Realtime Video Delivery Technique using Temporal Distance based Particle Domain Selection Method on the Internet

Jun Pyo Lee

*S/W R&D Lab., LIG Nex1, 702, Sampyeong-dong, Bundang-gu,
Seongnam-city, Gyeonggi-do, Korea
junpyolee@lignex1.com*

Abstract

Network based video proxy server can store the videos in order to minimize initial latency and network traffic significantly. However, due to the limited storage space in video proxy server, an appropriate video selection method is needed to store the videos which are frequently requested by clients. Thus, we propose a temporal distance based particle domain selection method using concurrent request pattern in large scale streaming system. We exploit the short-term temporal locality of two consecutive requests on an identical video. If the video is requested by user, it is temporarily stored during the predefined interval and then, delivered to the user. Due to the limited storage area in video proxy server, it is often required to replace the old video which is not serviced for long time with the newly requested one. This replacement causes the severe service delay and increase of network traffic. To circumvent these problems, we also propose an efficient storage space partitioning technique in order to stably store videos and present a time based free-up storage space technique using the expected variation of video data in order for avoiding the overflow on a video proxy server in advance. In order to verify the effectiveness and efficiency of the proposed method, we conduct a variety of simulations and demonstrate that our particle domain selection method leads to better performance.

Keywords: *Multimedia streaming, Delivery technique, QoS management, Video-on-demand service*

1. Introduction

In recent years, there has been an increasing interest in Video-on-Demand (VoD) service. A number of multimedia applications such as distance learning, digital library, video conferencing, and Entertainment-on-Demand rely on the streaming technique. In order to realize multimedia applications such as VoD over Internet, suitable quality of service (QoS) of the continuous media such as audio and video have to be guaranteed in accordance with users' requirements, available computing and network resources during the service session. The current Internet transmits video data in best-effort way using streaming technique and can't guarantee the QoS for video streaming. Thus, it is important to find a feasible solution for the explosively increasing video applications. There are three main obstacles in video streaming over the Internet: inadequate bandwidth in the Internet to deliver a large number of concurrent video streams, limited capacity of a VoD server, and limited capacity of a client. In general, these obstacles can be solved by video proxy server. A video proxy server is essentially a middle computer system that sits between the client and the central video server which is located at the remote location [1-5]. By storing video data, a video proxy server close to the clients can be used to assist video delivery and alleviate the load of the central video servers. This video proxy server can partially satisfy the need for rapid multimedia data delivery by providing multiple clients with a shared storing location. The requested videos are always delivered from the central video server through the video proxy server to clients, thus the video proxy server is able to intercept and store these videos to decrease the amount of video data that has to be delivered by the central video server. In this context, if a requested video exists in a storage area in video proxy server, clients get a stored video, which delivery time is typically reduced.

The storing and deletion techniques of video proxy server are one of the key solutions to improve the performance of multimedia service systems on the large scale network environment. However, since the storage capacity of video proxy server does not have an infinite-capacity for keeping all the continuous video data, the challenge for the video proxy server is to determine which videos should be stored or removed from the storage area of video proxy server. The well-known storing and deletion techniques using video proxy server are interval storing [6] method and distance storing [7] method. Nevertheless, these two techniques are memory based storing method and not appropriate to be applied to the disk based storing which has limited bandwidth. The LRU (Least Recently Used) [8] method is a recentness algorithm while the LFU (Least Frequently Used) [9] and the PLFU (Partial Least Frequently Used) [10, 11] methods are the frequency one. Due to the large sizes of streaming media objects, these algorithms are not providing optimal performance in the internet environment.

In this paper, the real-time video delivery technique using temporal distance based particle domain selection method is proposed in order to efficiently store frequently requested video in video proxy server. A video which is requested by user is loaded in temporary storage space in video proxy server. And then, this video is deleted or moved into the permanent storage space of video proxy server depending on the user request pattern. In addition, in order for storing the video data efficiently and avoiding the external fragmentation effectively due to the fact that the storing and deleting are frequently performed, the storage space of video proxy server is divided into specific data areas. Also, we present a time based free-up storage space technique using the expected variation of video data in order for avoiding the overflow on a video proxy server in advance.

