EVALUATION OF SEGMENT-BASED PROXY CACHING FOR VIDEO ON DEMAND

Muhammad Muhammad, Wei Tu, and Eckehard Steinbach

Institute of Communication Networks, Media Technology Group
Technische Universität München, 80333 Munich, Germany
muhammad@mytum.de, {wei.tu, eckehard.steinbach}@tum.de

ABSTRACT

In this paper, we derive an analytical model for the evaluation of the performance of a Video on Demand (VoD) system. The model estimates the mean waiting time achievable by the Popularity-Aware Partial cAching (PAPA) algorithm from our previous work. Two approximation strategies are proposed for low computational complexity. Furthermore, we also consider the influence of a starting point shift on the quality of experience and combine the two factors into a universal user satisfaction metric. In order to find the relation between the two impairments, waiting time and starting point shift, sophisticated subjective tests are performed. With the final score model, a more comprehensive evaluation of the system can be obtained with very low computational complexity.

1. INTRODUCTION

Proxy servers become a widely standard deployed component to improve the quality of web browsing. Recently, they have also been introduced to VoD services, in order to decrease the load at the remote server and to minimize the client's waiting time. If, as shown in Fig. 1, the proxy and the clients are located within the same geographical area (e.g., a LAN), the Round-Trip-Time (RTT) between the clients and the proxy becomes negligible compared to that between the proxy and remote server(s) located somewhere in the Internet.

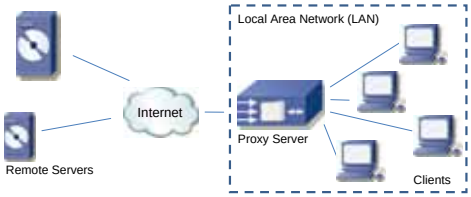


Fig. 1. Server-Proxy-Client network structure

When a client wants to play a video, a request is sent to the proxy. The latter checks if the desired content is available in the local cache and if yes, it is directly fed to the client. Otherwise, the request is forwarded to the remote server to download the missing frames. The newly downloaded frames stay for some time in the proxy cache, in case other users want to watch the same scene. The popularity of the video content determines its survival time in the cache.

Via the introduction of proxy servers, data traffic between the clients and the remote server is minimized with partial caching algorithms in [1]. Since the storage requirement of a video object is significantly larger than that for other Internet objects - such as images, HTML objects, etc... it is infeasible to store a large number of complete videos on the proxy. To increase the number of videos that can benefit from the proxy, a segment-based caching scheme is introduced in [2]. Moreover, in order to maintain the proxy cache capacity, proxy cache update algorithms are introduced in [3] and [4], to dynamically manage the cache on the proxy. Because clients usually do not necessarily start videos from the beginning, random access must be taken into account. As stated in [5], the log file of a real VoD system shows that the video contents are randomly accessed instead of being played sequentially.

One of the most important aspects that reflect the quality of a VoD service is the user perceived initial delay. In [6], a novel Segment-Prefix caching structure together with a Popularity-Aware Partial cAching (PAPA) algorithm have been proposed. The approach in [6] is based on the observation that a small shift of the starting point is more tolerable to the user than initial delay. In PAPA, the video is partitioned into segments of several Groups of Pictures (GoPs) and playout always starts from the beginning of a segment if the requested starting point is inside the segment. The partial caching strategy in PAPA keeps the beginning of popular segments in the proxy cache and hence playout can start immediately after a request. The amount of data stored for a segment is enough for the remote server to deliver the missing frames to the proxy and hence avoid playout interruption at the client. Different to the common assumption in most of the related works (e.g., in [7]), in PAPA, the popularity is considered on a smaller unit level than the whole video, while still following a Zipf distribution [8]. This is because the popularity may differ within one video. For instance, in a football match the scenes with goals are typically the ones most frequently visited.

The algorithm in [6] only considers the waiting time, while the degradation of user satisfaction by early start is not considered. In this paper, we derive an analytical model to estimate the waiting time for a large scale system with low computational complexity. Furthermore, the model for early start and user satisfaction will also be elaborated. In order to have the user satisfaction score as a function of early start and waiting time, a more sophisticated subjective test environment is also developed in this work.

In the next section we introduce the subjective tests and the obtained results. Section 3 presents the expected waiting time, the early start time and the final satisfaction score models. Some experimental results are shown in Section 4 and Section 5 concludes the paper.

