BANDWIDTH ALLOCATION ISSUES IN NEAR VIDEO ON DEMAND SERVICES

Jehan-François Pâris[1]
Department of Computer Science
University of Houston
Houston, TX 77204-3475
paris@cs.uh.edu

Steven W. Carter Darrell D. E. Long[2]
Department of Computer Science
Baskin School of Engineering
University of California
Santa Cruz, CA 95064
{carter, darrell}@cse.ucsc.edu

1. INTRODUCTION

Video on demand (VOD) proposes to offer to its subscribers the possibility of watching the program of their choice at the time of their choice, as if they were watching a rented video cassette. Despite the obvious appeal of the concept, VOD has yet to succeed on the marketplace because it has to compete against cheaper alternatives such as pay-per-per view and video cassette rentals.

One way to reduce the cost of VOD is to schedule repeated broadcasts of the videos that are likely to be watched by many viewers rather than waiting for individual requests. This technique is known as *video broadcasting*. The savings that can be achieved are considerable, as it is often the case that 40 percent of the demand is for a small number, say, 10 to 20, of hot videos [3]. Depending on the frequency at which these videos are rebroadcast, customers may have to wait between a few minutes to, say, half an hour before watching the video of their choice. Hence this type of service can either be referred to as near video on demand (NVOD) or enhanced pay per view (EPPV).

One of the most important decisions that has to be made by the provider of a NVOD service is the time interval at which each video should be rebroadcast. Very frequent rebroadcasts will minimize the average customer waiting time but also increase cost of the service. Less frequent rebroadcasts may on the other hand result in a significant loss of revenue if too many customers decide not to wait for the next time the video will be rebroadcast.

This paper addresses the problem of selecting the optimal rebroadcasting frequency for each video. The solution we propose consists of selecting the rebroadcasting frequencies that maximize the expected number of viewers per unit of bandwidth. To evaluate this expected number of viewers, we will represent the customer demand by (a) the rate at which any video would be ordered if it was available for immediate viewing and (b) a *tolerance function* expressing the customer willingness to wait for a given amount of time before watching the video of their choice. We will present two strategies tailored to two specific tolerance functions; namely, a step function and a three-level staircase function. As it may be expected, the outcome of these strategies will strongly depend on the cost of increasing the frequency at which a given video is rebroadcast.

---

[1] Most of the work reported here was performed while this author was on sabbatical leave at the Department of Computer Science, Baskin School of Engineering, University of California, Santa Cruz.

[2] This research was supported by the Office of Naval Research under Grant N00014-92-J-1807

The remainder of this paper is organized as follows: Section 2 will introduce our model, section 3 will present our aproach and section 4 will show our conclusion.

2. OUR MODEL

We will consider an NVOD service that has a total bandwidth *S* to allocate to the videos it will broadcast on any given day. This is a realistic hypothesis because total bandwidth is a good representative of both the transmission costs and the server workload. Hence, it is also very likely to be the limiting resource for most video-on-demand servers. Given that broadcasting more frequently any given video will require more bandwidth, the number *m* of videos being broadcast will monotonic non-decreasing function of the broadcasting frequencies of these *m* videos.

Our objective function will be to maximize the total number of viewers during any given time interval. To achieve this objective, we will have to find the optimal trade-off between the *breadth* of the video selection, that is the number *m* of videos being broadcast, and the *waiting times* the customers are willing to accept.

Two major factors seem to affect the number of customers who will order a given video. These are the *popularity* of the video itself, and the *frequency* at which it is broadcast (some people may want not to watch a video if they have to wait for more than. say, fifteen or twenty, minutes). Let $\lambda_i(t)$ the rate at which video *i* will be ordered if it is broadcast at a frequency $f = \frac{1}{t}$, that is, if the maximum waiting time is equal to t. We will assume that $\lambda_i(t)$ can be decomposed into a product

$$\lambda_i(t) = f_i w(t)$$

where f_i is the rate at which video i would be ordered if it was always available for immediate viewing and $w(t)$ is a *delay tolerance function* expressing the customer willingness to wait for a given amount of time t before watching the video of their choice. The coefficient f_i represents the popularity of the video and can be conveniently approximated by a Zipf's law [6]. The function $w(t)$ is a monotonic non-increasing function of t with $w(0) = 1$. It is unfortunately much more difficult to estimate.

