1 Modeling of the P2P service migration problem

We suppose there are M videos, and N ISPs. There are one on-premise server and one cloud node in each ISP.

1.1 Optimization of the problem with Lyapunov optimization

This is a combination of optimization for one time deployment and time-average variables. The placement of content is one time deployment while the schedule is for time-average.

Notation definition:

B_s: storage capacity of the on-premise server

B_u: upload bandwidth capacity of the on-premise server

h_j: charging rate for storage on the cloud at the $j-th$ ISP

k_j: charging rate for upload bandwidth on the cloud at the $j-th$ ISP

s_m: storage of $m-th$ video

$x_m^j = \{0, 1\}, m = 1, ..., M$: $x_m^j = 1$ if the placement of the $m-th$ video is on the on-premise server at the $j-th$ ISP; $x_m^j = 0$ otherwise;

$y_m^j = \{0, 1\}, m = 1, ..., M$: $y_m^j = 1$ if the placement of the $m-th$ video is on the cloud at the $j-th$ ISP; $y_m^j = 0$ otherwise;

D_s^{ji} is the delay from source j to on premise server i, and D_c^{ji} is the delay from source j to on cloud node i.

$A_m^j(t)$: at time slot t, number of requests of the $m-th$ video generated from the $j-th$ ISP.

$r_m^j(t)$: at time slot t, number of requests of the $m-th$ video that are admitted into the system. $r_m^j(t) \leq A_m^j(t)$

$S_m^j(t)$: at time slot t, number of requests for video m that are routed from region j to on-premise server i

$C_m^{ji}(t)$: at time slot t, number of requests for video m that are routed from region j to cloud node i

$Q_m^j(t)$: at time slot t, queues of requests from video m from ISP j.

Note: The queue update is: $Q_m^j(t+1) = \max[Q_m^j(t) + r_m^j(t) - S_m^j(t) - \sum_{i=1}^{N} C_m^{ji}(t), 0]$

Different from the previous sub section, $S_m^j(t)$ and $C_m^{ji}(t)$ is not a schedule of fraction of arrival rates for all time slots. Now they are schedule of number of requests (integers) for each time slot.

Note: minimize sum of:

- time average spending cost of upload bandwidth at cloud node
- spending cost of time average upload bandwidth at on premise server
- cost of storage at cloud
- cost of storage at on premise server
- time average weighted delay

minimize $\overline{\sum_{m=1}^{M}\sum_{j=1}^{N}\sum_{i=1}^{N}(s_m C_m^{ji}(t)k_i)} + \alpha\sum_{m=1}^{M}\sum_{j=1}^{N}\sum_{i=1}^{N}\overline{s_m S_m^j(t)} + \beta\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m y_m^j h_j) + \gamma\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m x_m^j) - \rho\sum_{j=1}^{N}\sum_{i=1}^{N}\overline{\sum_{m=1}^{M} s_m(C_m^{ji}(t)D_c^{ji} + S_m^{ji}(t)D_s^{ji})}$

subject to:

$y_m^j = \{0,1\}, \forall j = 1, ..., N, \forall m = 1, ...M$

$x_m^j = \{0,1\}, \forall j = 1, ..., N, \forall m = 1, ...M$

$0 \leq S_m^j(t) \leq S_m^j(t)x_m^j, \forall j = 1, .., N, \forall i = 1, ..., N, \forall m = 1, ..., N, \forall t$

(instead, we can assume that on premise server keeps all videos)

$0 \leq C_m^{ji}(t) \leq C_m^{ji}(t)y_m^t, \forall j = 1, .., N, \forall i = 1, ..., N, \forall m = 1, ..., N, \forall t$

$\sum_{m=1}^{M} s_m x_m^j \leq B_s, \forall j$ (on-premise server's storage constraint)

$\sum_{m=1}^{M}\sum_{j=1}^{N} s_m S_m^j(t) \leq B_u, \forall i = 1, ..., N, \forall t$ (on-premise server's upload bandwidth constraint)

Queues $Q_m^j(t)$ is stable, $\forall m, j$, i.e., $\overline{r_m^j(t)} \leq \overline{\sum_{i=1}^{N} S_m^j + \sum_{i=1}^{N} C_m^{ji}}$