2. Temporal Distance based Particle Domain Selection

2.1. Video segment storing technique using temporal locality

In this section, we present the video segment storing technique using temporal distance based partial domain selection method for large scale streaming system. We exploit the short-term temporal locality of two consecutive requests on an identical video. For this purpose, videos are divided into segments of equal length and the video proxy server loads the requested video segment which is delivered from the central video server in temporary storage area and forwards it to users directly. The requested video segment is not residing in the video proxy server in first loaded in temporary storage area. But, if the same video segment is requested once again by a different user in case when the previous request is consuming the same video, we utilize the time distance between the earlier request and the latest request. Within this time distance, video proxy server can preserve the segment which is delivered by the latest request in temporary storage area.

In addition, if the new request starts to utilize the video in temporary storage area, video proxy server can store the segments which are utilized by the new request in order to exploit the finite-capacity storage area efficiently. We expect that this video will be requested a lot of times within a short time. Figure 1 depicts the detail scenario of our temporal distance based particle domain selection method.

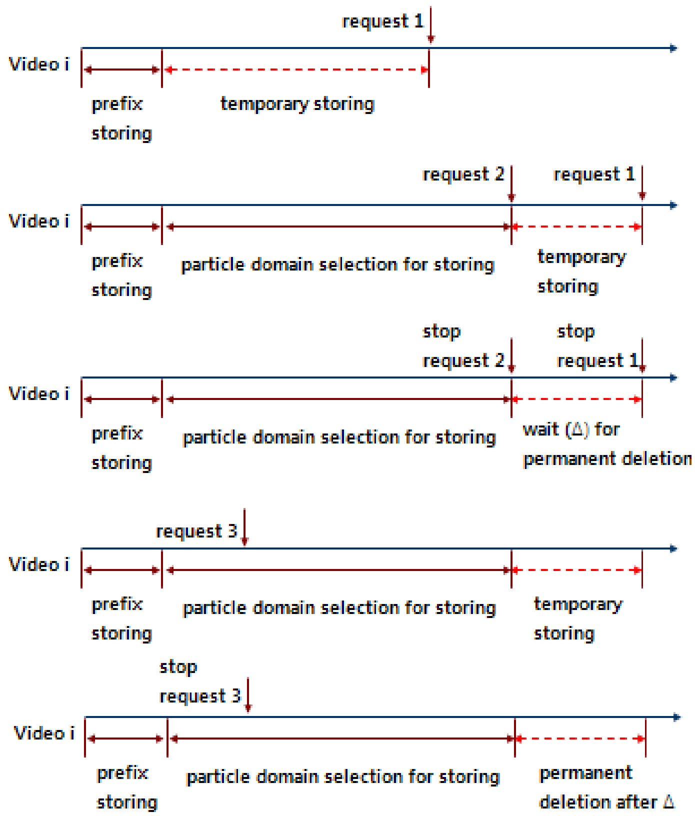

Figure 1. The scenario of temporal distance based particle domain selection method

In figure 1, prefix size and expectation time (Δ) of video i are determined by average initial latency within a predefined interval, and calculated by equation (2.1) respectively. In this paper, optimal value of predefined interval is derived by simulations.

$$\mu_i(k) = \frac{1}{n}\sum_{i=1}^{n}\alpha_i(k), \tag{2.1}$$

where, $\mu_i(k)$ and n denote the expectation time of segment k of video i and total number of segment k, respectively. Also, $\alpha_i(k)$ represents the time difference between recently request time and its previous request time.

In order to avoiding the excessive time distance and the storage space consumption for temporary storing, we set the limit time difference (γ) which is also derived by simulations.

2.2. Time-based free-up storage space technique

Since the storage space of video proxy server does not have infinite-capacity storage for keeping all the continuous video data, the video proxy server should have an appropriate deletion algorithm to make storage space for newly stored data in case when there is not free space enough to hold the new one. The challenge for the deletion algorithm is to determine which video segment should be stored or removed from the storage area of video proxy server. The crucial aspect of deletion algorithm lies in selecting victim. This deletion causes the service delay and increase of network traffic. To circumvent this problem, we propose an efficient storage space partitioning technique in order to stably store video segments and present a time based free-up storage space technique using the expected variation of video data in order for avoiding the overflow on a video proxy server in advance. Figure 2 depicts the linked list based file structure for video storing and deletion in video proxy server.