2. SUBJECTIVE TESTS

We have implemented a windows media player like client using the open source VideoLAN/VLC media player for the core functions through its .net bindings. Our media player shown in Fig. 2 is called "TUMPlayer". It has the functionality of *Play*, *Pause*, *Stop*, *Fast Forward/Rewind* and controllers for *Volume*, *Thumbnail View* and *Full Screen*.

Fig. 2. Interface of TUMPlayer

When a volunteer starts the test, two images selected from the beginning of two popular scenes pop up on the left side of the player, as shown in Fig. 2. By clicking on each of these thumbnails the user will experience one of the following two modes:

1. The playout starts after an initial delay of 1, 3, 4, 5 or 7 seconds and exactly from the desired point. Moreover, a "Buffering..." text message is displayed in a panel at the bottom of the player.
2. The playout starts immediately without delay but with a start shift (early playout) of 2, 4, 6, 9 or 12 seconds from the requested scene.

The test persons are asked to wait until the scene they clicked on appears in order to make sure that the early start becomes noticeable. After viewing both scenes, the test persons are asked to randomly access the video using the slide bar to get a better feeling of the interactivity with the system. When finishing every test, a grade is to be granted reflecting their satisfaction about the system with the randomly picked system parameters.

Three videos with different characteristics: *News*, *Sport* and *Movie* are used in our tests. The *News* video is freshly captured from the CNN news TV, the *Sport* clip is the final match of the "Copa Libertadores 2007," and the film Shrek-I is used for the *Movie* category. All the videos are encoded using MPEG-1 with CIF at 25 fps. Fig. 3(a), (b), (c) show the results obtained for the *News*, *Sport* and *Movie* tests, respectively. The X-axis represents the initial waiting time or early start time in seconds while the Y-axis represents the user satisfaction on a scale from 0 to 10. The higher the score, the more satisfied the user. Each point on every curve is derived by averaging the grades from 28 test persons. The dashed curve represents the client satisfaction as a function of the initial delay, which shows that the user satisfaction declines rapidly as the waiting time increases. The dotted curve illustrates the user satisfaction with respect to early start, which shows a much slower degradation of user satisfaction. Hence we can draw the conclusion that a VoD user is more comfortable with early start than initial delay.

Since the three categories of the videos show the same trend, a universal user satisfaction metric can be realized by averaging and interpolating the results from Fig. 3(a)-(c), as shown in Fig. 3(d). It can be mathematically represented by:

$$S_{WT} = max(0, -0.653 \cdot t_{WT} + 8) \quad (1)$$

$$S_{ES} = max(0, -0.137 \cdot t_{ES} + 8), \quad (2)$$

where S_{WT} and S_{ES} denote the user satisfaction score for the waiting time and the early start, respectively. According to (1) and (2), when the waiting time or the early start equals to 0 second, a score of 8 is obtained. It does not achieve the full score of 10 because the test persons are too conservative to give the full points. Therefore, in the following, when used in the model, we assume 8 to be the highest score.

3. ANALYTICAL MODELS

As mentioned in [6], a video segment consists of several GoPs. The GoPs at the beginning of a segment represent the prefix and the remaining GoPs the suffix. A prefix frame, in case of limited cache size, is dropped from the proxy cache, by PAPA, either because of its low popularity or because of its short loading time from the remote server. On the other hand, due to the fixed Segment-Prefix structure, an early start is employed to minimize the initial waiting time. In this section, we build analytical models to estimate the early start time, the waiting time as well as the subjective score of a given system. Table1 gives a brief overview of the parameters used in the models. The superscript (v, s, g, f) denotes frame f in the g-th GOP of segment s of video v.

Table 1. Parameters for analytical models

Parameters	Description
S_V	Total size of videos in bytes
$S_F^{(v,s,g,f)}$	Size of frame (v, s, g, f) in bytes
$S_G^{(v,s,g)}$	Size of the GoP (v, s, g) in bytes
S_C	Cache size in bytes
L_P	Prefix length in number of GoPs
L_S	Segment length in number of GoPs
L_G	GoP length in number of frames
N_V	Number of videos on the service list
R^v	Link rate for video v
$P^{(v,s)}$	Popularity of segment (v, s)
r_f	Frame rate
E_{WT}	Expected Waiting Time
E_{ES}	Expected Early Start
N_S^v	Number of segments in video v
$p_s = \frac{S_C}{S_V}$	Percentage of video content cached

