2010 3rd International Conference on Computer and Electrical Engineering (ICCEE 2010)
IPCSIT vol. 53 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V53.No.1.68

Segment-based Proxy Caching for Streaming Media Delivery

ZHAO Zheng-de[+] and ZHAO KAI

College of computer engineering and science
Shanghai University
Shanghai 200072, China

Abstract. At present the majority of streaming media files is large and it requires a lot of network bandwidth and disk bandwidth. We propose an adaptive segment-based method, the cache replacement method and the multi technologies combined optimized transmission policy. Simulation results indicate that they are highly efficient methods for use of caching proxy server resources, reducing the startup latency and save of bandwidth of backbone network.

Keywords: Streaming media; Patching algorithm; Segment-based proxy caching

1 Introduction

The traditional proxy cache is valid only on the text and other static objects, but scores poorly on the streaming media object which is large volume of data and requires a long-lasting access time. For streaming data, the delay will cause the client playback jitter. Data download before playing provides continuous playback, but to spark the startup latency. In addition, it requires a lot of caches in the client.

There are a number of approaches of partial buffer about how to use the proxy to cache streaming data ([1] [2] [3] [4] [5], etc.). However, these methods lack adaptive ability for the popularity of streaming media objects and changes of user access patterns. When a streaming media object is very hot, the vast majority of this file or all data can be stored in proxy cache. When its heat reduces, there is only a small part of data needs to be cached.

To solve the problem, this paper studies how to use existing mobile streaming media proxy cache server to effectively distribute streaming media content to the client,. The method of dynamic adjustment the size of sections according to changes in heat of media object and prefetching combined with patching stream transmission scheme are proposed.

2 Patching Algorithm

Patching algorithm combines the batching algorithm and user cache algorithm's virtues. To use local cache while receiving two or more media streams, customers can get services without delay. It uses the multicast stream as far as possible to merge customers and makes system to work with high efficiency [6]. Algorithms require customers to keep a data buffer, which should be able to cache the media stream. When the client sends the request behind the current multicast stream, the client uses cache to receive and cache the multicast stream. At the same time it generates a unicast media stream compensation for data, this unicast stream is called patching stream.

Patching algorithm is more efficient than other traditional dynamic flow dispatching algorithm. But it's also affected by certain factors [7]. In patching stream playback, multicast streams are cached in the client buffer, which requires the length of the patching stream cannot exceed the length of the buffer. Patching

[+] Corresponding author.
E-mail address: zhdzhao@163.com.

algorithm efficiency is effected by patching stream. Patching stream is sent to the client by unicast stream, which needs to consume system and network resources.

3 Segment-based Proxy Cache Strategy

3.1 Basic parameters

In order to effectively determine the size of segment and assess access model, The frequency of accessing stream media object λ_i and the average length of accessing stream media object r_i are global statistics, not just limited to a calculation in the course of access. It needs to cache all the data of the object at first to determine the size of the initial section [8]. This requirement is difficult to satisfy when the streaming media objects accessed are too many and the size of proxy cache is limited. In addition, in order to further optimize performance, these two variables can be expanded a dimensional array, which is statistics of the average access frequency and access length of a streaming media object in different period of time.

3.2 Segmentation

In the part-level cache strategy it needs to determine the size of section step by step and to dynamically adjust it to meet the requirements of the variability of streaming media object and the uncertainty of user access patterns. Figure 3.1 shows that it makes an assumption that $v_i = 20$ as the size of section when the client accesses an object first time (determination of the initial value of v_i according to the number N of streaming media objects and the size S of the proxy server cache). It determines the size of section according to the average interval of user access after the streaming media is played completely, that is $v_i = r_i$. And it is dynamic adjustment according to follow-up access. In this way, it can meet the needs of adaptive change.

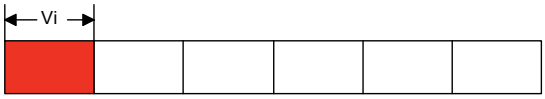

Figure 3.1 Determination of the Size of Section

3.3 Cache replacement algorithm

Resources of proxy server cache are limited, so after running for some time the contents of the cache must be changed. The key of replacement algorithm is how to select evicted segment. Exponential segment strategy uses LRU (least recently used) algorithm. In this paper, the principle of replacement took into account the user's access gap, frequency, bandwidth consumption and other factors to forecast the next access.