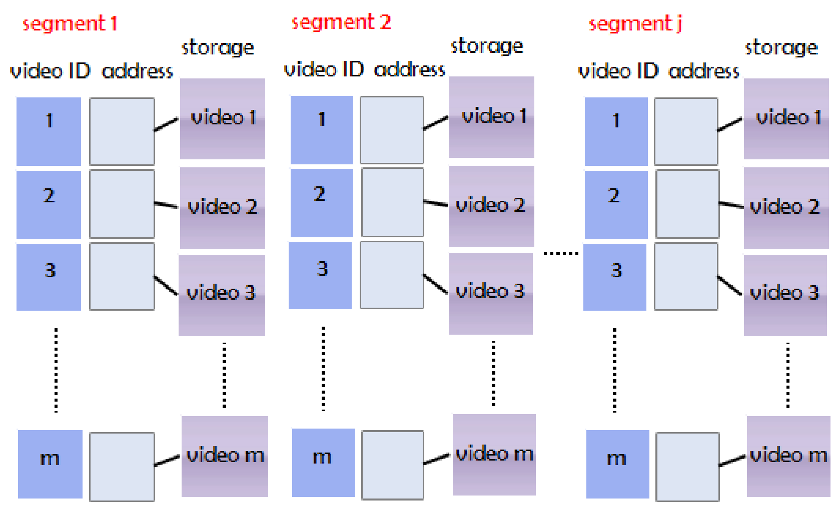

Figure 2. Segment based file structure in video proxy server

In this paper, we utilize the allocated size of each segment for efficiently storing and deletion in segment based file structure in video proxy server. The allocated size of each video segment within a predefined interval is calculated using equation (2.2).

$$SVS_k = \frac{\{P(k) \times S(k)\}}{\sum_{i=1}^{m} \{P(i) \times S(i)\}} \times TAS, \tag{2.2}$$

where, SVS_k and m denote the allocated size of video segment k and total number of each segment group in storage area. Also, $P(\bullet)$, $S(\bullet)$, and TAS represent the request probability which is calculated request number of specific segment divided by request number of all segments, the size of segments which are stored in storage area, and total available storage size of video proxy server, respectively.

In order for efficiently utilizing the storage space in video proxy server, we expect the required storage space for storing and then, free-up storage space in advance within a predefined interval using the variation value of video segment. In this paper, the variation value of video segment is calculated using equation (2.3).

$$VOS(k) = \left\lceil \beta(k) \times P_i(k) \times \frac{1}{n} \sum_{i=1}^{n} \alpha_i(1) \right\rceil, \tag{2.3}$$

where, $\lambda_i(k)$ denotes the difference value of request number of segment k and $Cnt_i(k)$ and $Cnt_i(k+1)$ represent request number of segment k and $k+1$, respectively.

We select the key segment which remarks the maximum request difference value (λ) and remove the last segment in deletion segment group in deletion segment group when the storage capacity is running out of space. Figure 3 depicts the preservation segment group and deletion segment group.

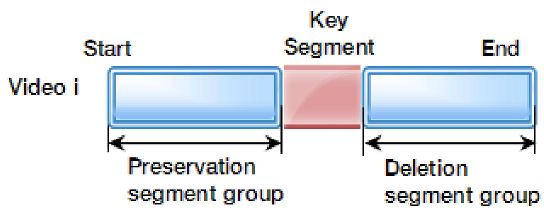

Figure 3. Preservation and deletion segment group

In addition, if a segment which is located in preservation segment group of video i is not requested within a expectation time of segment k, $\mu_i(k)$, this segment is moved to deletion segment group directly. Our segment deletion process is shown in figure 4.