3.1. Early Start Model

In PAPA, as the playout always starts from the beginning of a segment, once the segment structure is defined, the mean early start time is a constant, which can be presented as

$$E_{ES} = \frac{L_G}{r_f} \cdot \frac{1}{L_S} \cdot \sum_{i=1}^{L_S-1} i = \frac{L_G \cdot (L_S - 1)}{2 \cdot r_f}. \quad (3)$$

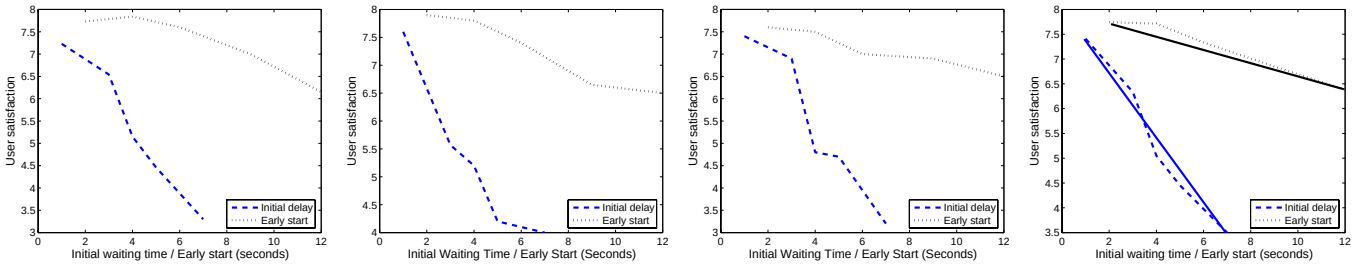
However, if the cache size is larger than the total prefix size (i.e., $S_C > \frac{L_P}{L_S} \cdot S_V$), suffix frames can also be stored which leads to a smaller early start time when one or more complete suffix GoPs after the prefix are cached. In this case, the playout can start one or more GoPs later, which reduces the early start time. We assume that at maximum the proxy cache can hold

$$S_C = \frac{L_P \cdot x}{L_S} \cdot S_V \quad (x \geq 1), \quad (4)$$

where $L_P \cdot x$ is the cached part in the segment, and x can be represented with:

$$x = \frac{p_s \cdot L_S}{L_P}, \quad (5)$$

The number of fully cached suffix GoPs y can be determined by



(a) News (b) Sport (c) Movie (d) Universal model

Fig. 3. Results of subjective tests

$$\begin{aligned} y &= max(\lfloor L_P \cdot (x-1) \rfloor, 0) \\ &= max(\lfloor p_s \cdot L_S - L_P \rfloor, 0). \end{aligned} \quad (6)$$

For example, if we have a segment consisting of 2 prefix GOPs and 3 suffix GOPs, then by caching a third GoP, instead of serving the requests for the GoPs 3, 4 or 5 from the beginning of the segment, i.e., GoP 1, we can start by sending GoP 2 to the client. The new expected early start time is then determined by

$$\begin{aligned} E_{ES} &= \frac{L_G}{r_f} \cdot \frac{1}{L_S} \cdot \sum_{i=1}^{L_S-y-1} i \\ &= \frac{L_G}{r_f \cdot 2} \cdot \frac{(L_S-y) \cdot (L_S-y-1)}{L_S}. \end{aligned} \quad (7)$$

Please note, when $x < 1$, y will equal to 0 and (7) can be simplified to (3).

3.2. Expected Waiting Time

When all prefix frames are available in the cache, the waiting time is equal to 0. On the contrary, if some prefix frames are not available from the requested segment, i.e., $S_C \leq \frac{L_P}{L_S} \cdot S_V$, the client has to wait until the proxy finishes loading all missing frames from the remote server. A general estimated waiting time model can be presented by:

$$E_{WT} = \sum_{v=1}^{N_V} \sum_{s=1}^{N_S^v} \sum_{g=1}^{L_P} \Big(\sum_{f=1}^{L_G} \frac{S_F^{(v,s,g,f)} \cdot A_F^{(v,s,g,f)}}{R^v} \Big) \cdot P^{(v,s)}, \quad (8)$$

where $A_F^{(v,s,g,f)}$ determines the availability of the frame, which can be obtained by running PAPA. It is defined as

$$A_F^{(v,s,g,f)} = \begin{cases} 1, & \text{if frame } (v,s,g,f) \text{ is not in cache} \\ 0, & \text{otherwise} \end{cases}$$

Since this model needs the pre-run of PAPA everytime when any parameter changes, we herein propose the following two approximating schemes to estimate the waiting time.

