International Journal of Multimedia and Ubiquitous Engineering
Vol.9, No.3 (2014), pp.85-96
http://dx.doi.org/10.14257/ijmue.2014.9.3.09

Dynamic Scheduling Algorithms for Streaming Media Based on Proxy Caching in Education system

Zhang Yuhong[1], Xu Zhikun[2] and Gao Yabin[2]

[1] *College of Education, Hebei Normal University of Science & Technology, Qinhuangdao, Hebei, P.R.China, 066004*

[2] *College of Finance, Hebei Normal University of Science & Technology, Qinhuangdao, Hebei, P.R.China, 066004*

13933690306@163.com, xuvikp@126.com, 522865482@qq.com

Abstract

With development of digital multimedia, many applications, such as VoD, remote education, and so on, becomes a reality. Multimedia resource in education system becomes very rich. With the increasing number of multimedia files in school server, how terminal computers in classrooms fetch multimedia at fast speed and low cost is an important problem to solve. Multimedia used in education system has a long history. In order to decrease bandwidth consumption, a new method of dynamic schedule algorithm based on patching prefetching and proxy prefetching has been proposed. Proxy cache first use unicast to get data from server, and then, data sending to each terminal computer would adopt multicast. Proxy server pre-fetches the patching data for the subsequent request from the ongoing entire stream and caches them at the buffer. Simulation results show that the algorithm can effectively decrease bandwidth without add space of cache.

Keywords: *education system; streaming multimedia; prefix caching; scheduling algorithm*

1. Introduction

Multimedia has greatly changed human's lifestyle. We can find its application in various areas, including advertisements [1], art [2], education, entertainment [3], engineering [4], medicine, mathematics [5], business, scientific research [6], and spatial temporal applications. Multimedia has also used in multidisciplinary [7]. It nearly appears with every aspects in our life. Whatever we do, we would contact multimedia.

In order to improve efficiency of teaching, multimedia is widely used in education. Due to its vivid pictures and videos, multimedia helps teachers give students more clear expression. The technology can help students to concentrate on learning and improve efficiency. Visualization technology can change abstract and boring lesson to be vivid 3D picture or video to improve intuition. In fact, studies on multimedia used in education have been greatly developed. The studies focus on many subjects, including teaching method [8-10], education system [11-13] and education tools [14-15]. Education direction is also researched [16].

Here, we would concentrate on the management of multimedia in education system. We would develop an algorithm to manage the multimedia files and distribute them properly. With the development of distributed system, education system now is integrated into school network. Computers in classroom or students' computers are as terminal computers in the system in the future. All resources would be stored in server and each terminal computer in classroom or other places request the resource. In the system, data transmission can use

ISSN: 1975-0080 IJMUE

streaming media technology. The server sends resource to proxy and proxy sends resource to terminal computers. This mechanism can reduce the bandwidth and improve the utilizing efficiency. Also, it can decrease a lot of cost in memory. In education system, multimedia in classroom must play smoothly, with no delay and low jitter. Or else, it may be influence teaching effect. So, the algorithm of how to send multimedia to terminal computers is very important.

In internet, many algorithm is developed. But in education system, it has its own characteristics. For example, curriculum is influenced by term and in each term, some curriculums would be taught to students at the same time, terminal computers would request for same files concurrently. But in internet, users send request is random. In education system, the algorithm is different from that in internet. However, the two systems have some same characteristics too. For example, they can use steaming media technology.

In the paper, we develop a new algorithm for education system. Streaming multimedia system is used in the education system. Server stores all the resources and sent to proxy. Proxy sends resources to terminal computers. The system is showed in Figure 1.

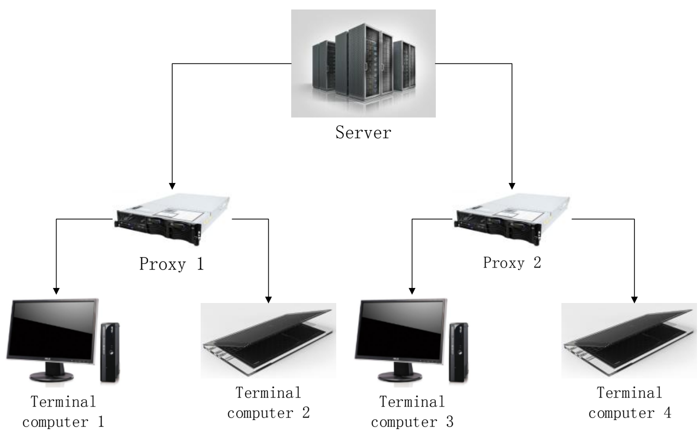

Figure1. Proxy based streaming multimedia system

Usually, streaming multimedia system operation needs segments dividing, caching mechanism, data transmission and other aspects. Segments dividing has some common method [17-18] and caching methods are also has relative studies [19-20]. Also, indispensable part of the technology is data delivering mechanism. This mechanism is divided into merging [21-23], batching [24] and patching [25].

