Weekly Report

Jian Zhao

May 10, 2010

Locality of P2P live streaming

In pull-based live streaming system, one peer exchanges its bitmap with its partners periodically. After getting its partners' bitmap, it will be able to send requests for the chunks that its partners have but it doesn't have. The priority for requesting different chunks are determined by chunk scheduling strategy, which may be different for different live streaming systems. When a peer receives the requests for chunk transmissions from its partners, it will assign its upload capacity among its partners according to some peer selection strategy.

One simple way to improve the locality of P2P live streaming is biased peer selection. In biased peer selection, one peer only has a fixed number of partners in other ISPs. And most of the partners are from the same ISP.

We first analyze the inter-ISP traffic of P2P live streaming under random peer selection strategy and latest useful chunk selection strategy. Then, we consider the inter-ISP traffic under some locality mechanism: biased peer selection. Compare the performance and inter-ISP traffic without locality mechanism and with locality mechanism. We could get the insight of the trade off between the performance and inter-ISP traffic.

We analyze three cases: Homogeneous case, Heterogeneous case, peer churn case.

Homogeneous case

In homogeneous case, the environment of P2P live streaming considered is: N_1 peers in ISP_1,N_2 peers in other ISPs, which are outside ISP_1; the upload capacity is assumed homogeneous, suppose the upload capacity u_p is equal to the playback rate r. The buffer has N units. The probability of buffer unit i is filled by a chunk is $p_1(i)$ for peers in ISP_1 and $p_2(i)$ for peers in other ISPs. One chunk will be played in one time slot. The chunk in buffer unit N is playing. And due to the upload capacity, no more than one chunk will be uploaded by a peer. The server's capacity in ISP_1 is $u_(s1)$. Each peer has n partners chosen from their peer list.

First, we calculate the probability that the partners of one peer has chunk i under different situations.

1)calculate the probability that the partners of one peer in ISP_1 has chunk i when a peer just builds intra-ISP connections:

$$Pr[H_p(i)] = 1 - [1 - p_1(i)]^n$$

Here, a question is that how to determine the appropriate number of partners. (?)

2)calculate the probability that the partners of one peer has chunk i when a peer randomly selects neighbors from ISP_1 and other ISPs.

$$Pr[H_p(i)] = 1 - [1 - p_0(i)]^n$$
$$p_0(1) = \frac{N_1 p_1 + N_2 p_2}{N_1 + N_2}$$

$p_0(i)$ is the probability that a peer's buffer unit i is filled by a chunk under the random peer selection. The initial value $p_0(1)$ equals to the mean value of ISP_1 and other ISPs.

3)calculate the probability that the partners of one peer has chunk i under biased peer selection. Suppose every peer has n_i intra-ISP partners. The inter-ISP partners is then $n - n_i$. For different ISPs, the probability is different. For peers in ISP_1, the probability is:

$$Pr[H_p(i)] = 1 - [1 - p_1^{'}(i)]^{n_i}[1 - p_2^{'}(i)]^{(n-n_i)}$$
$$p_1^{'}(1) = p_1(1); p_2^{'}(1) = p_2(1);$$

Next, The relationship between $p_1(i+1)$ and $p_1(i)$ is given as follows:

$$p_1(i+1) = p_1(i) + q(i)$$

The $q(i)$ is the probability that one peer gets the chunk to fill buffer unit i+1 in time slot i from partners, $p_1(i)$ is the probability that the peer's buffer unit i+1 is filled by the chunk in the buffer unit i after one time slot.

The calculation of $q(i)$ is:

$$q(i) = Pr[H_p(i)] \times Pr[W(i)] \times Pr[S(i)|H_p(i), W(i)]$$

$Pr[W(i)]$ is the probability that one peer doesn't have chunk i:$Pr[W(i)] = 1 - p_1(i)$. $Pr[S(i)|H_p(i), W(i)]$ is the probability that one peer selects chunk i to download when this peer doesn't have chunk i and its partners have chunk i. This probability is related to the chunk selection strategy. Here we refer to the rarest first strategy. And for the rarest first strategy, $Pr[S(i)|H_p(i), W(i)] = 1 - p_1(i)$.

