Stochastic Model for ISP-aware VoD Streaming

***Abstract*— The abstract goes here.**

I. Introduction

II. Model and Notation

We first introduce the VoD system model.

We consider a VoD system involving $M = |\mathcal{M}|$ ISPs. ISP i has totally $N_i = |\mathcal{P}_i|$ participating peers, with each peer either ON (*i.e.*, present in the system) or OFF (*i.e.*, logged off) at a time. The average peer upload bandwidth in ISP i is U_i. A VoD system supplies multiple video channels. We do not focus on the notion of videos or channels, instead focus on a collection of $J = |\mathcal{C}|$ chunks, $\mathcal{C} = \{c_1, c_2, ..., c_J\}$, regardless of which video they belong to. A peer $p \in \mathcal{P}_i, i \in \mathcal{M}$ can cache and serve chunks from different videos. Every peer has a cache size B. Denote $\mathcal{B} = \{c^{(1)}, c^{(2)}, ..., c^{(B)}\}$ as the cache state of a peer. We model the servers as special peers that have cached all chunks. Denote S_i as the server capacity in ISP i.

A. Chunk Request

In the VoD system, we model peers in different states according to the chunk they select to download. A peer that selects chunk $j, c_j \in \mathcal{C}$ to download is at state j. The OFF peer is at state 0. The user behavior in the VoD system can be modeled as state transitions. It can be described by a transition matrix. In the matrix, element q_{jk} is the probability that a peer selects chunk k to download when finishing downloading chunk j. The stationary state distribution can be derived based on the transition matrix. Denote $(\pi_0, \pi_1, \pi_2, ..., \pi_J)$ as the stationary state distribution. There are $m_{i,j} = N_i \cdot \pi_j$ peers at state j.

We assume one unit time is needed for peers to finish downloading one chunk. A request for a chunk is resulted when a peer selects to download the chunk. The number of requests for chunk j in ISP i in one unit time is a random variable of binomial distribution, $P(Req = k) = C_{N_i}^k \pi_j^k (1-\pi_j)^{N_i - k}$. For large N_i, the binomial distribution can be approximated by the Poisson distribution. The probability that peers in ISP i send k requests for chunk j in one unit time is:

$$P(Req = k) = \frac{\lambda_{i,j}^k}{k!} e^{-\lambda_{i,j}},$$
$$\lambda_{i,j} = N_i \cdot \pi_j.$$

Request rate for chunk j is $\lambda_{i,j}$.

B. Chunk Distribution in Peers' Cache

In the VoD system, a peer contributes some size of storage to cache chunks. The state of a peer's cache is $\mathcal{B} = \{c^{(1)}, c^{(2)}, ..., c^{(B)}\}$. If not consider the sequence of the cached contents, there are $K = C_J^B$ states. If we consider the sequence of the cached contents, there are $K = B! \cdot C_J^B$ states. The stationary distribution of cache states under a specific cache replacement strategy is $\gamma_{\mathcal{B}}$, for state$\mathcal{B}$. A simple replacement strategy is Least Recently Used (LRU). In VoD systems adopting LRU, at each time slot, a peer's cache is kept as a sequence of chunks arranged in order of latest access time. Denote $(c_t^{(1)}, c_t^{(2)}, ..., c_t^{(B)})$ as the state of a peer's cache at time slot t. The LRU caching model is an irreducible and aperiodic Markov Chain. The stationary probability of cache states exists.

C. Traffic Locality Model

The request rate for chunk j in ISP i is $\lambda_{i,j}$. The probability that a peer's request is not served by its local cache is D_j for chunk j. Denote a_{ii} as the fraction of requests asking for service from peers in the same ISP, ISP i. Denote a_{mi} as the fraction of requests routed from ISP m to ISP i. The request rate for chunk j in ISP i is:

$$\nu_{i,j} = \sum_{m=1}^{M} a_{mi} \cdot D_j \cdot \lambda_{m,j}$$

D. Performance Metrics

The VoD streaming is a delay-sensitive application. The requests that can not be served by peers are redirected to servers. When server capacities are not enough to support the requests, some requests are dropped, which results in loss rate for chunks.