Algorithm in the paper refers data transmission mechanism between server, proxy and terminal computers. In the system, the prefix part is pre-stored in proxy server before lesson starting at the aim of decreasing the start delay in the terminal computer. Adding to data transmission technology, more terminal computers hare a same multicast streaming to effectively reduce the bandwidth and server resource consumption with good performance. When one file has high access frequency, it would still consume more resource of the system according to algorithms introduced before.

The main contribution of the paper is development of new Dynamic Scheduling Algorithms based on Proxy Caching for Streaming Media in Education system. Algorithm can decrease the bandwidth of main network, server and proxy cache, especially with high access multimedia object. The remainder of the paper is organized as follows: Related algorithms are revisited in Section 2. Dynamics scheduling algorithm is derived in Section 3. The simulation is given in Section 4 and conclusion is in Section 5.

2. Existing Technology Analysis

Batching technology [26] give a short delay to an access request to wait for more arriving of accesses. It process many accesses at a time and start a multicast to serve more users. Patching technology is an effective and timely dynamic schedule technology [27]. In the algorithm, bandwidth is seen a group logical channel and bandwidth of each channel equals video play rate. Regular channel delivers the whole multimedia object and patching channel deliver patches. Data flow in regular channel is called regular flow or regular stream and in patching channel is called patch flow or patch stream. If an access arriving, server checks whether the flow is in channel. If not, server start a new one. If has, notice the computer to receive the flow and start patching channel to deliver patches. Usually, in order to get better performance, combination algorithm of batching and patching is often used.

OBP + Prefix caching (optimized batch patching & prefix caching) [28] method needs terminal computers with good performance. Computers should have capability of receiving three streams. The three streams received are prefix stream, patch stream and regular stream respectively. The algorithm also needs big cache size. When the algorithm is in use, patches would be fetched repeatedly from server. That is to say that the algorithm waste the bandwidth of server and main link in network.

OBP + Prefix & patch caching (optimized batch patching + Prefix & patch caching) [29] is derived from OBP + Prefix caching. The algorithm uses patches caching in segments. In a patch window W, terminal computer can fetch patch segments from last batch time zone. Then, in different batch time zone, terminal just fetch the missing segments from sever. This algorithm can fetch less segments than OBP + Prefix caching and save bandwidth. At the same time it just need that the terminal computer receives two streams.

OBP + Prefix & patch caching has better performance than OBP + Prefix caching. But, when accesses have high arriving ratio, proxy would process lots of accesses in each batch time zone. In this condition, proxy may deliver data as regular stream in one time zone and process same data as patch stream in next time zone. This is a waste of system resources. In order to avoid the phenomenon, dynamic scheduling algorithm has been proposed in the paper.

3. Dynamics Scheduling Algorithm

In education system, unicast is also used in network between server while proxy and multicast is used in network between proxy and terminal computers. We should make some assumptions before describing the algorithm. Access from each terminal computer is always from the beginning of media file. The proxy would pre-store prefix of multimedia.

3.1 Dynamics scheduling algorithm

Basic idea of the algorithm is proxy server pre-fetch and cache patches when it delivers regular data flow. Whether to pre-fetch is due to the whether there are accesses in. If there is no access in during batch processing zone but has in next zone, missing patches is appeared and these patches need to fetch from server through patch channel. The algorithm can greatly decrease number of patches through patch channel, decrease flow rate across main link path and reduce the number of concurrent flow in server.

As shows in Figure 2. Multimedia object with time length L is divided into two parts: prefix and postfix. Client request A, B, C and D point to a same object. If we define time length of prefix $L_p = b$ and all the rest of the object is postfix $L_s = L - L_p$. For the convenient study, the postfix is also divided in to many segments to respond the time length of batch processing. T represents the threshold of start-up period in regular channel.