3. Performance Evaluation

In general, hit rate has been used as the performance measurement of the storage schemes for traditional data such as text and image. But it is not proper for continuous media data. Thus, we use the hit rate of block defined as a segment, which can reflect the proxy server management method. In addition, we also use the number of block deletion in order to verify the efficiency of storage management method more correctly. We compare our method with the well-known algorithms such as PLFU (Partial Least Frequently Used) [10], Distance-based [7] and Reallocation Affinity [12] method through simulations. In order to verify the effectiveness of the proposed method, we conduct simulation under fixed-length segmentation. In fixed-length segmentation, a video is divided into multiple segments (blocks) for storing and deletion segments efficiently and readily. We consider a set of approximately 654Mbyte long videos and assume that the request rate is 1,200, 2,400 and 3,600 per minute, respectively. The detail simulation parameters are shown in Table 1.

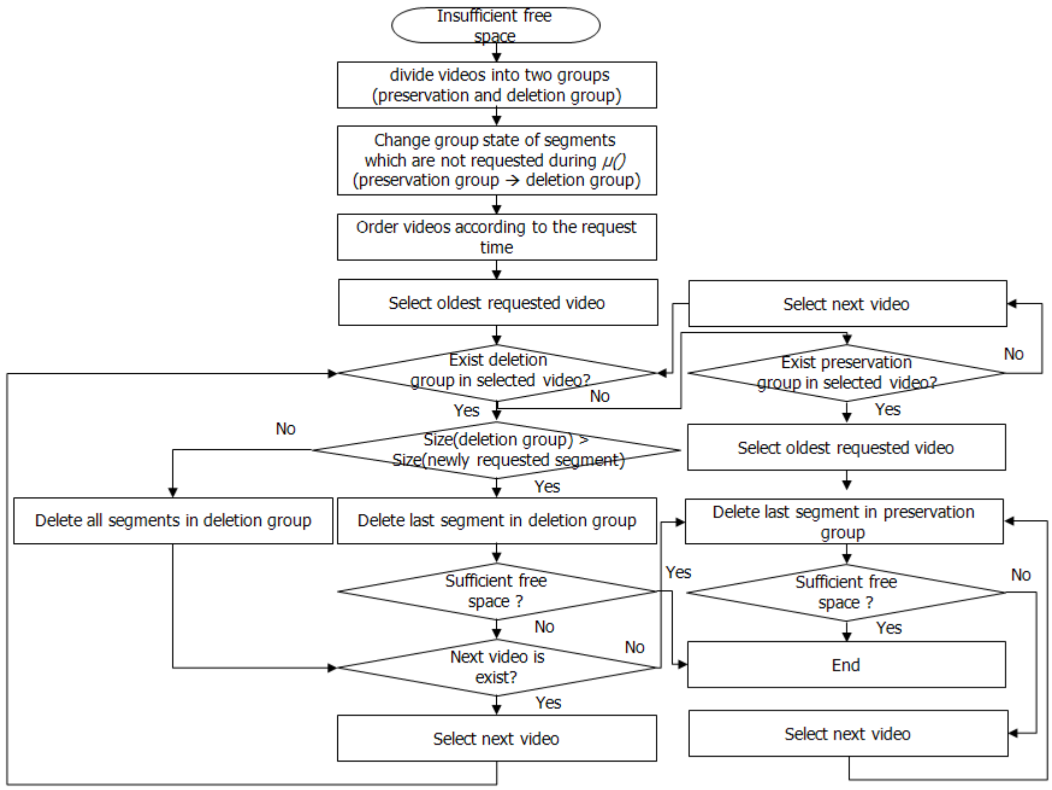

Figure 4. Flowchart of the segment deletion algorithm of video proxy server

Table 1. Simulation parameters

Parameters	Value
Simulation time	72 hours
Number of video	1,200
Video size	654.7 MB
Bit rate	1,024 Kbps
Block play time	5 second
Request time	1,200, 2,400, and 3,600 per minutes
Interval size	10 minute
Limit time difference (γ)	45 second
Storage size	50, 100, 150, 200, 250, 300, and 350 GB

In our simulation, to verify the effectiveness of our method, we conduct simulations under random access pattern when the video proxy storage sizes of 50, 100, 150, 200, 250, 300, and 350GB are used respectively. Figure 5, 6, and 7 show that the hit rate of our method compared with that of other well-known algorithms under various video proxy server storage sizes and different video request rates. The simulation results show that the proposed algorithm performs better than other algorithms such as PLFU, Distance-based and Reallocation Affinity method in terms of block hit rate.

