu_m})^B \cdot (\frac{\sum_{c_j \in \bar{\mathcal{A}}_{J-B}} \nu_{m,j}}{B})^B) \\
&\geq \nu_m - (\sum_{c_j \in \bar{\mathcal{A}}_{J-B}} \nu_{m,j}) \cdot (\frac{\sum_{c_j \in \bar{\mathcal{A}}_{J-B}} \nu_{m,j}}{\nu_m})^{B-1} \\
&\geq \sum_{c_j \in \mathcal{A}_{J-B}} \nu_{m,j} = \text{L.H.S of (6)}
\end{aligned}$$

(iii) For $k \leq J - B$, suppose $\forall \mathcal{A}_k$ satisfy (6). Let's see 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 a $\mathcal{A}_k = c_j \cup \mathcal{A}_{k-1}$. Hence, we get $(J-k+1)$ sets of $\mathcal{A}_k$. 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 (6),

$$\sum_{c_j \in \mathcal{A}_k^t} \nu_{m,j} \leq \eta_m \cdot U_m \cdot \sum_{i:\mathcal{A}_k^t \cap s_i \neq \emptyset} N_m^{(i)}, 1 \leq t \leq J-k+1$$

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

$$\begin{aligned}
&\sum_{t=1}^{J-k+1} \sum_{c_j \in \mathcal{A}_k^t} \nu_{m,j} = (J-k+1) \sum_{c_j \in \mathcal{A}_{k-1}} \nu_{m,j} + \sum_{c_j \notin \mathcal{A}_{k-1}} \nu_{m,j} \\
&\leq \sum_{t=1}^{J-k+1} \eta_m \cdot U_m \cdot \sum_{i:\mathcal{A}_k^t \cap s_i \neq \emptyset} N_m^{(i)} \\
&= (J-k+1)\eta_m \cdot U_m \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)} \\
&+ \eta_m \cdot U_m \sum_{i:s_i \cap \mathcal{A}_{k-1} = \emptyset} N_m^{(i)}
\end{aligned}$$

Hence,

$$\begin{aligned}
&(J-k+1) \sum_{c_j \in \mathcal{A}_{k-1}} \nu_{m,j} + \nu_m - \sum_{c_j \in \mathcal{A}_{k-1}} \nu_{m,j} \\
&\leq (J-k+1)\eta_m \cdot U_m \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)} \\
&+ \eta_m \cdot U_m (N_m - \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)})
\end{aligned}$$

We have,

$$\sum_{c_j \in \mathcal{A}_{k-1}} \nu_{m,j} \leq \eta_m \cdot U_m \sum_{i:\mathcal{A}_{k-1} \cap s_i \neq \emptyset} N_m^{(i)}$$

Hence, we prove Lemma 2.

4.2 Optimality Property of LRU Cache Replacement Algorithm

With the optimal cache placement strategy, in this part, we theoretically prove that the system's stationary cache state distribution under Least Recently Used (LRU) cache replacement strategy is the optimal cache distribution proposed in Lemma 2. This is consistent with the work of Wu *et. al.* [14], which states that LRU cache replacement algorithm performs as well as the optimal cache replacement strategy.

Let us consider chunk j's position in a peer's cache n time unit after the peer starts playing videos, under LRU algorithm. We define a chunk's position in a peer's cache as the chunk's state, which is denoted by s_n^j. We first derive the state transition equations. Note that users watching videos have a probability ϕ of replaying videos. As users replay videos, they watch the chunks that are already cached in the peer's cache, users do not need to download new chunks and the replayed chunk's position will change to 1 in the cache. Hence, the probability of playing new contents is $1 - \phi$. We assume when a user replays videos, the starting positions of replaying have a uniformly distribution among cached chunks. Hence, the cached chunk will be randomly uniformly played. When users play new contents, a new chunk will be downloaded and cached at position 1 in the cache. The chunk in the last position of cache, position B, will be evicted. Positions of all other cached chunks will increase by 1. Given s_n^j, we can derive the probability for chunk j's position at time $n+1$, s_{n+1}^j.