The forecast for the possibility of follow-up access can be used the following methods: set the current time T_c, the last access time T_r. If $T_c - T_i > 1/\lambda_i$, the probability of a new access is low. The opposite is high. In addition, the probability that the user reaches can also be calculated in a period of followed time according to Poisson distribution. Select the cache of the object which possibility of follow-up arrival is low as evicted segment.

When the calculation results of several objects are close, the bandwidth consumption can be calculated according to formula (2).Then choose the one that occupies the least bandwidth as the evicted segment and preserve the ones occupying larger bandwidth, therefore the backbone bandwidth occupation brought by the subsequent access can be reduced.

3.4 Optimized transmission scheme

Streaming media objects transfer between the proxy and the client by the combination of unicast and multicast, for saving network bandwidth as much as possible. In order to keep the continuity between the prefix and suffix, the suffix part can be requested in the course of prefix transmission, which is active prefetching method: when a request arrives in prefix time v_i, proxy server would be connected by unicast to send prefix; Suffix $L_i - v_i$ will be gotten by active prefetching through unicast connection between proxy server and content server before playing of first request reaches the tail of prefix, while proxy server builds multicast to transmit the contents of the suffix to the user. Acquisition of prefix t_2 which reaches after v_i is the same. The difference is acquisition of suffix. There are two methods: When the request reaches in patch time G_i, part $L_i - t_2$ will be gotten from multicast channel of the last suffix and lack part $t_2 - v_i$ will be

obtained by building a patch channel to content server. If the request reaches after patch time G_i, the proxy server will build a suffix channel to content server and create a multicast to internal network. According to [9], in the first case the backbone bandwidth occupied by the system is such as Formula 1 below:

$$C_s(i) = \frac{\frac{\lambda_i G_i^2}{2} + L_i - v_i}{1 + (v_i + G_i)\lambda_i} \lambda_i b_i \tag{1}$$

In another transmission the backbone bandwidth occupied by the system is such as Formula 2 below:

$$C_s(i) = \frac{L_i - v_i}{1 + v_i \lambda_i} \lambda_i b_i \tag{2}$$

To take full advantage of the existing multicast channel and reduce the backbone bandwidth usage, patching channel need to be established. But the price cannot exceed building a new multicast channel. It is assumed that here is a Poisson arrival process. In v_i, here are $1 + \lambda_i v_i$ times of requests. Backbone bandwidth occupied by using the first transmission is $\lambda_i G_i^2 / 2$ in period of time G_i. According to backbone bandwidth consumption in both scenarios, the size of G_i can be determined by $2(L_i - v_i) / (1 + v_i \lambda_i)$.

4 Experimental Results

Experimental environment has a content server, a proxy server and 15 PCs, server bandwidth is set as 10M, bandwidth between the proxy server and the PC is 100M. Subjects are 20 video files, of which total size close to 5G. There were 1,500 times of concurrent random access. The startup latency, hit rate, bandwidth occupancy, cache size and other information were recorded respectively, where the cache size is the ratio between the size of the cache S of proxy server and the size of all the streaming media object. Startup latency ratio is the ratio between the number of requests affected by startup latency and the total number of requests. Byte hit rate is defined as the ratio between number of bytes that the client get from the proxy server and the total number of bytes.

The experimental results showed in Figure 4.1 show that the proposed cache transmission strategy has a smaller

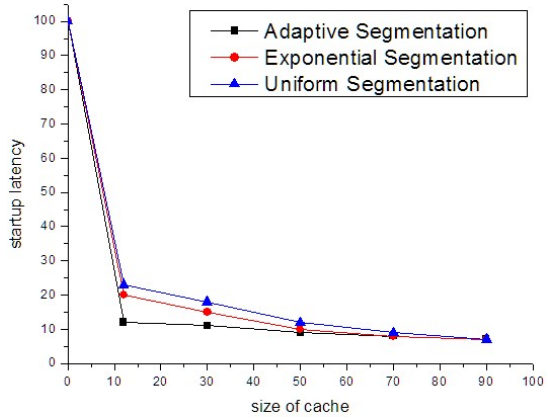

Figure 4.1: Startup Latency

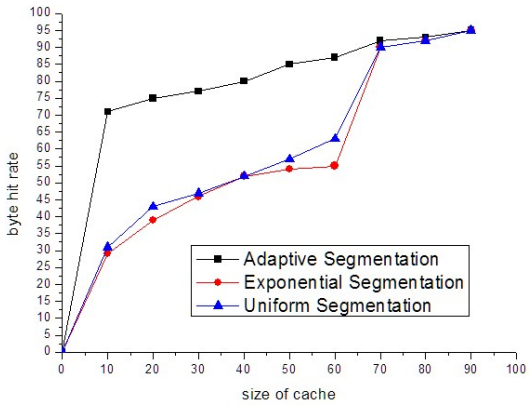

Figure 4.2: Byte Hit Rate

startup latency. Figure 4.2 shows adaptive Segmentation method has obvious advantages in byte hit rate. Figure 4.3 shows the content server and the proxy server's network bandwidth utilization in different strategies.