One last factor we need to consider is the bandwidth required for broadcasting a given video at frequency $f = \frac{1}{t}$. The simplest broadcasting protocol is *staggered broadcasting* [7]. A video broadcast under that protocol is continuously retransmitted over *k* distinct streams at equal time intervals. The approach does not necessitate any significant modification to the set-top box but requires a large number of streams per video to achieve a reasonable waiting time. If *D* is the duration of the video and *b* the consumption rate of the video, the total bandwidth required to achieve a maximum waiting time of *t* time units is given by

$$B_{SB}(t) = \frac{bD}{t},$$

which can be rewritten as $B_{SB}(t) = bfD$. In other words, the total bandwidth required for broadcasting a video at a frequency f is directly proportional to f.

The last two years have seen the development of several new broadcasting protocols, among which are Viswanathan and Imielinski's *pyramid broadcasting protocol* [7], Aggarwal, Wolf and Yu's permutation-based pyramid broadcasting protocol [1], Hua and Sheu's *skyscraper broadcasting protocol* [3], Juhn and Tseng's *harmonic broadcasting protocol* [4] and its variants [5]. These protocols share the common objective of reducing the total bandwidth required to achieve a given maximum waiting time. The results obtained so far have been impressive: some recent broadcasting protocols, such as harmonic broadcasting and its variants, require slightly less than four times the video consumption rate to provide a maximum waiting time of five minutes for a two-hour video. Thus we would only need the equivalent of eighty conventional video streams to service all the customers who want to watch one of the top twenty videos. At the risk of grossly oversimplifying, we could say that harmonic broadcasting and its variants require a total bandwidth that is only proportional to the logarithm of the broadcasting frequency f.

These excellent results however come with a price: the user set-top box (STB) or set-top computer (STC) must have enough local storage to store up to 40 percent of the video. In the current state of the storage technology, this implies that the STB must have a local disk drive. Recent advances in disk technology have made this requirement much less of a problem as large capacity disk drives are much cheaper today than they were a few years ago

3. OUR APPROACH

The simplest delay tolerance function we can imagine is a step function

$$w(t) = \begin{cases} 1 & t \le t_{\max} \\ 0 & \text{otherwise} \end{cases}.$$

This delay tolerance function represents a situation where *all* viewers desirous to watch one of the videos being broadcast are willing to wait for up to $t_{\max}$ time units and *none* is willing to wait much longer.

The best bandwidth allocation for this delay tolerance function is quite simple. Let S be the total available bandwidth and $B_i(t_{\max})$ the bandwidth required to guarantee a maximum waiting time $t_{\max}$ for video i. For each video in the library, we compute the expected number of viewers per unit of time and unit of bandwidth

$$v_i = \frac{f_i}{B_i(t_{\max})}.$$

and select the top m videos in the order of their decreasing v_i's until there is not enough remaining bandwidth to add one more video.

Another, more realistic, delay tolerance function can be described by the staircase function

$$w(t) = \begin{cases} 1 & t \le t_1 \\ p & t_1 < t \le t_2 \\ 0 & t > t_2 \end{cases}.$$

This delay tolerance function represents a situation where *all* viewers desirous to watch one of the videos being broadcast are willing to wait for up to t_1 time units, a fraction *p* of them is willing to wait up to t_2 time units and *none* is willing to wait much longer. Let $B_i(t_1)$ and $B_i(t_2)$ be the bandwidths required to guarantee the respective maximum waiting time t_1 and t_2 for video *i*. The expected number of viewers per unit of time and unit of bandwidth for video *i* if it is broadcast with a maximum waiting time t_2 is given by

$$u_i = \frac{pf_i}{B_i(t_2)}.$$

The expected number of additional viewers per unit of time and unit of bandwidth if the same video is broadcast with a maximum waiting time t_1 is then given by

$$v_i = \frac{(1-p)f_i}{B_i(t_1) - B_i(t_2)}.$$

We can now construct two ordered lists of videos, the first one ordered by decreasing u_i's and the second by decreasing v_i's. We will then select the entries from both lists with the highest u_i's or v_i's until the total available bandwidth is exhausted and no entries can be selected.