When $2 \leq b \leq B$,

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

The event that chunk j's position increases by 1 when $s_n^j = b-1$ can be divided into two disjoint events: one is that the peer plays a new chunk; the other is that the peer replays a chunk cached at positions behind $b-1$:

$$\begin{aligned}&Pr[c_j\text{'s position increases by } 1 | s_n^j = b-1] = \\ &(1-\phi) + \phi \cdot \frac{B-b+1}{B}\end{aligned}$$

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

$$Pr[c_j\text{'s position does not change} | s_n^j = b] = \phi \cdot \frac{b-1}{B}$$

Hence,

$$\begin{aligned}Pr[s_{n+1}^j = b | s_n^j] &= (1 + \phi \cdot \tfrac{1-b}{B}) \cdot Pr[s_n^j = b-1] \quad (7) \\ &+ \phi \cdot \tfrac{b-1}{B} \cdot Pr[s_n^j = b]\end{aligned}$$

Equation (8) **(should be (7))** shows that the next state of chunk j is only related to the previous state of chunk j, we can use Markov Chain to model the change of chunk j's states. We use state $b, 1 \leq b \leq B$, to denote chunk j's position is at cell b of the peer's cache. State 0 denotes chunk j is not in the peer's cache. s_{n+1}^j denotes the state of chunk j at time slot $n+1$.

We analyze the stationary state distribution for a peer caching chunk j, which is the probability distribution for chunk j's positions at a peer's cache. Let s^j denote the stationary state of chunk j when n increases to infinity. Based on equation (8) **(should be (7))**, for $2 \leq b \leq B$, we have,

$$\begin{aligned}Pr[s^j = b] &= (1 + \phi \cdot \tfrac{1-b}{B}) \cdot Pr[s^j = b-1] \\ &+ \phi \cdot \tfrac{b-1}{B} \cdot Pr[s^j = b]\end{aligned}$$

Hence,

$$Pr[s^j = 1] = Pr[s^j = 2] = \ldots = Pr[s^j = B].$$

This implies that a chunk may be cached in different positions with the equal probabilities. $Pr[s^j = 1]$ is the probability that chunk j is cached at a peer cache's position 1, which equals to the probability that a peer is playing chunk j, i.e., the popularity of chunk j. Hence, $Pr[s^j = 1] = \pi_j$. Hence, under LRU algorithm, the probability that a peer caches chunk j is $B \cdot Pr[s^j = 1] = B \cdot \pi_j$, which also means the proportion of peers caching chunk j is $\rho_j = B \cdot Pr[s^j = 1] = B \cdot \pi_j$. This is just the optimal cache distribution proposed in Lemma 2.

4.3 Optimal Chunk Request Routing Condition

To achieve the minimum average chunk request loss rate, besides placing cache optimally, peers in P2P VoD systems need to route their chunk requests among different ISPs appropriately.

We have the average chunk request loss rate in ISP m:

$$L_m = max\{1 - \frac{U_m \cdot N_m}{\sum_{j=1}^{J} \nu_{m,j}}, 0\}.$$

The average chunk request loss rate for the VoD system is:

$$L = \frac{\sum_{m=1}^{M} L_m \cdot \nu_m}{\sum_{m=1}^{M} \nu_m} = \frac{\sum_{m=1}^{M} max\{\nu_m - U_m \cdot N_m, 0\}}{\sum_{m=1}^{M} \nu_m}.$$

Hence, to minimize system chunk request loss rate, we have:

$$\begin{aligned}min \quad & \tfrac{\sum_{m=1}^{M} max\{\nu_m - U_m \cdot N_m, 0\}}{\sum_{m=1}^{M} \nu_m} \\ over \quad & \sum_{l=1}^{M} a_{lm} \cdot r_l = \nu_m \\ & \sum_{m=1}^{M} a_{lm} = 1, 1 \leq l \leq M \qquad (8)\end{aligned}$$

From (8), we can derive the conditions a_{ml}'s should satisfy to achieve the minimum system chunk request loss rate, i.e., the optimal chunk request routing conditions. Let us consider (8) in two situations:

1) The peers' total upload bandwidth is no smaller than the demand for serving chunk requests, i.e., $\sum_{m=1}^{M} U_m N_m \geq \sum_{m=1}^{M} \nu_m$.

In this situation, there exist a_{lm}'s, which satisfy $\nu_m = \sum_{l=1}^{M} a_{lm} \cdot r_l \leq U_m N_m$, for $1 \leq m \leq M$. Hence, $max\{\nu_m - U_m \cdot N_m, 0\} = 0$, the objective function in (8) can take minimum value $L = 0$. The optimal chunk request routing condition that the value of a_{lm}'s should satisfy is:

$$\nu_m = \sum_{l=1}^{M} a_{lm} r_l \leq U_m N_m, 1 \leq m \leq M \qquad (9)$$

2) The peers' total upload bandwidth is smaller than the demand for serving chunk requests, i.e., $\sum_{m=1}^{M} U_m N_m < \sum_{m=1}^{M} \nu_m$.

