A Fuzzy-based Method for Reducing Mobile Video-quality Fluctuation

Linh Van Ma[1], Sanghyun Park[2], JaeHyung Park[3], Jiseung Nam[4], Jonghyun Jang[5], Jinsul Kim[6]
[1,2,3,4,6]School of Electronics and Computer Engineering
Chonnam National University, Gwangju, Korea
linh.mavan[1]@gmail.com

sanghyun079[2]@gmail.com

hyeoung[3]@ chonnam.ac.kr

jsnam[4]@ jnu.ac.kr

[5]Electronics and Telecommunications Research Institute
218 Gajeong-ro, Yuseong-gu, Daejeon, 34129, Korea
jangjh[5]@ etri.re.kr

Abstract

Mobile devices are susceptible to changing bandwidth when moving. Current HTTP-based adaptive streaming deals with such bandwidth situation by providing an appropriate video representation for current available bandwidth. Recent studies on adaptive streaming showed that solving network problems with predictions could improve the quality of service. Therefore, in this paper, we propose a method of predicting downloading segments to solve the problems of frequent network changes. In this method, we first calculate the average of previous bandwidth values, the average time to download previous segments. Second, we trace-back at some earlier bandwidth values, average downloading time values to infer a degree of current network fluctuation. We then predict a new request segment for a mobile client by using fuzzy logic. In the experiment, our results show that the method significantly reduces the fluctuation compared to previous research.

Keywords: *Dynamic Adaptive Streaming, Fuzzy Inference, Moving Average, Bandwidth Fluctuation, Mobile Video Quality, Bandwidth Usage.*

1. Introduction

The fast growing internet users in recent year causes many problems in the internet delivering services. For example, a user might have rapidly fluctuated throughput from 7Mbps to 0 and then back up. More specifically, it starts off blazing fast, time out for a while then come back up. That is a typical case when mobile users move around frequently. In another case, a mobile user has signed up to a 5G service/80 Mbps recently, and he experienced internet speed drops to as low as 3G/800 Kbps. Typically, the speed usually fluctuates at a weak-signal place such as a remote area. The fluctuation changes a lot, like dropping to 800 Kbps and might just stay on for a few hours before the speed returns to normal.

To solve such fluctuation network, Dynamic Adaptive Streaming over HTTP (DASH) [1] has recently emerged as a server-client driven. The server divides a video file into multiple chunks and encodes a source video at different rates. The client does two tasks: 1) periodically measures server-to-client bandwidth; 2) chooses the maximum coding rate sustainable given current bandwidth. However, this streaming technology still has some challenges with the problems of frequent and sudden change of the network. For example, the challenge could be a change in video quality along with the network condition.

In this paper, we propose a prediction bandwidth method to solve the problems of frequent network changes such as bandwidth. We implement our proposed method in a DASH-based open source. Besides, we evaluate the performance of the method in comparison with previous works. This paper is organized as follow. We describe the related works in Section 2. Details of proposed method are given in Section 3. Section 4 describes and present the method performance. Finally, we conclude this research in Section 5.

2. Related Research

Adaptive Bitrate Streaming (ABS) is an approach of distributing multimedia content from a server to a client end-user. The approach is HTTP-based which use HTTP as the transmitting protocol to deliver the content over the networks. Thus, it receives many benefits from the HTTP-based technology such as packets can traverse the firewall and devices without having many difficulties in a Network Associate Translation (NAT). Besides, a server and a client do not require a persistent connection between them, and the server does not need to retain session information on the client.

The ABS technique works by encoding a multimedia media source at different bit-rates such as a video file with different resolutions. The bit-rates vary from low quality to high quality to serve a user in all conditions of the network bandwidth. The technique then divides each bit-rate into multiple chunks. Continually, each chunk divides into small parts or segments with a duration length which varies from two to ten seconds. All the information of the segments and chunks are included in a manifest Media Presentation Description (MPD) file. The information contains chunks location, segment duration, type of bit-rate, etc. By this way, a playback client can recognize the multimedia stream aiming to create multimedia streaming. The client usually does two tasks to create the streaming. It first measures its speed connection to a server streaming. It then chooses the appropriate bit-rate which is sustainable to the given measurement. The advantage of the technique is that it can switch the streaming among the different bit-rates automatically without awareness of a user. If the network speed is greater than a certain bit-rate, then it requires downloading the next adjacent higher bit-rate.

While most of the previous research focused on a client side, the authors [2] formed a problem of finding a set of representations in adaptive streaming because no any scientific study supported the problem. They addressed that the issue of network changes can be solved if we have forecasting algorithms. The Crystal-ball algorithm [3] considered buffer occupancy and bitrates to predict bandwidth. They also analyzed when the predictions are useful, what effect of prediction quality on performance. Their experiment results showed that prediction-based algorithms improve the quality of service (QoS). However, their research did not pay attention to the value of bandwidth before the present time. It might result in incorrectness prediction results. In the other hand, Geo-predictive crowdsourcing [4] has an approach on clients' previous download experience. Their crowdsourcing-based algorithms outperformed all other methods regarding QoS. They used client's buffer level and current segment to predict bandwidth without considering last bandwidth values. If they could combine the earlier bandwidth values and the buffer level, the result might significantly improve. These things motivate us to come up with the idea of bandwidth prediction based on a number of factors such as the buffer level and bandwidth.