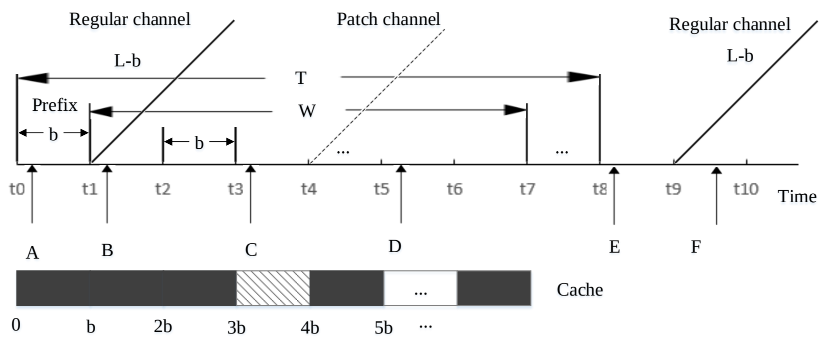

Figure 2. of Dynamics scheduling algorithm

Detail description:

(1) For those Client requests of arriving time in $t \in [t_0, t_1)$, proxy server would deliver the prefix L_p according to unicast. If there has no postfix of request object, the proxy would request server for regular flow $L-b$ at time t_1. At the same time, the proxy allocate a space of size b to cache the coming data section $[b, 2b)$. The data section would be as patch of client request in time zone of $[t_1, t_2)$. The proxy deliver the data to clients after accept the regular flow. Client with request arriving time in $[t_0, t_1)$ would join the multicast channel.

(2) There are also client request arriving at $[t_1, t_2)$, proxy server would deliver the prefix to each client according with unicast. It also allocate an cache space with size b to cache data $[2b, 3b)$. The data is as patch of client request with arriving time in $[t_2, t_3)$. In addition, proxy deliver the patch data through patch channel. As system operates at time t_3, two patches of $[b, 2b)$ and $[2b, 3b)$ have been cached in proxy cache area.

(3) In patch window $[t_0, t_0 + W)$, if there is no request arriving, proxy server would not allocate cache space. For example, there is no request arriving in zone $[t_2, t_3)$, space would not allocate at time t_3. If there is request arriving in time zone of $[t_3, t_4)$ and proxy just cache two patches of $[b, 2b)$ and $[2b, 3b)$ with missing patch $[3b, 4b)$, proxy would fetch the missing patch through patch channel. In this condition, the proxy allocates space with size of $2b$ at time t_4. The allocate space would cache patches of $[3b, 4b)$ and $[4b, 5b)$.

(4) If client request arrives in time zone of $[t_0 + W, t_0 + L_p + W)$, proxy would not pre-fetch patches. This is the biggest cache size the proxy allocates. Based on the distribution condition, patches pre-fetched in proxy cache as well as re-delivery patches according to starting patch channel are also showed in figure 2. Pre-fetched patches are showed in shadow area and re-delivery patches are showed in oblique line area.

(5) If $T > t_0 + L_p + W$ and request arriving time is in $[t_0 + L_p + W, t_0 + T)$, the proxy can't allocate space for multimedia object. Then, proxy start patch channel to fetch patch with time length of $t - W - L_p$ and deliver them to clients.

(6) If client request arrives at time $T > t_0 + T$, proxy ask server to deliver a new regular flow and begin a next period.

In the algorithm, proxy use prefix cache and unicast to deliver prefix of media to each client. This can diminish the time delay in each classroom. The method can provide the normal class. Rest data is delivered according to patch data or regular data through multicast mechanism. This method can raise transmission efficiency. When arriving rate of requests of terminal computers is higher, patches according to patch channel to deliver would be less. Ideal condition is that requests arrive in every time zone and patch pre-fetching method can satisfy all the demands of terminal computer. The worst condition is just one request arrives in the last time zone. That means proxy misses （N-1)b patches. It needs proxy to start patch channel to fetch the missing patches.

3.2 Analysis of algorithm

In order to analyze the performance of the algorithm, many parameters need to calculate, such as, patches number in cast period between adjacent groups, mean bandwidth of server output, bandwidth of network and proxy cache size, and so on. In order to compare with other algorithm simply, we make an assumption of the system in ideal condition. The ideal condition includes no deliver time delay, no jitter, and so on. The streaming multimedia use CBR (constant bit rate) as coding way and terminal computers request the multimedia play from the beginning part.