In this situation, we have

$$\begin{aligned}L &= \tfrac{\sum_{m=1}^{M} max\{\nu_m - U_m \cdot N_m, 0\}}{\sum_{m=1}^{M} \nu_m} \\ &\geq \tfrac{max\{\sum_{m=1}^{M}(\nu_m - U_m \cdot N_m), 0\}}{\sum_{m=1}^{M} \nu_m} = \tfrac{\sum_{m=1}^{M}(\nu_m - U_m \cdot N_m)}{\sum_{m=1}^{M} \nu_m} \\ &= 1 - \tfrac{\sum_{m=1}^{M} U_m N_m}{\sum_{m=1}^{M} \nu_m}\end{aligned}$$

Hence, the minimum system chunk request loss rate is $L = 1 - \frac{\sum_{m=1}^{M} U_m N_m}{\sum_{m=1}^{M} \nu_m}$. To obtain this minimum system chunk request loss rate, we just need

$$\frac{\sum_{m=1}^{M} max\{\nu_m - U_m \cdot N_m, 0\}}{\sum_{m=1}^{M} \nu_m} = \frac{max\{\sum_{m=1}^{M}(\nu_m - U_m \cdot N_m), 0\}}{\sum_{m=1}^{M} \nu_m}.$$

Hence, the optimal chunk request routing condition a_{ml}'s should satisfy is:

$$\nu_m = \sum_{l=1}^{M} a_{lm} r_l \geq U_m N_m, 1 \leq m \leq M. \tag{10}$$

We can combine condition (9) and condition (10) under the two different situations into one condition:

$$(\sum_{m=1}^{M} \nu_m - \sum_{m=1}^{M} U_m N_m)(\sum_{l=1}^{M} a_{lm} r_l - U_m N_m) \geq 0. \tag{11}$$

5. CROSS-ISP TRAFFIC AND PERFORMANCE RELATIONSHIP

P2P VoD systems can achieve the minimum system chunk request loss rate under optimal cache condition and optimal chunk request routing condition. In this section, we study what is the minimum volume of necessary cross-ISP traffic when P2P VoD systems achieve the minimum system chunk request loss rate; how the volume of cross-ISP traffic impacts the loss rate. We assume peers have the optimal cache distribution. This implies that we just need to consider the limitation of peers' total upload bandwidth.

5.1 Minimum Cross-ISP Traffic for Minimum Chunk Loss Rate

The minimum cross-ISP traffic when achieving minimum chunk loss rate can be derived from the following optimization problem:

$$\begin{array}{ll} min & T = \sum_{m=1}^{M} T_m = \sum_{m=1}^{M} \sum_{k \neq m} a_{mk} \cdot r_m \cdot (1 - L_k) \\ over & (11); \\ & \sum_{m=1}^{M} a_{lm} = 1, 1 \leq l \leq M \end{array} \tag{12}$$

The optimal chunk request routing solutions, a_{ml}'s, for (12) is:
For ISPs with $I_m = U_m \cdot N_m - r_m \geq 0$,

$$a^*_{ml} = \begin{cases} 1 & : \quad l = m, \\ 0 & : \quad l \neq m, 1 \leq l \leq M. \end{cases}$$

For ISPs with $I_m = U_m \cdot N_m - r_m < 0$,

$$a^*_{ml} = \begin{cases} max\{1 - \frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s}(\frac{-I_m}{r_m}), \frac{N_m U_m}{r_m}\} & : \quad l = m, \\ min\{\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s}, 1\}(\frac{-I_m}{r_m}) \cdot \frac{I_l}{\sum_{t,I_t>0} I_t} & : \quad l \neq m, I_l \geq 0, \\ 0 & : \quad l \neq m, I_l < 0. \end{cases}$$

Proof: Let us consider two situations:

1) The peers' total upload bandwidth is no smaller than the demand for serving chunk requests, i.e., $\sum_{m=1}^{M} U_m N_m \geq \sum_{m=1}^{M} \nu_m$. In this situation, $\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} \geq 1$. Hence, for ISPs with $I_m < 0$,

$$a^*_{ml} = \begin{cases} \frac{N_m U_m}{r_m} & : \quad l = m, \\ (\frac{-I_m}{r_m}) \cdot \frac{I_l}{\sum_{t,I_t>0} I_t} & : \quad l \neq m, I_l \geq 0, \\ 0 & : \quad l \neq m, I_l < 0. \end{cases}$$

First, it is not difficult to verify that a^*_{ml}'s satisfy the constraints which are necessary for minimum system chunk request loss rate. For ISPs with $I_m < 0$, $\nu^*_m = \sum_{l=1}^{M} a^*_{lm} r_l = a^*_{mm} r_m = U_m N_m$; for ISPs with $I_m \geq 0$, $\nu^*_m = \sum_{l=1}^{M} a^*_{lm} r_l = r_m + \frac{-\sum_{l,I_l<0} I_l}{\sum_{t,I_t>0} I_t} I_m \leq U_m N_m$. Hence, $L^*_m = 0$ for $1 \leq m \leq M$.