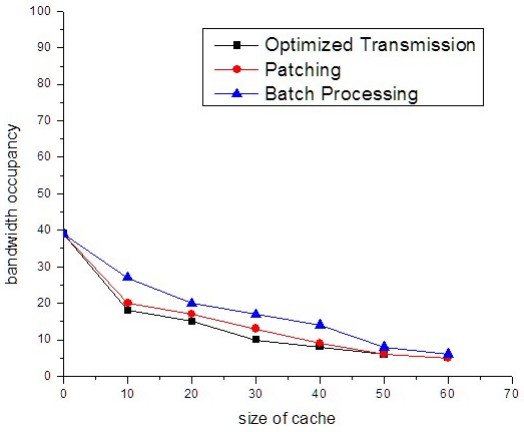

Figure 4.3: Bandwidth Occupancy

Adaptive segment-based method has advantages on startup delay. When the cache size is in 15%-35%, the effect is obvious in the byte hit rate. The adaptive segment-based method is better than the other two methods. In addition, the optimized transmission scheme also has some advantages on saving network bandwidth.

5 Conclusions

Adaptive segment-based method according to heat of streaming media objects, the cache replacement method and the multi technologies combined optimized transmission policy proposed in paper are highly efficient methods for use of caching proxy server resources, reducing the startup latency and save of bandwidth of backbone network.

6 References

[1] H. Fahmi, M. Latif, S. Sedigh-Ali, et al. Proxy Servers for Scalable Interactive Video Support[J]. IEEE Computer, 2001, vol. 34(9):54–60.

[2] Z. L. Zhang, Y. Wang, D. H. C. Du, et al. Video Staging: A Proxy-server-based Approach to End-to-End Video Delivery over Wide-area Networks[J]. IEEE Transactions on Networking, 2000, vol. 8:429–442.

[3] J. Kangasharju, F. Hartanto, M. Reisslein, et al. Distributing Layered Encoded Video through Caches[J]. IEEE Transactions on Computers, 2002, vol.51(6):622-636.

[4] S. Chen, B. Shen, S. Wee, et al. Investigating Performance Insight of Segment-based Proxy Caching of Streaming Media Stategies[C]//Proceedings of SPIE/ACM International Conference on Multimedia Computing and Networking(MMCN'04). Santa Clara, CA, United States.2004.Bellingham, WA 98227-0010, United States: International Society for Optical Engineering, 2004. 148-165.

[5] S. Podlipnig, and L. Boszormenyi. Replacement Strategies for Quality Based Video Caching[C]//IEEE International Conference on Multimedia and Expo(ICME). Lausanne, Switzerland. 2002. Piscataway, NJ, USA:IEEE, 2002. 49-52.

[6] M. Zink, O. Heckmann, J. Schmitt, et al. Polishing: A Technique to Reduce Variations in Cached Layer-encoded Video[C]. Proceedings of Multimedia Computing and Networking. San Jose, CA, United States. 2004. Bellingham, WA 98227-0010, United States: International Society for Optical Engineering, 2004. p187-198.

[7] Ivana Radulovic, Pascal Frossard, Olivier Verscheure. Adaptive Video Streaming in Lossy Networks: versions or layers? [C]//Proceedings of IEEE International Conference on Multimedia and Expo(ICME' 04). Piscataway, NJ, USA: IEEE, 2004. 1915-1918.

[8] K. L. Wu, P. S. Yu, J. L. Wolf. Segment-Based Proxy Caching of Multimedia Streams[C]//Proceedings of the 10th international conference on World Wide Web. New York, NY, USA: ACM, 2001. 36-44.

[9] WANG Wen-bo. Research on the Caching & Scheduling Strategy for P2P Streaming System[D]. Xi an: College of Information Science &Technology, Northwest University, 2009. 6.