Assumptions in the algorithm are showed as the following.

(1) Requests from terminal computers obey Poisson distribution. Mean request arriving rate is λ . Then, probability in a time zone without any requests is $p = e^{-\lambda b}$. This is to say that time interval between two terminal computers is bigger than b . Probability in a time zone with requests is $1 - p$

(2) y_i is the number of requests in the i^{th} time zone. y_1, y_2, y_3,… y_N are independent and with the same distribution.

(3) Whether needs patch service in i^{th} batch processing is up to values of y_{i-1} and y_i . If $y_{i-1} = 0$ and $y_i \neq 0$, patch channel is needed to start up to deliver patch of $[(i-1)b, ib)$. If $y_i = 0$, patch channel is not needed to start. If y_1, y_2, y_3,… y_N are equal, whether they are 0 or not 0, patch channel is not needed to start.

(4) In patch window W , size of patch service is μ_N .

$$\mu_N \in \{\mu_N \mid \mu_N = 0, 1b, 2b, \ldots (N-1)b\}$$

If p_i represents the probability of $i * b$, that is to say

$$p(\mu_N = ib) = p_i, \qquad i = 0, 1, \ldots, N-1$$

So, we can get the expectation of μ_N

$$E\mu_N = b\sum_{i=0}^{N-1} i \cdot p_i = b\sum_{i=1}^{N-1} i \cdot p_i$$

Rest work is to determine p_i

$$p_0 = p(\mu_N = 0)$$
$$= (1-p)^N + (1-p)^{N-1}p^1 + (1-p)^{N-2}p^2 + ... + (1-p)^2 p^{N-2} + (1-p)p^{N-1}$$
$$= C_{N-0}^0(1-p)^N + C_{N-1}^1(1-p)^{N-1}p^1 + C_{N-2}^2(1-p)^{N-2}p^2 + ...$$
$$+ C_2^1(1-p)^2 p^{N-2} + C_1^0(1-p)p^{N-1}$$
$$= \sum_{j=0}^{N} C_{N-j}^0(1-p)^{N-j}p^j$$

$$p_1 = p(\mu_N = 1b)$$
$$= (N-1)(1-p)^{N-1}p^1 + (N-2)(1-p)^{N-2}p^2 + ... + 2(1-p)^2 p^{N-2} + (1-p)p^{N-1}$$
$$= C_{N-1}^1(1-p)^{N-1}p^1 + C_{N-2}^1(1-p)^{N-2}p^2 + ...$$
$$+ C_2^1(1-p)^2 p^{N-2} + C_1^1(1-p)p^{N-1}$$
$$= \sum_{j=1}^{N-1} C_{N-j}^1(1-p)^{N-j}p^j$$

$$p_2 = p(\mu_N = 2b)$$
$$= (N-2)(1-p)^{N-2}p^2 + (1-p)^{N-3}p^3 + ... + 2(1-p)^2 p^{N-2} + (1-p)p^{N-1}$$
$$= C_{N-1}^2(1-p)^{N-2}p^2 + C_{N-2}^2(1-p)^{N-3}p^3 + ... + C_3^2(1-p)^2 p^{N-2} + C_2^2(1-p)p^{N-1}$$
$$= \sum_{j=1}^{N-2} C_{N-j}^2(1-p)^{N-j-1}p^{j+1}$$

$$p_3 = p(\mu_N = 3b)$$
$$= (N-3)(1-p)^{N-3}p^3 + (N-4)(1-p)^{N-4}p^4 + ... + (1-p)^2 p^{N-2} + (1-p)p^{N-1}$$
$$= C_{N-1}^3(1-p)^{N-3}p^3 + C_{N-2}^3(1-p)^{N-4}p^4 + ... + C_3^4(1-p)^2 p^{N-2} + C_3^3(1-p)p^{N-1}$$
$$= \sum_{j=1}^{N-3} C_{N-j}^3(1-p)^{N-j-2}p^{j+2}$$

So, total probability of p_i can be deducted as the following:

$$p_i = \sum_{j=1}^{N-i} C_{N-j}^i(1-p)^{N-i-j+1}p^{i+j-1}$$

Where, $i = 1,2,...,N-1$. And

$$E\mu_N = b\sum_{i=1}^{N-1}\sum_{j=1}^{N-i} i \cdot C_{N-j}^i(1-p)^{N-i-j+1}p^{i+j-1}$$