Second, we show that $T^* = \sum_{m=1}^{M} \sum_{k \neq m} a^*_{mk} \cdot r_m \cdot (1 - L_k) = \sum_{m=1}^{M} (1 - a^*_{mm}) \cdot r_m = \sum_{m,I_m<0} (r_m - U_m N_m)$ is the minimum cross-ISP traffic to achieve the minimum chunk request loss rate. Suppose there are a set of a'_{ml}'s, different from a^*_{ml}'s, satisfying the constraints and inducing the volume of cross-ISP traffic $T' < T^*$. Hence $L'_m = L^*_m = 0$ for $1 \leq m \leq M$. $T' = \sum_{m=1}^{M} \sum_{k \neq m} a'_{mk} \cdot r_m \cdot (1 - L'_k) = \sum_{m=1}^{M} (1 - a'_{mm}) \cdot r_m < T^*$. 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 $a'_{mm} > a^*_{mm} = \frac{N_m U_m}{r_m}$, $\nu'_m \geq a'_{mm} \cdot r_m > N_m U_m$, this contradicts with $L'_m = 0$. Hence, such a'_{ml}'s do not exist. a^*_{ml}'s are the optimal solutions. $T^* = \sum_{m,I_m<0} (r_m - U_m N_m)$ is the minimum cross-ISP traffic.

2) The peers' total upload bandwidth is smaller than the demand for serving chunk requests, i.e., $\sum_{m=1}^{M} U_m N_m < \sum_{m=1}^{M} \nu_m$. In this situation, $\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} < 1$. Hence, for ISPs with $I_m < 0$,

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

First, for ISPs with $I_m < 0$, $\nu^*_m = r_m \cdot a^*_{mm} = r_m + I_m \cdot \frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} = I_m(\frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s} - 1) + U_m N_m \geq U_m N_m$. For ISPs with $I_m > 0$, $\nu^*_m = r_m + \sum_{l,I_l<0} a^*_{lm} r_l = r_m + I_m = U_m N_m$. Hence, a^*_{ml}'s satisfy condition (10). For ISPs with $I_m \geq 0$, $L^*_m = 0$.

Second, we will show $T^* = \sum_{m=1}^{M} \sum_{k \neq m} a^*_{mk} \cdot r_m \cdot (1 - L_k) = \sum_{m,I_m<0} (1 - a^*_{mm}) r_m = \sum_{m,I_m<0} \frac{\sum_{t,I_t>0} I_t}{-\sum_{s,I_s<0} I_s}(-I_m) = \sum_{t,I_t>0} I_t$ is the minimum cross-ISP traffic. Suppose there are a set of a'_{ml}'s, different from a^*_{ml}'s, satisfying the optimal chunk request routing constraint (10). Hence, for ISPs with $I_m \geq 0$, $\nu'_m \geq \nu^*_m = U_m N_m$. $T' = \sum_{m=1}^{M} \sum_{k \neq m} a^*_{mk} \cdot r_m \cdot (1 - L_k) = \sum_{k=1}^{M} \frac{U_k N_k}{\nu'_k}(\nu'_k - r_k a'_{kk}) \geq \sum_{k,I_k>0} \frac{U_k N_k}{\nu'_k}(\nu'_k - r_k a'_{kk}) \geq \sum_{k,I_k>0} (U_k N_k - r_k \cdot a'_{kk} \cdot \frac{U_k N_k}{\nu'_k}) \geq \sum_{k,I_k>0} (U_k N_k - r_k) = T^*$. Hence, T^* is the minimum cross-ISP traffic.

Based on the optimal solutions a^*_{ml}'s, we propose an

ISP-aware algorithm to reduce unnecessary cross-ISP traffic while minimizing chunk loss probability.

ISP-aware Chunk Request Routing Algorithm: (1) The tracker sorts the ISPs according to the value of $I_m = (U_m \cdot N_m - r_m)$, which has a positive value when peer resource in ISP m is larger than its generated chunk requests, a negative value when peer resource in ISP m is smaller than its generated chunk requests;
(2) The tracker sets the value of a_{ml}'s equal to a_{ml}^*'s;
(3) Each peer in ISP m keeps a proportion of a_{ml}^* neighbors from ISP l, $1 \leq l \leq M$. Peers update their neighbor lists to keep peers in their neighbor lists having chunks they want to download. They randomly send their chunk requests to their neighbors. Hence, with probability a_{ml}^*, the chunk requests generated in ISP m will be routed to peers in ISP l.