1. **Fairness First** approach drops from every prefix GoP a percentage number of frames in such a way that the non-dropped data can fit into the cache. Therefore, E_{WT} in (8) can be replaced with

$$E_{WT} = \sum_{v=1}^{N_V} \sum_{s=1}^{N_S^v} \Big(\sum_{g=1}^{L_P} \frac{S_G^{(v,s,g)} \cdot p_d^{(v,s,g)}}{R^v} \Big) \cdot P^{(v,s)}, \quad (9)$$

where p_d^g is the dropping percentage from each prefix GoP, and can be identified by

$$p_d^{(v,s,g)} = 1 - \frac{p_s \cdot L_S}{L_P}. \quad (10)$$

2. **Popularity First** approach drops all prefix GoPs with the lowest popularity until the remaining data can fit into the proxy cache. Hence, we rewrite (8) with

$$E_{WT} = \sum_{v=1}^{N_V} \sum_{s=1}^{N_S^v} \Big(\sum_{(v,s,g) \in \mathfrak{G}} \frac{S_G^{(v,s,g)}}{R^v} \Big) \cdot P^{(v,s)}, \quad (11)$$

where $\mathfrak{G}$ is the set of prefix GoPs with low popularity.

In both techniques, we consider the GoP as the smallest accessible unit in a video as each GoP is independently decodable. Video content is dropped in number of bytes, without taking care of which frames are removed.

3.3. Final Satisfaction Score

By extracting the waiting time and early start values from the corresponding models and retrieving their scores from (1) and (2), we can generate a final score model to evaluate the overall performance of the VoD system as

$$S_F = S_{ES} - (S_{Top} - S_{WT}), \quad (12)$$

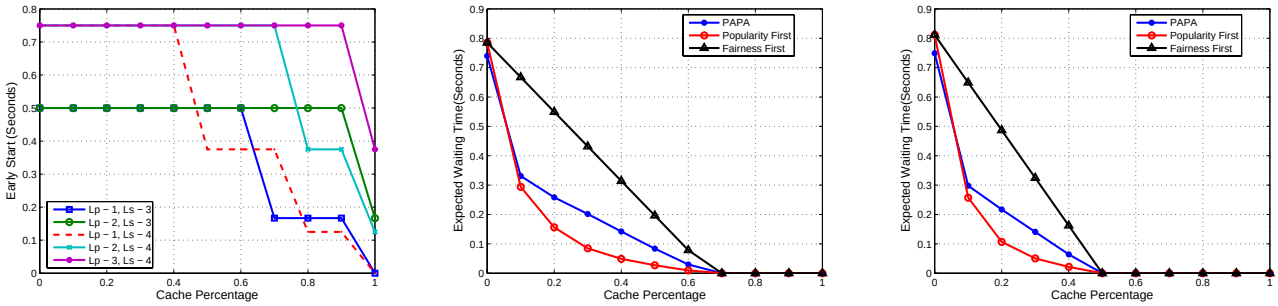
where S_F stands for the final score, and S_{Top} is the full score achieved by the subjective tests.

4. SIMULATION RESULTS

To evaluate our general model and its approximations, in this work, online trace files [9] are used for the experiments. The test video has a length of 3375 GoPs, GoP size of 16 frames, and 3 B-frames are placed between two I- or P-frames. Frame rate is assumed to be 32 fps, i.e., the playout duration of 2 GoPs is one second. The transmission rate used in the models is the mean rate of the test video.

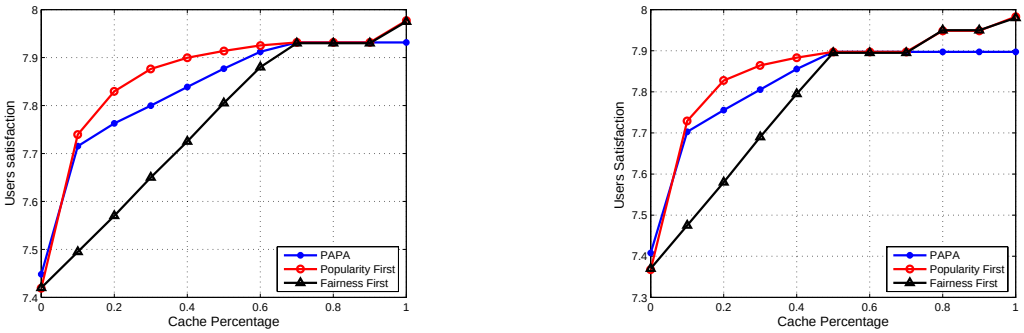
The results for our early start model are shown in Fig. 4(a). The X-axis represents the percentage of the videos cached on the proxy and the Y-axis represents the expected early start time in seconds. As can be noticed, for all segment-prefix formations, when there is enough cache size to store more suffix frames, early start is decreased. The early start time does not go to zero for some segment structures, because we restrict our prefix within one segment and the latest starting point of a segment is the $(L_S - L_P)$-th GoP in the segment.