The next simulation verifies the efficiency of proposed method. Figure 8, 9, and 10 show the efficiency of our temporal distance based partial domain selection method in terms of deletion, where the number of deletion of our method is significantly fewer than that of other algorithms under various video proxy server storage sizes and different video request rates.

Our performed various simulations can be calculated and summarized as follows. In the case of figure 5 and 8(video request rate: 1,200), the proposed algorithm performs about 5.87% better than PLFU, about 5.61% better than Distance-based, and about 3.67% much better than Reallocation Affinity method. And, in the case of figure 6 and 9(video request rate: 2,400), the proposed algorithm performs about 7.82% better than PLFU, about 5.27% better than Distance-based, and about 3.17% much better than Reallocation Affinity method. Finally, in the case of Figure 7 and 10(video request rate: 3,600), the proposed algorithm performs about 5.71% better than PLFU, about 3.72% better than Distance-based, and about 3.76% much better than Reallocation Affinity method. These improvements may be due to the fact that the segmented video which represents a low request ratio can be deleted quickly in storage area of video proxy server. Therefore, we prove that the deletion overhead of our method is not significant compared to other algorithms.

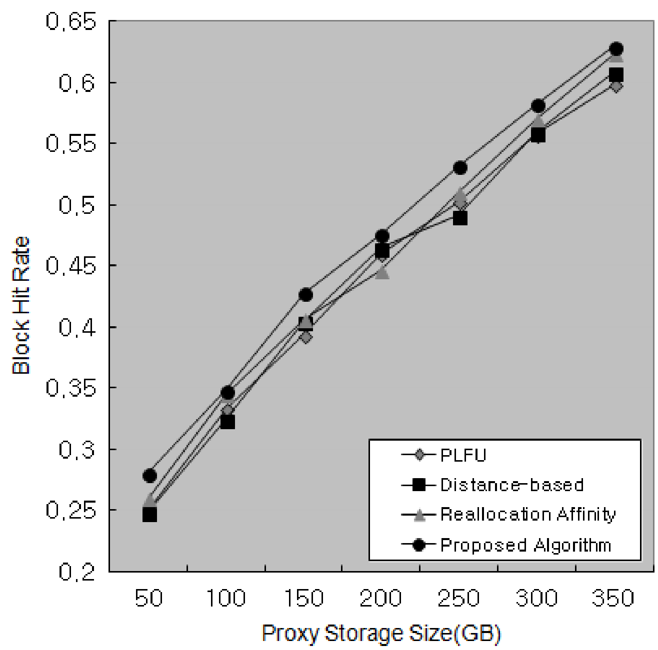

Figure 5. Comparison of block hit rate under various video proxy server storage sizes (video request rate: 1,200)

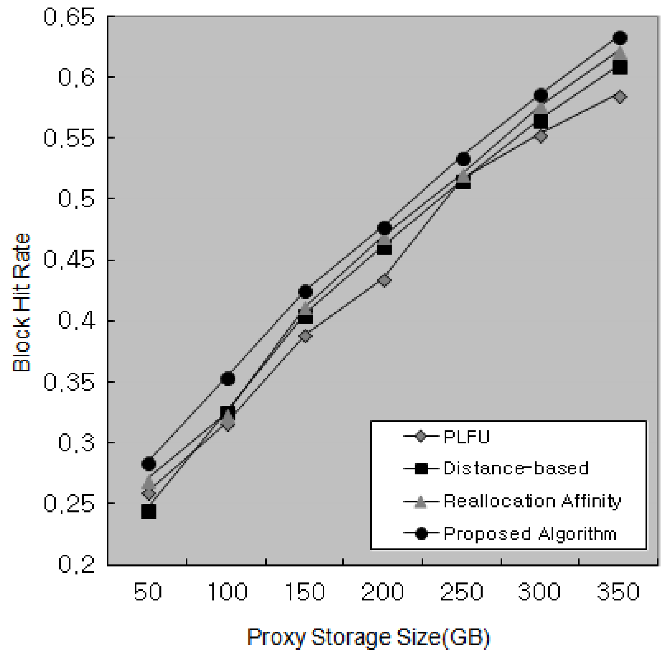

Figure 6. Comparison of block hit rate under various video proxy server storage sizes (video request rate: 2,400)