We define the loss rate $L_{i,j}$ for requests of chunk j in ISP i when no server capacity is used. The total needed server capacity in ISP i is:

$$S_i = \sum_{j=1}^{J} L_{i,j} \cdot \nu_{i,j}$$

III. Analysis

A. loss network analysis

In the VoD system, the chunk requests are first submitted to the P2P system. If they are accepted, upload bandwidth is used to serve them at the video streaming rate. They are rejected if their acceptance would require disruption of an ongoing request service. Rejected requests are redirected to servers. This model ensures zero waiting time for requests, which is desirable for VoD application. The system can be analyzed using loss network model.

Let $\mathbf{n_i} = \{n_{i,j}\}_{c_j \in \mathcal{C}}$ be the vector of request numbers being served concurrently. We define d_{jp} as number of concurrent downloads of chunk c_j from peer p. The following two constraints should be satisfied in ISP i:

$$\sum_{p:p\in\mathcal{P}_i,c_j\in\mathcal{B}_p} d_{jp} = n_{i,j}, \forall c_j \in \mathcal{C}$$
$$\sum_{j:c_j\in\mathcal{B}_p} d_{jp} \leq U_i, \forall p \in \mathcal{P}_i$$

$\mathcal{B}_p$ is the cache state of peer p.
A compact characterization of constraints is:

$$\forall\mathcal{A}\subseteq\mathcal{C}, \sum_{c_j\in\mathcal{A}} n_{i,j} \leq U_i\cdot|p\in\mathcal{P}_i : \mathcal{A}\cap\mathcal{B}_p\neq\emptyset|$$

As the peer number N_i grows large, by the law of large numbers, the above constraint becomes:

$$\forall\mathcal{A}\subseteq\mathcal{C}, \sum_{c_j\in\mathcal{A}} n_{i,j} \leq U_i\cdot N_i\cdot\sum_{\mathcal{B}:\mathcal{B}\cap\mathcal{A}\neq\emptyset}\gamma_{\mathcal{B}}$$

The vector $\mathbf{n_i}$ of requests number under service is a particular instance of a general stochastic process known as a loss network model. $\mathbf{n_i}$ has a product-form stationary distribution:

$$\pi(\mathbf{n_i}) = \frac{1}{G}\prod_{c_j\in C}\frac{\nu_{i,j}^{n_{i,j}}}{n_{i,j}!}$$

where G is the normalizing constant.

The fraction of dropped request numbers for chunk j in ISP i is $L_{i,j}$. The requests under service experience a delay of one unit time (service time). The loss requests experience a delay of 0. The average delay experienced by chunk requests is:

$$D_{i,j} = (1-L_{i,j})\cdot 1 + L_{i,j}\cdot 0 = (1-L_{i,j})$$

Upon applying Little's law to the system (with respect to chunk j), we obtain

$$\nu_{i,j}D_{i,j} = \mathbf{E}[n_{i,j}]$$

which yields

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

Due to the computational complexity of the exact stationary distribution, we take the 1-point approximate algorithm for computing the stationary loss probabilities. In the 1-point approximation, use $n^*_{i,j}$, which is the element of $\mathbf{n^*_i}$, the state having the maximum probability as a surrogate of $\mathbf{E}[n_{i,j}]$. $\mathbf{n^*_i}$ is the solution of the optimization problem:

$$\text{maximize}\sum_{j=1}^{J} n_{i,j}log\nu_{i,j} - logn_{i,j}!$$
$$\text{over}\forall\mathcal{A}\subseteq\mathcal{C}, \sum_{c_j\in\mathcal{A}} n_{i,j} \leq U_i\cdot|p\in\mathcal{P}_i : \mathcal{A}\cap\mathcal{B}_p\neq\emptyset|$$
$$\mathbf{n_i}\geq 0$$

We replace $logn_{i,j}!$ by $n_{i,j}logn_{i,j} - n_{i,j}$ according to Stirling's formula, $logn_{i,j}! = n_{i,j}logn_{i,j} - n_{i,j} + O(logn_{i,j})$. Relax integer vector $\mathbf{n_i}$ using a real vector $\mathbf{x_i}$.