3. Mathematics Modeling and System Propose

In network communication, we must consider many parameters to ensure the stability of network conditions as it is susceptible to fluctuations. Within a certain amount of time, we collect a set of the parameters, such as a communication speed measurement, a number of packages transmitted. We use two mathematic techniques in this research. First, we use a moving average to estimate the average bandwidth of a client. More specifically, moving average first averages the initial data set to get the first element. It then calculates the next element based on the last averaged data. The process is continuously repeated over the time until we stop the calculation. Secondly, we use fuzzy logic [5] to infer whether bandwidth is increased or decreased. We then predict a bandwidth value of the client.

In adaptive streaming, we consider two factors: 1) a bandwidth value of a client; 2) an interval time between two different segment requests. Suppose that at the given time t, the bandwidth value of the client is b_t. We name bandwidth requirement for high, medium and low-resolution streaming, is b_h, b_m, b_l. If $b_t > b_h$, a client can retrieve the high resolution streaming.

$$\mathrm{W}_{\mathrm{n+1}} = W_n + \frac{b_{n+1} - W_n}{n+1}. \tag{1}$$

$$\mathrm{U}_{\mathrm{n+1}} = U_n + \frac{t_{n+1} - U_n}{n+1}. \tag{2}$$

$$R_{n+1} = R_n + \frac{r_{n+1} - R_n}{n+1}. \tag{3}$$

Given a time series $\{t_0, t_1, t_2, \dots, t_n\}$, bandwidth values at those time is $\{b_0, b_1, b_2, \dots, b_n\}$, response resolutions $\{r_0, r_1, r_2, \dots, r_n\}$, then we calculate a moving average of the bandwidth W_{n+1}, the interval time U_{n+1}, and the resquest resolution R_{n+1} by (1), (2) and (3).

In the fuzzy logic modeling, we define seven linguistic sets are allowable to describe the variables of one's subjective judgment: 1) Negative High Difference – NHD; 2) Negative Medium Difference – NMD; 3) Negative Low Difference – NLD; 4) Difference – D; 5) Positive Low Difference – PLD; 6) Positive Medium Difference – PMD; 7) Positive High Difference – PHD. To specify the difference, we first normalize the transmit rate of each resolution and use a trace-back technique. The trace-back is based on the logic of the current state of an event relates to its previous states or occurrence of an event is caused by previous states.

Table 1. Fuzzy inference rules, the left outer column is bandwidth linguistics

B/T	**NHD**	**NMD**	**NLD**	**D**	**PLD**	**PMD**	**PHD**
NHD	PLD	D	NLD	NLD	NLD	NHD	NHD
NMD	PLD	D	NLD	NMD	NMD	NHD	NHD
NLD	PLD	D	D	NLD	NMD	NMD	NHD
D	PLD	D	D	NLD	NLD	NMD	NHD
PLD	PLD	PMD	D	D	NLD	NMD	NMD
PMD	PMD	PMD	PLD	D	D	NLD	NLD
PHD	PHD	PHD	PMD	PLD	PLD	D	D

We measure the client's bandwidth each time when it creates a request segment to the server. Each request is called a step. Suppose that we trace-back within k steps ($k \in N^+$), We compare a subtraction between the moving average and a bandwidth/time value at present, with that subtraction at earlier times. It results in a value that can be greater or less than one. Therefore, we normalize the value by comparing with the minimum different and maximum different bandwidth/time. The normalized result is in a range between -1 and +1. The degree of difference is defined in (4) and (5).

$$\mathrm{d_b} = \frac{1}{k}\sum_{i=n+1-k}^{n}\left(\frac{|b_i-W_i|}{b_{n+1}-W_{n+1}} \times \frac{\min\limits_{l=n+1-k,n}(|b_l-W_l|)}{\max\limits_{l=n+1-k,n}|b_l-W_l|}\right). \tag{4}$$

$$\mathrm{d_t} = \frac{1}{k}\sum_{i=n+1-k}^{n}\left(\frac{|t_i-U_i|}{t_{n+1}-U_{n+1}} \times \frac{\min\limits_{l=n+1-k,n}(|t_l-U_l|)}{\max\limits_{l=n+1-k,n}|t_l-U_l|}\right). \tag{5}$$

The fuzzy system has three main calculation processes. First, we form fuzzy input values of the system as (4) and (5), and this process is referred to as fuzzification. Then user or expert's knowledge is used as "*if and then*" rules to infer a fuzzy output, and this process is called inference. Finally, the system defuses the input value to calculate the output system, and this process is also known as defuzification.