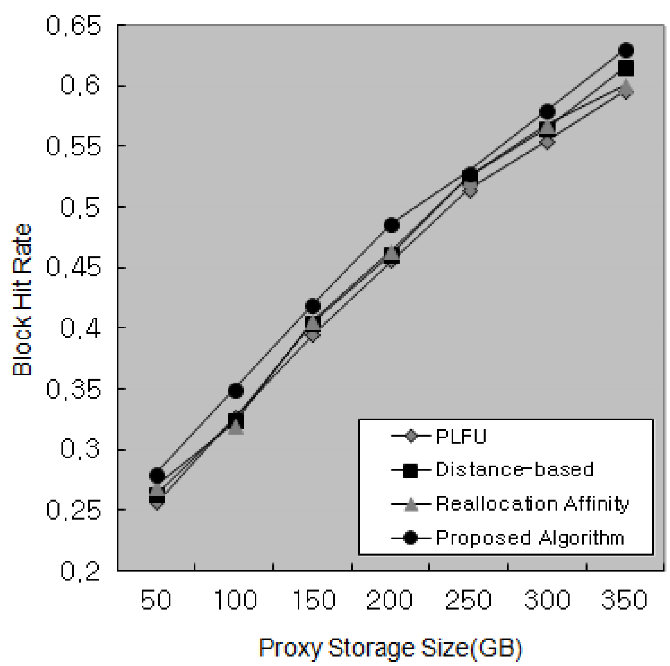

Figure 7. Comparison of block hit rate under various video proxy server storage sizes (video request rate: 3,600)

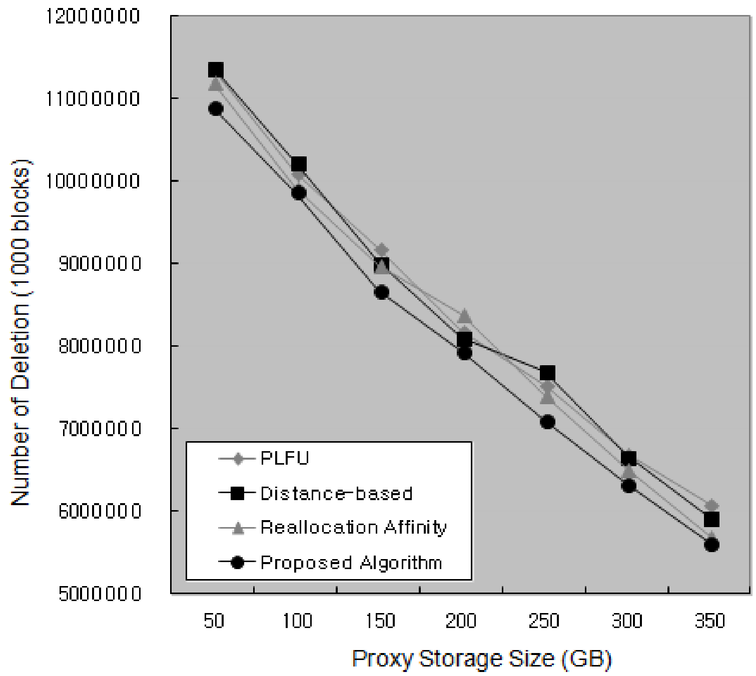

Figure 8. Comparison of number of deletion under various video proxy server storage sizes (video request rate: 1,200)