$$\text{maximize}\sum_{j=1}^{J} x_{i,j}log\nu_{i,j} - x_{i,j}logx_{i,j} + x_{i,j}$$
$$\text{over}\forall\mathcal{A}\subseteq\mathcal{C}, \sum_{c_j\in\mathcal{A}} x_{i,j} \leq U_i\cdot|p\in\mathcal{P}_i : \mathcal{A}\cap\mathcal{B}_p\neq\emptyset|$$
$$\mathbf{x_i}\geq 0$$

The corresponding Lagrangian is:

$$\begin{aligned}
L(\mathbf{x_i},\epsilon) &= \sum_{j=1}^{J}(x_{i,j}log\nu_{i,j} - x_{i,j}logx_{i,j} + x_{i,j}) \\
&+ \sum_{\mathcal{A}\subseteq\mathcal{C}}\epsilon_{\mathcal{A}}\cdot(N_iU_i\sum_{\mathcal{B}:\mathcal{B}\cap\mathcal{A}\neq\emptyset}\gamma_{\mathcal{B}} - \sum_{j=1}^{M}x_{i,j}) \\
&= \sum_{j=1}^{J}x_{i,j} + \sum_{j=1}^{J}x_{i,j}(log\nu_{i,j} - logx_{i,j} - \sum_{c_j\in\mathcal{A},\mathcal{A}\subseteq\mathcal{C}}\epsilon_{\mathcal{A}}) \\
&+ \sum_{\mathcal{A}\subseteq\mathcal{C}}\epsilon_{\mathcal{A}}\cdot N_i\cdot U_i\cdot\sum_{\mathcal{B}:\mathcal{B}\cap\mathcal{A}\neq\emptyset}\gamma_{\mathcal{B}}
\end{aligned}$$

The KKT conditions for this convex optimization problem are:

$$\forall\mathcal{A}\subseteq\mathcal{C}, \sum_{c_j\in\mathcal{A}} x_{i,j} \leq N_i\cdot U_i\cdot\sum_{\mathcal{B}:\mathcal{B}\cap\mathcal{A}\neq\emptyset}\gamma_{\mathcal{B}}$$
$$\epsilon_{\mathcal{A}}\geq 0$$
$$\forall\mathcal{A}\subseteq\mathcal{C}, \epsilon_{\mathcal{A}}\cdot(N_i\cdot U_i\cdot\sum_{\mathcal{B}:\mathcal{B}\cap\mathcal{A}\neq\emptyset}\gamma_{\mathcal{B}} - \sum_{c_j\in\mathcal{A}}x_{i,j}) = 0$$
$$x_{i,j} = \nu_{i,j}\cdot\exp(-\sum_{c_j\in\mathcal{A},\mathcal{A}\subseteq\mathcal{C}}\epsilon_{\mathcal{A}})$$

IV. Work flow

1. When the number of peers in ISP i, N_i, is large, $N_i \gg C_J^B$, C_J^B is the number of the cache states if not consider the sequence of the cache content, there are $\gamma_{\mathcal{B}}\cdot N_i$ peers in state $\mathcal{B}$. When $\gamma_{\mathcal{B}}$ are fixed, what is the maximum supporting request rate for different chunks? What is the total supporting request rate? A simple case is when $\gamma_{\mathcal{B}}$ is the same for different cache states, what is the maximum supporting request rate for different chunks?

2. When the number of peers is few, the cache of all peers in the same ISP is limited. Under this case, we can not use $\gamma_{\mathcal{B}}\cdot N_i$ to calculate the number of peers in state $\mathcal{B}$. How should we calculate the maximum supporting request rate for different chunks?

3. Given the chunk popularity in one ISP, what is the best cache distribution that makes the cross-ISP traffic minimum without influencing the performance? How much cross-ISP traffic is necessary? Through the results, we can see under which situations the cross-ISP can be limited, and under which cases the cross-ISP can not be limited too much.

4. When the chunk popularity in different ISPs are the same, what's the best cache replacement strategy? how much cross-ISP traffic is necessary?

5. When the chunk popularity in different ISPs are different, what is the best cache replacement strategy? how much cross-ISP traffic is necessary?

6. Poisson limit for chunk requests

7. The reduced load explanation for the primal-dual optimization problem.

8. matching over bipartite graph for KKT conditions.