So,

$$p_1(i+1) = p_1(i) + q(i) = p_1(i+1) = p_1(i) + [1 - p_1(i)]^2 \times Pr[H_p(i)]$$

So, substitute the $Pr[H_p(i)]$ of three different cases into the above recursive equation, we can get:

(1) the case that a peer just builds intra-ISP connections:

$$p_1(i+1) = p_1(i) + q(i) = p_1(i) + [1 - p_1(i)]^2 \times [1 - [1 - p_1(i)]^n]$$

(2) the case that a peer selects partners randomly from ISP_1 and other ISPs:

$$p_0(i+1) = p_0(i) + q(i) = p_0(i) + [1 - p_0(i)]^2 \times [1 - [1 - p_0(i)]^n]$$
$$p_0(1) = \frac{N_1 p_1 + N_2 p_2}{N_1 + N_2}$$

(3) the case of biased peer selection:

$$p_1^{'}(i+1) = p_1^{'}(i) + q(i) = p_1^{'}(i) + [1 - p_1^{'}(i)]^2 \times [1 - [1 - p_1^{'}(i)]^{n_i}[1 - p_2^{'}(i)]^{(n-n_i)}]$$
$$p_1^{'}(1) = p_1(1); p_2^{'}(1) = p_2(1);$$

Next step, it is necessary to analyze the influence of $Pr[H_p(i)]$ on $p_1(N)$ and inter-ISP traffic.

Analyze inter-ISP traffic: With the same server deployments and number of peers, how to calculate the inter-ISP traffic:

(1) the case that a peer just builds intra-ISP connections:
inter-ISP traffic is 0.

(2) the case that a peer selects partners randomly from ISP_1 and other ISPs:

(3) the case of biased peer selection:
inter-ISP traffic is $\sum_{i=1}^{n-1}[1-p_1^{'}(i)]^2 \times [1-p_1^{'}(i)]^{n_i}[1-[1-p_2^{'}(i)]^{n-n_i}]$
The impact of server deployments and number of peers:

Heterogeneous Case

In the heterogeneous case(in terms of peers' bandwidth), the peers' bandwidth are different. Assume there are h classes of peers. The upload capacity of each class is U^i, and the download capacity of each class is D^i, $U^1 < U^2 < U^3 < ... < U^r < ... < U^h$, $D^1 < D^2 < D^3 < ... < D^r < ... < D^h$. Let p^i denote the percentage of class i peers in the streaming system. We model the live streaming system as a discrete time system. At the beginning of one time slot, peers exchange bitmap with partners. Assume in one time slot class i peer(upload capacity U^i) could only transmit i chunks. Denote the playback rate of the streaming video by r chunks per time slot. The downloading rate of peers is around r chunks per time slot. And the upload capacity are the bottlenecks. So, we assume the uploading rates of peers are $u^i = U^i$ for class i peers.

We assume that every peer exchanges its bitmap with C partners every time slot. Then, the peer requests for the missing chunks in its buffer from its partners that have the chunks. After receiving the requests, the partners schedule the sending of chunks. One partner may receive requests for chunk transmissions from many peers. With the limit of upload capacity, we assume that in one time slot, the partner randomly selects one peer to transmit chunks. So, the probability for one partner sending chunks to a peer according to the requests will be $1/C$.

The upload capacity of the server is u^s, the new content rate pumped into the system by the server equals to the playback rate r, and the upload capacity of the server could support N^s peers simultaneously.

Now let's derive the probability that the buffer is filled in the heterogeneous case.

First, consider one connection between a peer with one of its partners. The probability that buffer unit j+r of the peer is filled is

$$P(j+r) = P(j) + Q(j)$$

Here P(j) is the probability that buffer unit j of the peer is filled. Q(j) is the probability that buffer unit j+r get the chunk to fill it up from other peers. The next step is to derive Q(j), using P(i) to express Q(j), then, we could get a recursive equation for P(j).

When one peer connects to a partner of class i, the probability that the peer could get chunk j is:

$$q_i(j) = [1-p(j)]p(j)s_i(j)$$

$s_i(j)$ is the probability that the partner of class i transmits chunk j to the peer. And we could derive that:

$$if j < i s_i(j) = 1 - p(1);$$
$$if j >= i s_i(j) = 1 - p(j-i+1);$$

So, for one connection, the probability that one peer gets chunk j is q(j), $q(j) = (1-\frac{N^s}{N})[p^1 q_1(j)+p^2 q_2(j)+p^3 q_3(j)+...+p^h q_h(j)]$. Here, $1-\frac{N^s}{N}$ means the probability that the connection isn't between the peer with the server. When the peer connects to the server, the server will send the new contents to the peer.

$$Q(j) = 1 - [1 - q(j)]^C$$

We assume the length of the buffer is nr. $P(r) = P(r-1) = ... = P(2) = P(1) = \frac{N^s}{N}$,

$$P[(i+1)r + k] = P[ir + k] + 1 - [1 - q(ir + k)]^C;$$

$$1 \le k \le r;$$

peer churn case

To be edited.