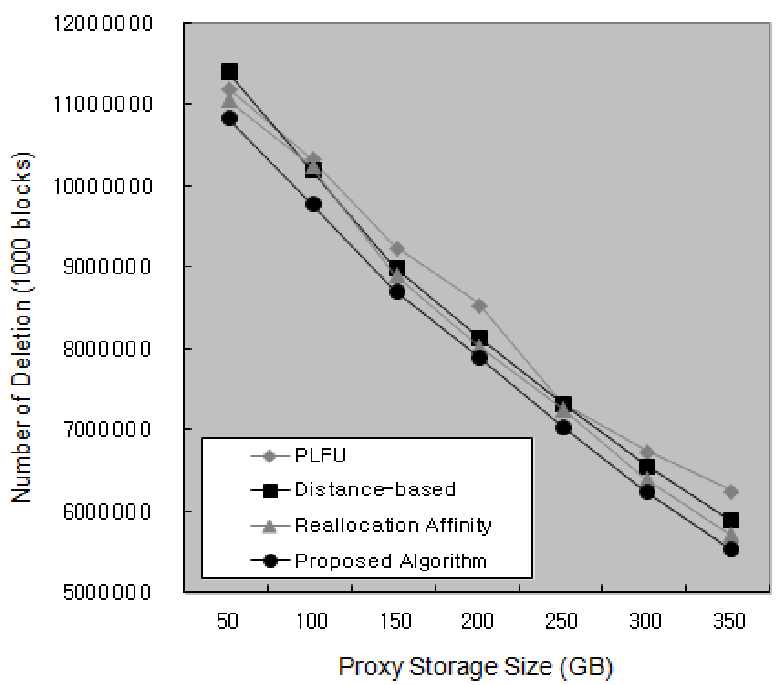

Figure 9. Comparison of number of deletion under various video proxy server storage sizes (video request rate: 2,400)

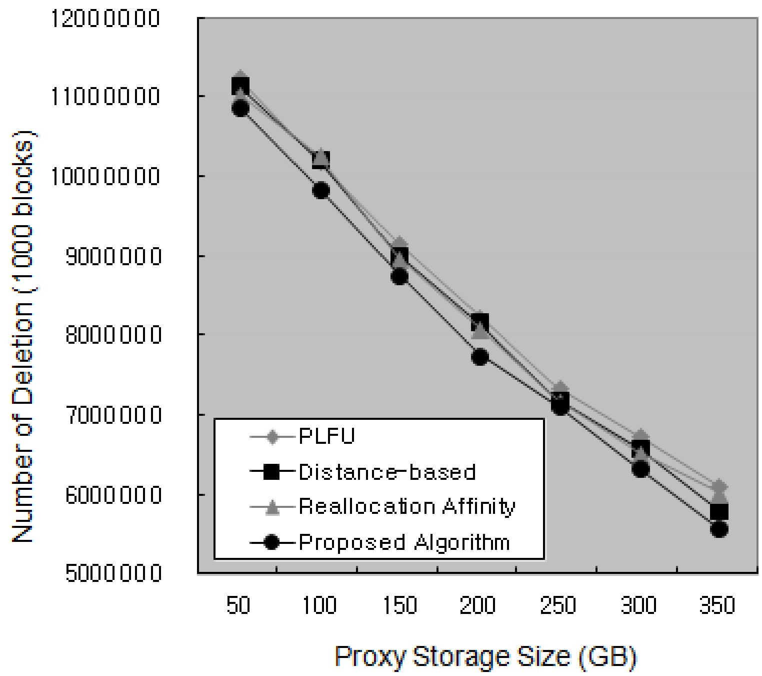

Figure 10. Comparison of number of deletion under various video proxy server storage sizes (video request rate: 3,600)