Then, we can know mean patch number μ in arbitrary two regular multicast flow.

$$\mu = \begin{cases} E\mu_N + \dfrac{\lambda(T - W - L_p)^2}{2}, & W + L_p < T \\ E\mu_N, & W + L_p \geq T \end{cases}$$

Mean bandwidth of server output is

$$R = \frac{\mu_r + (L - L_p)r}{T + 1/\lambda} = \frac{\mu_r + (L - b)}{T + 1/\lambda}$$

Mean proxy cache occupied is:

$$s = br + (\mu + \mu')r$$

Where, μ' is patches fetched from regular channel. br represents cache memory occupied by prefix and $(\mu + \mu')r$ represents that of patches occupied. μ' can be determined by the following equation.

$$\mu' = b + (1 - p)(N - 1)b = b + (W - b)(1 - p)$$

Bandwidth of proxy is determined as the following equation.

$$B = \frac{(\lambda Tb + \lambda W^2 / 2)r}{T + 1/\lambda} + R$$

Maximum cache memory in terminal computer is $(W + b)r$.

4. Simulation

In the simulation, start period T in regular channel is set to be $W + b$. That is to say maximum cache memory proxy can allocate is equal to that of period threshold of regular flow start up.

The streaming multimedia objects code with MPEG-1 with the transmission speed of 1.5Mbit/s, space size is in 300MB~800MB and time length is 60min; prefix size is 40s. For the reason of OBP + Prefix & patch caching having better performance than OBP + Prefix caching, the simulation just compares the results of dynamic schedule algorithm and OBP + Prefix & patch caching algorithm.

Table 1. Percentage of bandwidth saved under dynamic schedule algorithm

	Percentage of bandwidth saved/%		
	λ=0.25	λ=1	λ=5
W=1	100	100	100
W=5	87.29	72.40	99.52
W=10	32.52	66.82	99.41
W=15	28.38	64.99	99.38
W=50	22.54	64.08	94.05

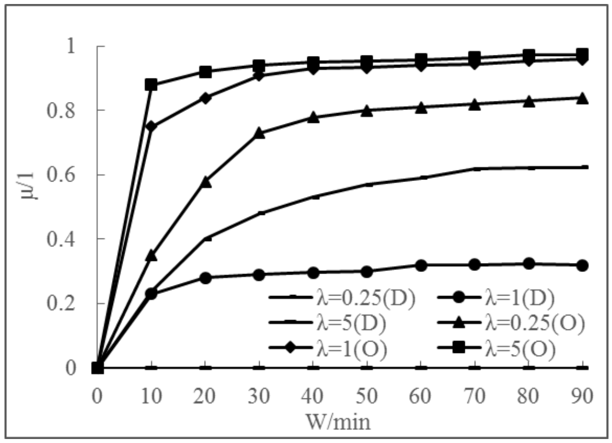

Figure 3. Relation between patches and W

Figure 3 shows bandwidth consumed by two algorithms and (D) represents dynamic schedule algorithm while (O) represents OBP + prefix & patch caching. It is obviously that dynamic schedule algorithm can decrease bandwidth consumption. With λ grows, percentage saved of bandwidth of patch channel increases. Table 1 shows the percentage of bandwidth saved in patch channel under dynamic schedule algorithm.

Table 2. Percentage of bandwidth saved in main link of system

	Percentage of bandwidth saved/%			
	λ=0.25	λ=1	λ=5	λ=10
W=1	0.25	0.69	1.08	1.05
W=5	1.23	3.35	4.32	5.28
W=10	2.29	6.51	9.28	10.45
W=15	3.35	9.08	13.54	14.79
W=20	4.18	11.76	17.72	18.23
W=50	8.29	22.96	35.76	38.96