5.2 The Impact of Cross-ISP Traffic on Chunk Request Loss Rate

In this part, we study how the volume of cross-ISP traffic will affect the P2P VoD streaming system's minimum chunk request loss rate.

Assume the constraint on cross-ISP traffic is that the volume of cross-ISP traffic should be no larger than T^c. As $T^c \geq T^*$, we could apply the ISP-aware chunk request routing algorithm to achieve the minimum chunk request loss rate and the cross-ISP traffic will be $T^* \leq T^c$. In the following, we study the minimum chunk request loss rate under the situations where $T^c < T^*$. This can be formulated as:

$$\begin{array}{ll} min & L = \frac{\sum_{m=1}^{M} max\{\nu_m - U_m \cdot N_m, 0\}}{\sum_{m=1}^{M} \nu_m} \\ over & T = \sum_{m=1}^{M} \sum_{k \neq m} a_{mk} \cdot r_m \cdot (1 - L_k) \leq T^c \end{array}$$

Let a_{ml}^c's denote the solutions for the above optimization problem. When $T^c = 0$, a_{mm}^c must equals to 1 for $1 \leq m \leq M$,

$$L_{min}^c = \frac{-\sum_{m, I_m < 0} I_m}{\sum_{m=1}^{M} \nu_m}$$

When $T^c < T^*$, fewer proportion of chunk requests will be routed to other ISPs, hence, a_{mm}^c's satisfy $a_{mm}^* < a_{mm}^c < 1$ for ISPs with $I_m < 0$, $a_{mm}^c = 1$ for ISPs with $I_m \geq 0$. For ISPs with $I_m < 0$, $\nu_m^c = a_{mm}^c r_m > a_{mm}^* r_m \geq U_m N_m$; for ISPs with $I_m \geq 0$, $\nu_m^c \leq \nu_m^* \leq U_m N_m$.

$$\begin{aligned} L_{min}^c &= \frac{\sum_{m, I_m < 0} \nu_m^c - U_m N_m}{\sum_{m=1}^{M} \nu_m} \\ &= \frac{-\sum_{m, I_m < 0} I_m - T^c}{\sum_{m=1}^{M} \nu_m} \end{aligned}$$

6. PERFORMANCE EVALUATION

In this section, we carry out numerical analyses for relationship between chunk loss rates and cross-ISP traffic using parameters driven from the empirical data in the real-world [15]. There are in total $M = 10$ ISPs in the system. The total number of concurrent users in the system is $N = 100000$. The users distribute in different ISPs according to the probability distribution function $p_m = \frac{(M-m+1)^\beta}{\sum_{m=1}^{M} (M-m+1)^\beta}, \beta \geq 0$. $N_m = p_m \cdot N$. The user distribution among different ISPs is more unbalanced when β is larger. The average upload bandwidth of ISP m equals to $U_m = 1 + \frac{\gamma - m}{10}, 1 \leq \gamma \leq 10$. Chunk is the unit for storage and advertising to neighbors what parts of a movie a peer caches [16]. A chunk usually has a size of several MB. The total number of different chunks shared in the system is 5000. Every peer has a cache of 200 chunks. We use the Zipf-Mandelbrot model to fit the chunk popularity distribution: $\pi_j = \frac{\frac{1}{(j+q)^\alpha}}{\sum_{j=1}^{J} \frac{1}{(j+q)^\alpha}}$, $\alpha = 0.78, q = 4$. The number of peer neighbors is $d = 30$.

6.1 Optimal Cache vs. Unoptimal Cache

We implement a maximum bipartite flow algorithm to simulate the process of serving chunk requests. One partition is the chunk requests that ask for download service. The other partition is peers who accept or reroute the chunk requests. When a chunk request is accepted (i.e., served) by a peer, the tracker will record it and add it to the number of served chunk requests. We simulate the optimal cache and two unoptimal cache cases. In the optimal cache case, we place the chunks in peers' cache according to the actual chunk popularity with parameters $\alpha = 0.78, q = 4$. In unoptimal cache case 1, we place the chunks in peers' cache according to the estimated chunk popularity with parameters $\alpha = 0.60, q = 3$. In unoptimal cache case 2, we estimate $\alpha = 1.20, q = 2$.

We first simulate the chunk request loss rate in an ISP with 10000 peers and peers' average upload capacity $U_m = 1.4$. The number of chunk requests is increased from 10000 to 20000 and the proportion of every chunk is generated according to chunk popularity distribution: $\pi_j = \frac{\frac{1}{(j+q)^\alpha}}{\sum_{j=1}^{J} \frac{1}{(j+q)^\alpha}}$, $\alpha = 0.78, q = 4$. Fig. 2 compares the chunk request loss rate under optimal cache, unoptimal cache to theoretical results. We see that when the number of chunk requests is around 14000, which equals to the peers' total upload capacity, the chunk request loss rate increases the most largely. This implies the cache strategy has a large impact on the performance when the peers' upload capacity is equal to the number of chunk requests. When the peers' total upload bandwidth is insufficient, the bandwidth is the main impact on the chunk request loss rate.