$Q_m^j(0) = 0, \forall m, j$

Note:

known values: B_s, B_u, h_j, k_j, s_m, $r_m^j(t)$, D_c^{ji}, D_s^{ji}

optimization variables: x_m^j, y_m^j, $S_m^j(t)$, $C_m^{ji}(t)$

$\overline{\sum_{m=1}^{M}\sum_{j=1}^{N}\sum_{i=1}^{N}(s_m C_m^{ji}(t)k_i)} + \alpha\sum_{m=1}^{M}\sum_{j=1}^{N}\overline{s_m S_m^j(t)} + \beta\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m y_m^j h_j) + \gamma\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m x_m^j) - \rho\sum_{j=1}^{N}\sum_{i=1}^{N}\overline{\sum_{m=1}^{M} s_m(C_m^{ji}(t)D_c^{ji} + S_m^{ji}(t)D_s^{ji})}$
$= \sum_{m,j,i} s_m\overline{C_m^{ji}}k_i + \sum_{m,j}\alpha s_m\overline{S_m^j} - \sum_{m,j,i}\rho s_m\overline{C_m^{ji}}D_c^{ji} - \sum_{m,j}\rho s_m\overline{S_m^j}D_s^{ji} + \beta h\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m y_m^j) + \gamma\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m x_m^j) = \sum_{m,j,i} s_m\overline{C_m^{ji}}k_i + \sum_{m,j}\alpha s_m\overline{S_m^j} - \sum_{m,j,i}\rho s_m\overline{C_m^{ji}}D_c^{ji} - \sum_{m,j}\rho s_m\overline{S_m^j}D_s^{ji} + \beta h\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m y_m^j) + \gamma\sum_{m=1}^{M}\sum_{j=1}^{N}(s_m x_m^j)$

To tackle the admission problem, we need a virtual queue: $Y_m^j(t) = max[Y_m^j(t) + A_m^j(t) - r_m^j(t), 0]$

$\Delta(Q(t)) + Vcost$
$\leq B + \sum_{m,j} Q_m^j(t)(r_m^j(t) - S_m^j(t) - \sum_{i=1}^{N} C_m^{ji}(t)) + \sum_{m,j} Y_m^j(A_m^j(t) - r_m^j(t)) + V(\sum_{m,j,i} s_m C_m^{ji}(t)k_i + \sum_{m,j}\alpha s_m S_m^j(t) - \sum_{m,j,i}\rho s_m C_m^{ji}(t)D_c^{ji} - \sum_{m,j}\rho s_m S_m^j(t)D_s^j)$
$= B - \sum_{m,j,i} C_m^{ji}(t)(Q_m^j(t) - Vs_m k_i + V\rho s_m D_c^{ji}) - \sum_{m,j} S_m^j(t)(Q_m^j(t) - V\alpha s_m + V\rho s_m D_s^j) - \sum_{m,j} r_m^j(t)(Y_m^j(t) - Q_m^j(t)) + \sum_{m,j} Y_m^j(t)A_m^j(t)$

2 Reading note for the paper "Content-aware caching and traffic management in content distribution networks"

model:
1. the constraint is the link capacity between each pair of source and cache.
2. queue: source s for content c
3. in each time slot, a source can only request a type of content from a cache.
4. the schedule x * presence at the cache p =1
5. refresh baed on queue length. MWI= max-weight optimization independent of cache contents.
MWP: Max-Weigth schedule except that it must now be calculated subject to the presence of schedule content
PMW: Periodic max-Weight schduleing=MWI at refresh times, MWP at the inter-refresh time
6. "throughput optimal" is interpreted as "queue is stable whenever the arrival rate is inside capacity region"
Therefore it doesn't consider any "utility". The proof is only to prove that the queue is stable.

This paper focused at:
1. prove that the PMW schudule is throughput optimal (the queue is stable) with refresh period 1 and D

We have different concern:
1. the capacity is constrained only be the link between each source and each cache server. (like a switch)
2. we want to minimize the cache replacement, the upload bandwidth while he only wants to minimize the delay(queue length)

similarity:
1. we also need to do the cache replacement
2. there are also multiple types of content in our system
3. we also want to minimize other utility, such as delay
4. there are problems of scheduling/placement of content in migration as well.