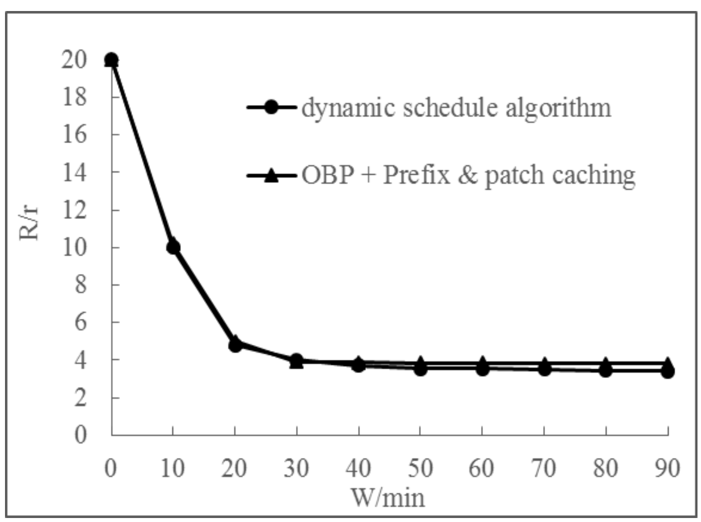

Figure 4. Bandwidth saving under two algorithms ($\lambda = 0.25$)

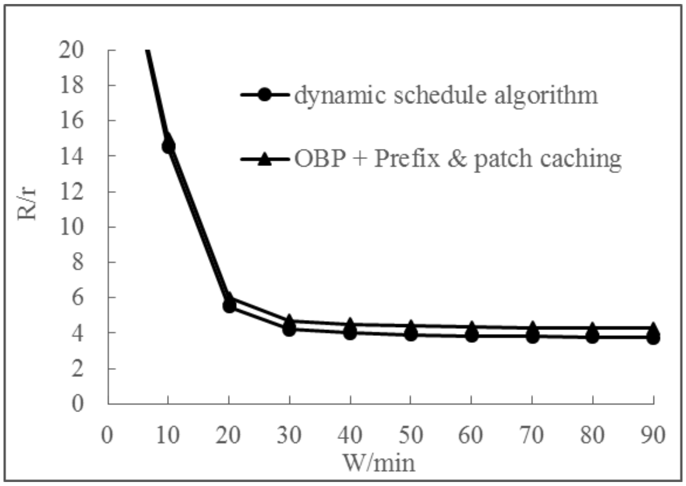

Figure 5. Bandwidth saving under two algorithms ($\lambda = 1$)

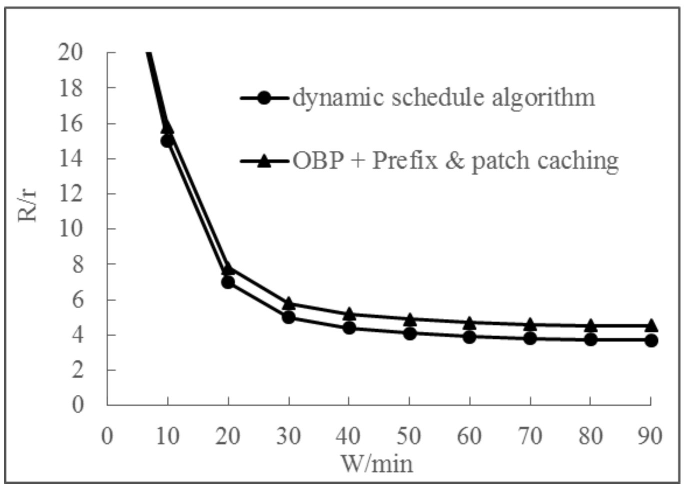

Figure 6. Bandwidth saving under two algorithms ($\lambda = 5$)

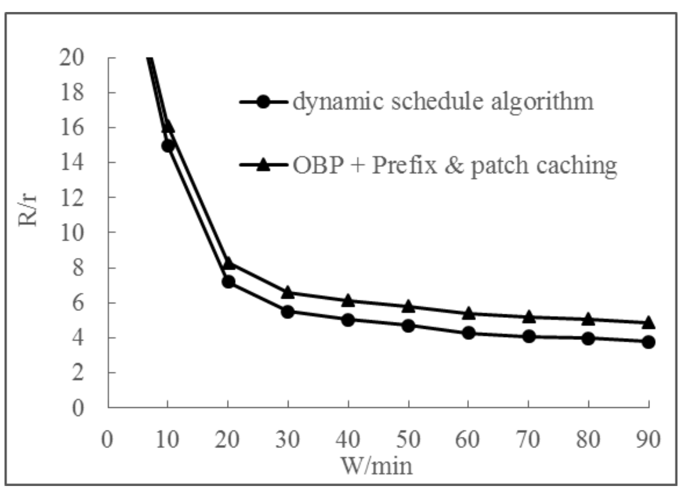

Figure 7. Bandwidth saving under two algorithms ($\lambda = 10$)

Figure 4 to Figure 7 show bandwidth consumed in different access ratio under the two algorithms. We can see that bandwidth of main link in network decreases with size of window W increases and dynamic schedule algorithm has a higher decreasing ratio and consumes less bandwidth. Especially with bigger λ, the difference is even more. Table 2 shows the bandwidth saving of main link in system under dynamic schedule algorithm.

