

# Weekly Report

Jian Zhao

May 2, 2010

## Locality of P2P live streaming

### Heterogeneous Case

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.

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  $D^i$ ,  $U^1 < U^2 < U^3 < \dots < U^r < \dots < U^h$ ,  $D^1 < D^2 < D^3 < \dots < D^r < \dots < D^h$ . 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:

$$\begin{aligned} if j < i s_i(j) &= 1 - p(1); \\ if j \geq i s_i(j) &= 1 - p(j - i + 1); \end{aligned}$$

So, for one connection, the probability that one peer gets chunk  $j$  is  $q(j)$ ,  $q(j) = (1 - \frac{N^s}{N})[p^1 q_1(j) + p^2 q_2(j) + p^3 q_3(j) + \dots + p^h q_h(j)]$ . 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  $nr$ .  $P(r) = P(r - 1) = \dots = P(2) = P(1) = \frac{N^s}{N}$ ,

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

$$1 \leq k \leq r;$$

To be edited.