The algorithm can be better understood using an example. Consider a very small NVOD service having a total available bandwidth of 40 channels. Let us assume that all videos last exactly 120 minutes and have f_i's proportional to $\frac{1}{i}$ where *i* is the rank of the video in their ordering by decreasing popularities. Let us further assume that the delay tolerance function $w(t)$ is given by

$$w(t) = \begin{cases} 1 & t \le 15\,\text{min.} \\ 0.6 & 15\,\text{min.} < t \le 30\,\text{min.} \\ 0 & t > 30\,\text{min.} \end{cases}$$

In other words, all customers are willing to wait 15 minutes, 60 percent of them are willing to wait up to 30 minutes and nobody is willing to wait more than 30 minutes. Finally, let us assume that the server uses staggered broadcasting to broadcast its videos. Since all videos have the same duration, all $B_i(t)$'s will only depend on the maximum waiting intervals *t* and we will have $B(15) = 8$ channels and $B(30) = 4$ channels. The individual u_i's and v_i's of each video are then given by Table 1. Selecting the highest u_i's and v_i's in the table above, we find that the optimal bandwidth allocation is to broadcast videos 1 to 4 every 15 minutes and videos 5 and 6 every 30 minutes.

We had implicitly assumed in this informal presentation of our algorithm that we had $u_i > v_i$ for all videos. There could however be special combinations of $w(t)$, $B_i(t_1)$ and $B_i(t_2)$ for which it might not be true. If this is the case, one should enforce the additional rule of not

Table 1: Values of f_i, u_i and v_i for the first seven videos

i	f_i	u_i	v_i
1	1.00	0.15	0.1
2	0.500	0.075	0.05
3	0.333	0.05	0.033
4	0.250	0.038	0.025
5	0.200	0.003	0.02
6	0.167	0.025	0.017
7	0.141	0.021	0.014

selecting a v_i before the corresponding u_i has been selected. Doing otherwise, would result in allocating extra bandwidth for broadcasting more frequently a video that was not yet selected to be broadcast.

4. CONCLUSIONS

One way to reduce the cost of video on demand services is to schedule repeated broadcasts of the videos that are likely to be watched by many viewers. We have addressed in this paper the problem of selecting the optimal rebroadcasting frequencies for these videos. The solution we propose consists of selecting the rebroadcasting frequencies that maximize the expected number of viewers per unit of bandwidth. To evaluate this expected number of viewers, we have introduced a *tolerance function* expressing the customer willingness to wait for a given amount of time before watching the video of their choice.

More work still lies ahead. We need in particular to learn more about actual customer behavior and their response to changes in rebroadcasting frequencies. One clear conclusion of this early study is the benefit of broadcasting more frequently a smaller set of very popular videos rather than broadcasting less frequently more videos.

REFERENCES

[1] C. C. Aggarwal, J. L. Wolf, and P. S. Yu . A permutation-based pyramid broadcasting scheme for video-on-demand systems. *Proc. International Conference on Multimedia Computing and Systems*, pages 118–26, June 1996

[2] A. Dan, D. Sitaram and P. Shahabuddin. Scheduling policies for an on-demand video server with batching. *Proc. ACM Multimedia Conference*, pp. 15–23, 1994.

[3] K. A. Hua and S. Sheu. Skyscraper broadcasting: a new broadcasting scheme for metropolitan video-on-demand systems. *Proc. ACM SIGCOMM '97*, pages 89–100, Sept. 1997.

[4] L. Juhn and L. Tseng, Harmonic broadcasting for video-on-demand service. *IEEE Transactions on Broadcasting*, 43(3):268–271, Sept. 1997.

[5] J.-F. Pâris, S. Carter and D. D. E. Long, Efficient broadcasting protocols for video on demand. *Proc. 1998 MASCOTS Conference*, pages 127–132, July 1998.

[6] A. Tanenbaum, *Computer Networks*, 3rd edition, Prentice-Hall, 1997.

[7] S. Viswanathan and T. Imielinski, Metropolitan area video-on-demand service using pyramid broadcasting. *Multimedia Systems*, 4(4):197–208, 1996.