5. Conclusion

Decades has passed since multimedia technology used in education system. In the future, with the development of education and increasing number of multimedia files, streaming multimedia would be used in education system to save hardware resource.

In streaming distribution system, schedule algorithm is a key technology in the network. In order to improve data delivering efficiency and saving bandwidth, dynamic schedule algorithm has been proposed in the paper. The algorithm allocate corresponding space with time distribution of access arriving. With prefetching and caching of regular flow, number of patches in the system is greatly decreased. It also reduce the bandwidth of proxy and main link in network. According to the simulation results, the new algorithm has a better performance than OBP + prefix & patch caching. The algorithm is especially used in education system with high accesses ratio at short time.

In education system, some multimedia files with great importance should be pre-cached in proxy, and, with middle importance should be partly pre-cached in proxy. The algorithm is based on access ratio to deliver data flow and adjust patch window size with access arriving ratio. When the ratio is high, window size should be enlarged, if not, the size should be reduced.

There are still many problems in the system to be resolved, and schedule algorithm still need to improve. Further research is in progress.

References

[1] J. Lee, H. -J. Kim and J. Cho, "Charging station advertisement on digital multimedia broadcasting platform", Communications in Computer and Information Science, vol. 233, **(2011),** pp. 12-18.

[2] W. C. Yin, K. C. Weng, K. Chu, P. Mhd, M. Asyraf and W. M. Leong, "Audio-visual documentation method for digital storytelling for a multimedia art project", Lecture Notes in Computer Science, 8007 LNCS (PART 4), **(2013)**, pp. 750-758.

[3] Y. -J. Lee, H. C. Du David and W. -h. Ma, "SESAME: a scalable and extensible architecture for multimedia entertainment", Proceedings - IEEE Computer Society's International Computer Software & Applications Conference, **(1996)**, pp. 56-61.

[4] G. P. Oscar and B. -C. Anna, "Work in progress - Soft skills within a project - Based approach: A multimedia engineering degree", Proceedings - Frontiers in Education Conference, **(2010)**, (S2J1-S2J3).

[5] M. Zurina, Mohamed and R. E. Kartina, "Multimedia design and development in mathematics learning courseware for preschool education", Proceedings - International Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA 2005 and International Conference on Intelligent Agents, Web Technologies and Internet, vol. 2, **(2005),** pp. 514-517.

[6] G. Bo, S. Rahul and J. Ramesh, "Research Explorer: Gaining insights through exploration in multimedia scientific data", MIR'04 - Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval, **(2004)**, pp. 7-14.

[7] A. I. Reuther and D. G. Meyer, "The effect of personality type on the usage of a multimedia engineering education system", Proceedings - Frontiers in Education Conference, 1(T3A/7-T3A/12), **(2002)**.

[8] L. Tang and X. Wang, "Logistics courses network multimedia teaching method", 2010 International Conference on E-Health Networking, Digital Ecosystems and Technologies, vol. 2, **(2010),** pp. 448-451.

[9] J. Guo, F. Chen, G. Zhao, X. Huang and J. Liao, "The research on the value of multimedia teaching method in the course of university physics", OPEE 2010 - 2010 International Conference on Optics, Photonics and Energy Engineering, vol. 2, **(2010),** pp. 178-180.