Fig. 3 shows the chunk request loss rate in different ISPs under optimal cache, unoptimal cache case 1 and unoptimal cache case 2 and the theoretical minimum chunk request loss rate.

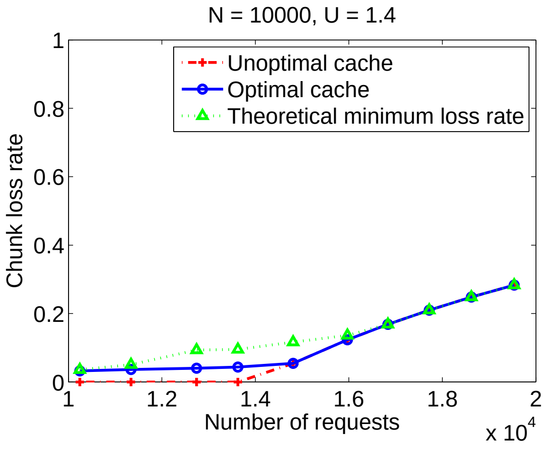

Figure 2: chunk request loss rate in an ISP under optimal cache, unoptimal cache, theoretical results as the number of chunk requests increases.

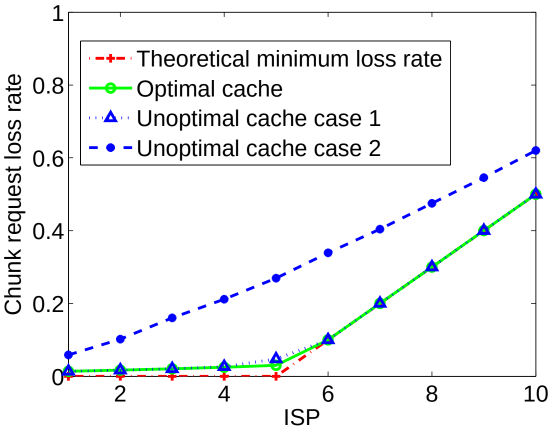

Figure 3: chunk request loss rate in 10 different ISPs under optimal cache, unoptimal cache, theoretical results.

6.2 Cross-ISP Traffic vs. Chunk Loss Probability

We first evaluate how peer distribution in ISPs and system average upload bandwidth will affect the minimum cross-ISP traffic and system minimum chunk request loss rate.

Fig. 4 and Fig. 5 show how the minimum cross-ISP traffic changes with the peer distribution and system average upload bandwidth. In Fig. 4, when β is smaller than 0, more peers are in ISPs with smaller peer upload bandwidth, when $\beta = 0$, the peers are equally distributed in different ISPs, when $\beta > 0$, more peers are in ISPs with larger peer upload bandwidth. We see that the minimum cross-ISP traffic to achieve the optimal performance is the largest when peers are equally distributed in different ISPs, and the minimum cross-ISP traffic increases as the system average upload bandwidth is close to video playback rate. In Fig. 5, we see that the minimum cross-ISP traffic has the largest value when γ reaches a specific value, which means the system average upload capacity is the closest to video playback rate. Fig. 6 shows system chunk request loss rate decreases as system upload bandwidth increases.

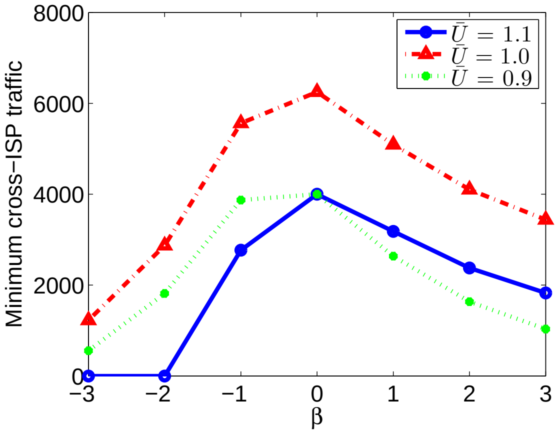

Figure 4: Minimum cross-ISP traffic under different peer distribution.