Fig. 4(b), (c) show the simulation results for the waiting time by running the PAPA algorithm, as well as that obtained from the **Fairness First** and **Popularity First** models. The Y-axis represents the expected waiting time in seconds. In Fig. 4(b) we show the waiting time when the prefix length is 2 GoPs and segment length is 3 GoPs. As can be observed, the waiting time goes to zero when around 70% of the whole video content is cached. Whereas, Fig. 4(c) represents the results for a prefix length of 2 GoPs and segment length of 4



(a) Early Start Time (b) Waiting Time (L_P-2, L_S-3) (c) Waiting Time (L_P-2, L_S-4)

Fig. 4. Mean early start time and waiting time as a function of cache percentage



(a) L_P-2, L_S-3 (b) L_P-2, L_S-4

Fig. 5. Expected final score as a function of cache percentage

GoPs, where the waiting time is minimized to zero when 50% of the video frames are cached. Finally, for both segment-prefix structures, the waiting time is further minimized, when dropping more data from unpopular contents with the **Popularity First** approach. It performs better than normal PAPA that works on the frame level. The **Fairness First** approach performs the worst as prefix frames are evenly dropped without taking into account their individual popularities.

Fig. 5 shows the final score of the PAPA as well as the **Fairness First** and **Popularity First** approaches. The **Popularity First** approach achieves the highest score at all caching rates. It performs better than PAPA at low caching percentage because of the smaller waiting time and smaller early start at high caching percentage. What can be also found is that the larger the segment size, the bigger the influence of the early start.

5. CONCLUSION

In this paper we present an analytical model to estimate the waiting time achievable by some popularity-aware partial caching algorithms. Another model to estimate the early start time for the PAPA algorithm is also created. We make further improvement to the original algorithm and achieve better results. Finally, we set up a more sophisticated subjective tests environment to survey the influence of waiting time and early start on the user satisfaction in VoD services. Based on this observation, we obtain a universal model which gives subjective score to the system with given parameters. It can well reflect the user satisfaction for VoD services.

6. REFERENCES

[1] S. Sen, J. Rexford, and D. Towsley, "Proxy prefix caching for multimedia streams," in *Proc. IEEE International Conference on Computer and Communications (INFOCOM'99)*, New York, NY, Apr. 1999.

[2] K. L. Wu, P. S. Yu, and J. L. Wolf, "Segment-based proxy caching of multimedia streams," in *Proc. International Conference on World Wide Web (WWW'01)*, Hongkong, China, May 2001.

[3] M. Hofmann, T. S. Eugene Ng, K. Guo, S. Paul, and H. Zhang, "Caching techniques for streaming multimedia over the internet," Tech. Rep. BL011345-990409-04TM, Bell Labs, Holmdel, NJ, Apr. 1999.

[4] E. Bommaiah, K. Guo, M. Hofmann, and S. Paul, "Design and implementation of a caching system for streaming media over the internet," in *Proc. IEEE Real-Time Technology and Applications Symposium (RTAS'00)*, Washington, DC, May/Jun 2000.

[5] C. Zheng, G. Shen, and S. Li, "Distributed prefetching scheme for random seek support in peertopeer streaming applications," in *Proc. ACM Workshop on Advances in Peer-to-Peer Multimedia Streaming*, Hilton, Singapore, Nov. 2005.

[6] L. Shen, W. Tu, and E. Steinbach, "A flexible starting point based partial caching algorithm for video on demand," in *Proc. IEEE International Conference on Multimedia and Expo(ICME'07)*, Beijing, China, July 2007.

[7] A. Dan, D. Sitaram, and P. Shahabuddin, "Dynamic batching policies for an on demand video server," in *Multimedia Systems*, June 1996.

[8] G. K. Zipf, *Human Behaviour and the Principles of Least Effort*, Addison-Wesley, Cambridge, MA, 1949.

[9] Arizona State University, "Video traces for network performance evaluation," http://trace.eas.asu.edu/.