[10] L. Lin and Y. Xiang, "A multimedia teaching method research of animation production in China", 2012 7th International Conference on Computer Science and Education, **(2012),** pp. 1495-1497.
[11] A. L. Y. Lowa, K. L. T. Lowb and V. C. Kooc, "Multimedia learning systems: a future interactive educational tool", Internet and Higher Education, vol. 6, **(2003),** pp. 25–40.
[12] T. K. Shih and R. E. Davis, "IMMPS: a multimedia presentation design system", IEEE Multimedia, vol. 4, no. 2, **(1997),** pp. 67–78.
[13] P. M. Lawther and D. H. T. Walker, "An evaluation of a distributed learning system", Journal of Education and Training, vol. 43, no. 2, **(2001),** pp. 105–116.
[14] K. Premkumar, W. Hunter, J. Davison and J. Penny, "Development and validation of an evaluation tool for multimedia resources in health education", International Journal of Medical Informatics, vol. 50, no. 1-3, **(1998),** pp. 243-250.
[15] C. A. Carver Jr., R. A. Howard and W. D. Lane, "Enhancing student learning through hypermedia courseware and incorporation of student learning styles", IEEE Transactions on Education, vol. 42, no. 1, **(1999)**, pp. 33–38.
[16] Y. Hu and W. Gui, "Multi-media Technology in education: where are they come and where are they go?", Proceedings of the 2007 1st International Symposium on Information Technologies and Applications in Education. **(2007)** November 23-25, Kunming, China.
[17] R. Rejaie, M. Handley, H. Yu, *et al.*, "Proxy caching mechanism for multimedia Playback streams in the internet", In Proceedings of International web Caching Workshop. **(1999)** March, San Diego, CA.
[18] K Wu, P. S. Yu and J. Wolf, "Segment-based proxy caching of multimedia streams", In Proceeding of WWW, **(2001)** May, Hong Kong, China.
[19] R. Rejaie, H. Yu and M. Handely, *et al.*, "Multimedia proxy caching mechanism for quality adaptive streaming applications in the internet", In Proceeding of IEEE IN FOCOM, **(2000)** March, Tel-Aviv, Israel.
[20] T. Kim and M. H. Ammar, "A comparison of layering and stream replication video multicast schemes", In Proceedings of ACM NOSSDAV 2001, **(2001)** June, NY.
[21] C. C. Aggaral, J. L. Wolf and P. S. Yu, "On Optimal piggyback merging policies for video-on-demand systems", Proc. ACM SIGMETRICS Conf., **(1996)** May, Philadelphia, PA.
[22] D. L. Eager, M. K. Vermon and J. Zahorjan, "Minimizing bandwidth requirements for on-demand data delivery", In Proceedings of the 5-th. International Workshop on Advances in Multimedia Information System (MIS'99), **(1999)**, pp. 80-87.
[23] B. N. Amotz and Ladnerre, "Efficient algorithms for optimal stream merging for media-on-demand", SISM J. Comput., vol. 33, no. 5, **(2004),** pp. 1011-1034.
[24] C. Aggarwal, J. Wolf and P. Yu, "On optimal batching policies for video on demand storage servers", In Proc. IEEE Int. Conf. Multimedia Computing and systems, **(1996)**, pp. 253-258.
[25] K. Hua, Y. Cai and S. Sheu, "Patching: a multicast technique for true video on demand services", In Proc. ACM Multimedia, **(1998)**, pp. 191-200.
[26] A. Dan, D. Sitaram and P. Shahabuddin, "Scheduling policies for an on2demand video server with batching", In Proceedings of ACM Multimedia, **(1994)**, pp. 15-23.
[27] Y. Cai, K. A. Hua and K. Vu, "Optimizing patching performance", In Proceedings of ACM/ SPIE Multimedia Computing and Networking, **(1999)**, pp. 203-215.
[28] P. White and J. Crowcroft, "Optimized batch patching with classes of service", ACM SIGCOMM Computer Communication Review, vol. 30, no. 4, **(2000),** pp. 21-28.

Authors

Zhang Yuhong

She received her bachelor's degree of education in Hebei Normal University of Science & Technology, Qinhuangdao, Hebei. (2001) and master's degree of education in Hebei Normal University, Baoding, Hebei. (2007), Now she is a lecturer in Hebei Normal University of Science & Technology, Qinhuangdao,Hebei. Her major fields of study are education technology, vocational education, and information education.

Xu Zhikun

He received his bachelor's degree of Economics in Inner Mongolia Agricultural University , (1993) and master's degree of management in North china Electric Power University, (2010) , Now he work in hebei normal university of science and technology institute of finance and economics,Qinhuangdao, Hebei. He is a Master's Supervisor . Her major fields of study is practical economics.

Gao Yabin

She received her bachelor's degree of Science in TANGSHAN Normal University Tangshan, Hebei, (2004) and master's degree of Software Engineering in Beijing University of Technology, Beijing, (2012), Now she is a lecturer in Hebei Normal University of Science & Technology, Qinhuangdao, Hebei. Her major fields of study are Education and teaching management,software engineering and education.