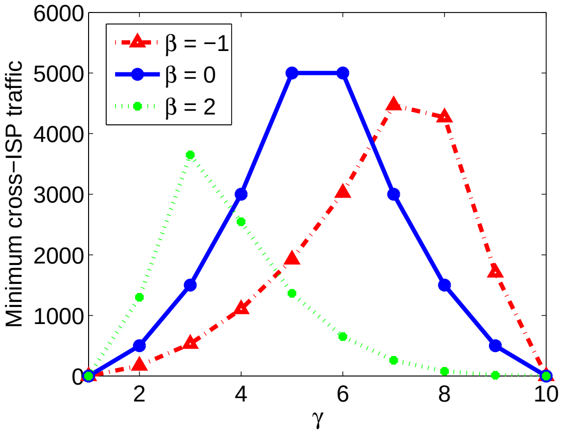

Figure 5: Minimum cross-ISP traffic under different system average upload bandwidth.

We then evaluate different ISP's necessary cross-ISP traffic and chunk loss rate under ISP-aware chunk request routing algorithm. We simulate four scenarios:
Scenario 1: $\beta = 0.5, \gamma = 3$;
Scenario 2: $\beta = 1, \gamma = 3$;
Scenario 3: $\beta = 0.5, \gamma = 6$;
Scenario 4: $\beta = 1, \gamma = 6$;

Fig. 7 & Fig. 8 plots each ISP's chunk loss rate and necessary cross-ISP traffic under ISP-aware chunk re-

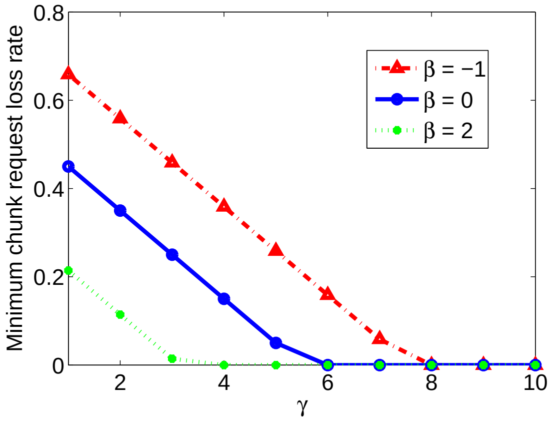

Figure 6: Minimum chunk request loss rate under different system average upload bandwidth.

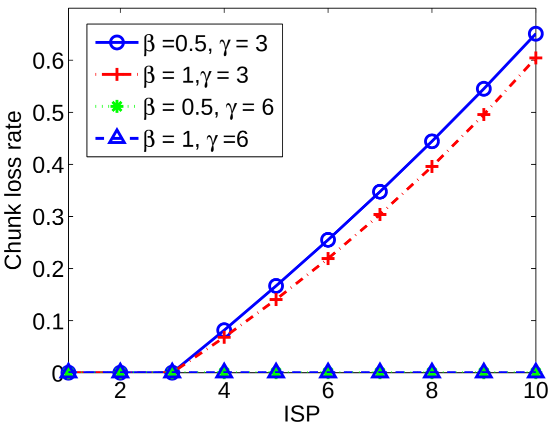

Figure 7: Chunk loss rate under ISP-aware chunk request routing algorithm.

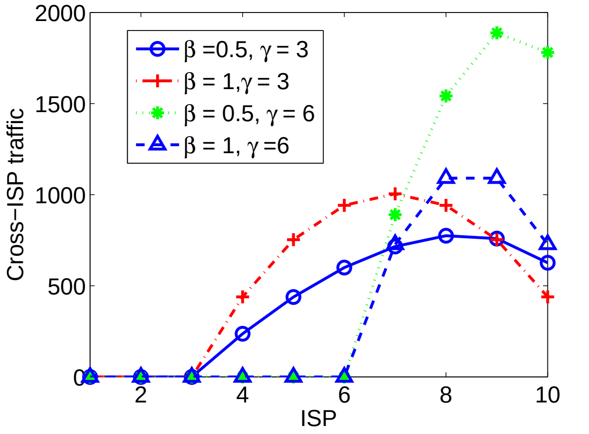

Figure 8: cross-ISP traffic under ISP-aware chunk request routing algorithm.

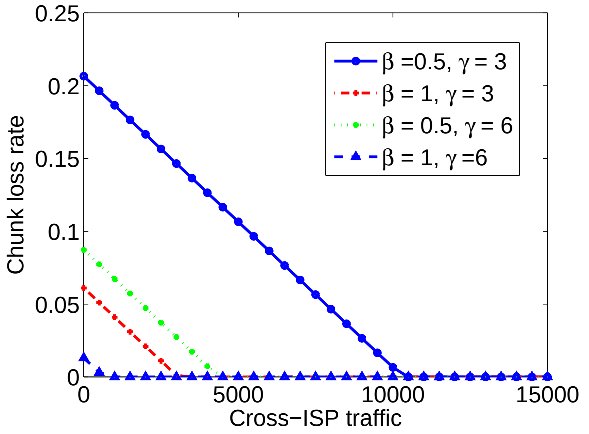

Figure 9: Relationship between chunk loss rate and cross-ISP traffic.