4. Conclusions

The storing and deletion technique of video proxy server are one of the key solutions to improve the performance of multimedia applications on large scale network environment. By storing frequently accessed video data at a storage area, client perceived latency, central server load, and network traffic can be reduced significantly. However, since existing storing and deletion techniques are for traditional data such as text and image, they are not suitable to continuous media data. Thus, we present a temporal distance based particle domain selection method using concurrent request pattern in large scale streaming system in order to efficiently store frequently requested video. In this paper, a video which is requested by user is loaded in temporary storage space in video proxy server. And then, this video is deleted or moved into the permanent storage space of video proxy server depending on the user request pattern. In addition, in order for storing the video data efficiently and avoiding the external fragmentation due to the fact that the storing and deleting are frequently performed, the storage of video proxy server is divided into specific data areas. Also, we present a time based free-up storage space technique using the expected variation of video data in order for avoiding the overflow on a video proxy server in advance. Finally, we can save storage space without degrading performance in video proxy server. Through simulation, we evaluate the performance of our temporal distance based particle domain selection method and compared with other well-known algorithms such as PLFU, Distance-based and Reallocation Affinity method. We demonstrate that the introduction of the concept of partial storing and deletion leads to better performance.

References

[1] C. -L. Chang, X. -H. Hsieh and W. -M. Chen, "The design of video streaming proxy in high-speed train", Proceedings of the International Conference on Information Networking, **(2012)** February 1-3; Yunlin, Taiwan.
[2] T. Wu, K. De Schepper, W. Van Leekwijck and D. De Vleeschauwer, "Reuse time based caching policy for video streaming", Proceedings of the IEEE International Conference on Consumer Communications and Networking, **(2012)** January 14-17; Antwerp, Belgium.
[3] J. Dai, F. Liu, B. Li, B. Li and J. Liu, "Collaborative Caching in Wireless Video Streaming Through Resource Auctions", Journal on Selected Areas in Communications, vol. 30, **(2012)**.
[4] Y. -T. Yu and S. -R. Tong, "An Adaptive Suffix-Window Caching Scheme for CM Proxy Server", Proceedings of the International Conference on Network-Based Information Systems, **(2010)** September 14-16; Kaohsiung, Taiwan.
[5] W. Tu, E. Steinbach, M. Muhammad and X. Li, "Page Help Proxy Caching for Video-on-Demand Using Flexible Starting Point Selection", IEEE Transactions on Multimedia, vol. 11, **(2009)**.
[6] B. Ozden, R. Rastogi and A. Silberschatz, "Buffer replacement algorithms for multimedia storage systems", Proceedings of the IEEE International Conference on Multimedia Computing and Systems, **(1996)** June 17-23; Murray Hill, NJ.
[7] S. Chen, B. Shen, S. Wee and X. Zhang, "Segment-based streaming media proxy: modeling and optimization", IEEE Transactions on Multimedia, vol. 8, **(2006)**.
[8] Dan and D. Towsley, "An approximate analysis of the LRU and FIFO buffer replacement schemes", Proceedings of the ACM SIGMETRICS conference on Measurement and Modeling of Computer Systems, **(1990)** May; NY, USA.
[9] H. Chou and D. DeWitt, "An evaluation of buffer management strategies for relational database systems", Proceedings of the International Conference on Very Large Data Bases, **(1985)** April 28; London Guildhall Univ., London.
[10] K. -S. Hsueh and S. -D. Wang, "A packet-based caching proxy with loss recovery for video streaming", Proceedings of the International Symposium on Dependable Computing, **(2002)** December 16-18; Taipei, Taiwan.
[11] S. Gonzalez, A. Navarro, J. Lopez and E. Zapata, "Load Sharing based on Popularity in Distributed Video on Demand Systems", Proceedings of the IEEE International Conference on Multimedia and Expo, **(2002)**; Malaga Univ., Spain.
[12] C. Spielvogel and L. Boszormenyi, "Quality-of-Service based video replication", Proceedings of the International Workshop on Semantic Media Adaptation and Personalization, **(2007)** December 17-18; Uxbridge, London.

Author

Jun Pyo Lee

Dr. Jun Pyo Lee is the corresponding author of this paper. He received the M.S. and Ph.D. degrees in computer science and engineering from Hanyang University, Republic of Korea, in 2003 and 2009, respectively. Since 2009, he has been a research engineer in Department of Software R&D Laboratory at LIGNex1, Republic of Korea. His research interests include multimedia system, distributed computing, image processing, ubiquitous computing, unmanned system, and intelligent control.