quest routing algorithm. The cross-ISP traffic for ISP m is the traffic flowing into ISP m. They show that all ISPs reach 0 chunk loss rate when the total peers upload bandwidth in the system is enough to support video playback rate (Scenario 3 & Scenario 4). When the total peers upload bandwidth in the system is not enough to support video playback rate (Scenario 1 & Scenario 2), those ISPs with insufficient upload bandwidth suffer chunk loss. In all four scenarios, ISPs with insufficient upload bandwidth need to download contents from ISPs with sufficient upload bandwidth. The cross-ISP traffic is related to both peer number and peers' upload bandwidth, hence the traffic volume is not monotonically increasing.

Fig. 9 plots the change of the system chunk loss rate with the total cross-ISP traffic of all ISPs. The level of system chunk loss rate is mainly determined by peers' total upload bandwidth in the system. Scenario 1 & 2 have a much higher chunk loss rate than Scenario 3 & 4 as the peers' total upload bandwidth in Scenario 1 & 2 is not sufficient to support the playback rate. In this situation, the content providers should reduce the video streaming rate. In a specific scenario, the chunk loss rate is affected by the cross-ISP traffic. The VoD system achieves a minimum chunk loss rate under an appropriate cross-ISP traffic.

7. CONCLUSIONS

This paper targets theoretical study of relationship between cross-ISP traffic and minimum system chunk request loss rate in ISP-aware P2P VoD systems. We apply the stochastic loss network model to analyze the chunk request loss problem, map the solutions to the corresponding maximum bipartite flow. We propose both the optimal cache condition and the optimal chunk request routing to achieve the minimum system chunk request loss rate.

8. REFERENCES

[1] C. S. Inc., "Cisco Visual Networking Index: Forecast and Methodology, 2010-2015, White Paper," June 2011.

[2] *PPLive*, http://www.pplive.tv.

[3] *UUSee*, http://www.uusee.com.

[4] H. Y. Xie, Y. R. Yang, A. Krishnamurthy, Y. B. G. Liu, and A. Silberschatz, "P4P: Provider Portal for Applications," in *Proc. of ACM SIGCOMM*, August 2008.

[5] W. Huang, C. Wu, and F. C. M. Lau, "The Performance and Locality Tradeoff in BitTorrent-like P2P File-sharing Systems," in *Proc. of IEEE ICC*, May 2010.

[6] F. Picconi and L. Massoulie, "ISP Friend or Foe? Making P2P Live Streaming ISP-Aware," in *Proc. of IEEE ICDCS*, June 2009.

[7] J. Wang, C. Huang, and J. Li, "On ISP-friendly Rate Allocation for Peer-assisted VoD," in *Proc. of ACM Multimedia 2008*, 2008.

[8] Y. Feng, B. C. Li, and B. Liam, "Peer-assisted VoD Prefetching in Double Auction Markets," in *Proc. of ICNP*, October 2010.

[9] D. P. Heyman and M. J. Sobel, *Stochastic Models in Operations Research: Stochastic Processes and Operating Characteristics*. Courier Dover, 2004.

[10] F. Kelly, "Loss networks," *The Annals of Applied Probability*, vol. 1, no. 3, pp. 319–378, Aug. 1991.

[11] B. R. Tan and L. Massoulié, "Optimal Content Placement for Peer-to-Peer Video-on-Demand Systems," in *Proc. of IEEE INFOCOM*, April 2011.

[12] K. Jung, Y. D. Lu, D. Shah, M. Sharma, and M. S. Squillante, "Revisiting Stochastic Loss Networks: Structures and Algorithms," in *Proc. of SIGMETRICS*, June 2008.

[13] L. R. J. Ford and D. R. Fulkerson, "Maximal Flow through a Network," *Canadian Journal of Mathematics*, pp. 399–404, 1956.

[14] J. H. Wu and B. C. Liam, "Keep Cache Replacement Simple in Peer-Assisted VoD Systems," in *Proc. of INFOCOM*, April 2009.

[15] J. Dai, B. Li, F. M. Liu, B. C. Li, and H. Jin, "On the Efficiency of Collaborative Caching in ISP-aware P2P Networks," in *Proc. of IEEE INFOCOM*, April 2011.

[16] Y. Huang, T. Z. J. Fu, D. M. Chiu, J. C. S. Lui, and C. Huang, "Challenges, Design and Analysis of a Large-Scale P2P-VoD System," in *Proc. of SIGCOMM*, August 2008.