In this research, we use Mamdani fuzzy model [6]. A triangular fuzzy number represents each linguistic value. It has a triangle-shaped membership function that can be viewed as a probability distribution. The fuzzy system has two input parameters that are a bandwidth value of a client and downloading time of each segment. The output is a bandwidth prediction value. In the system, we have total 49 rules. We use the rules to infer output from the given inputs as shown in Table 1 where B/T represents the bandwidth and the time interval.

We describe the table as, if the bandwidth is Negative High Difference (NHD) and the interval time is Positive Low Difference (PLD), then the new request segment is High Difference (HD), or the client requests higher resolution streaming compared to the previous segment resolutions. More specifically, we describe HD as, the bandwidth/interval time at present highly increases compared to previous values. In opposing, VSD means that the bandwidth/interval time at present dramatically decreases compared to the previous values. With the same idea as mentioned above, we can also predict bandwidth at the next client request.

$$r_{n+1} = R_n + \frac{1}{k \times d_r}\sum_{i=n-k}^{n}\left(|r_i - R_i| \times \frac{\min\limits_{l=n-k,n}(|r_l-R_l|)}{\max\limits_{l=n-k,n}|r_l-R_l|}\right). \tag{6}$$

In the above fuzzification process, we formed an input parameter by comparing the difference between the present value and the previous value within k steps. Therefore, the defuzification process must inverse the formulation. Suppose that, d_r is a fuzzy output of the system at the present time. r_{n+1} is the real resolution output after the defuzification process. It is calculated by (6), where R_l is calculated by (3).

4. Experimentation and Evaluation

In the experiment, we set up a system which has four components. First, a playback mobile client on Android-LibDash-based [8] where we implement the proposed method. Second, FFmpeg software is to encode a source file into different qualities. Third, MP4Box software is to divide those files into segments and create a manifest file that links to the segment files. Fourth, a Node.js [9] server stores an uploaded file from a user; then it segments the file into small segments.

We ran the system in two scenarios and compared the proposed method with the CrystallBall Algorithm [3]. First, we connected the mobile user with a Wi-Fi 5G to have a video streaming with the server. We then went away and moved closer to the Wi-Fi to have a bandwidth fluctuation. Second, a mobile user has subscribed to 4G service in South Korea. It also moved around an urban area and a remote area. In the MPD file, we have the bandwidth requirement for 4K video at 2160p is above 18367178 bps/s. The bandwidth requirement for 720p video is above 1381427 bps/s, and the bandwidth requirement for lowest video resolution at 480p is above 730448 bps/s.

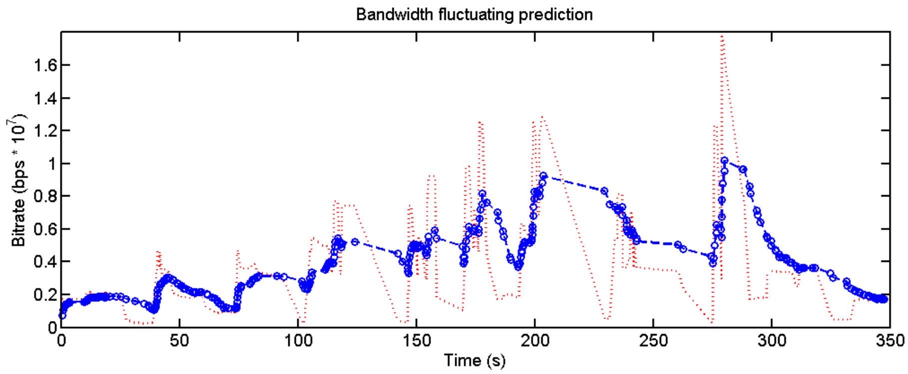

Figure 1. Bandwidth fluctuation and result of bandwidth prediction.

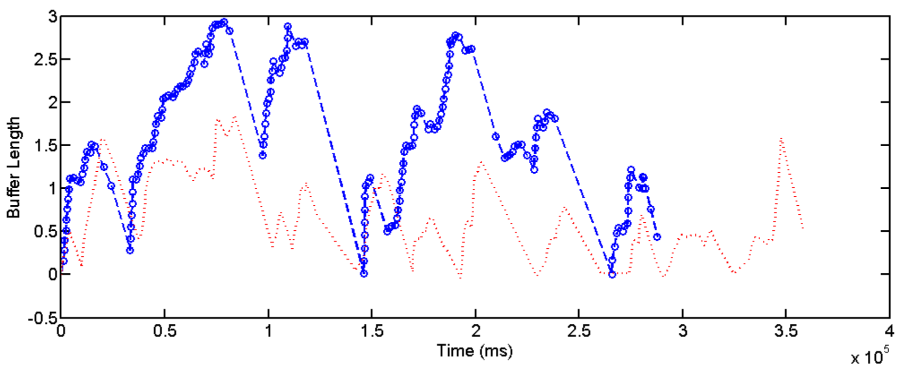

Figure 2. Variation of buffer length.

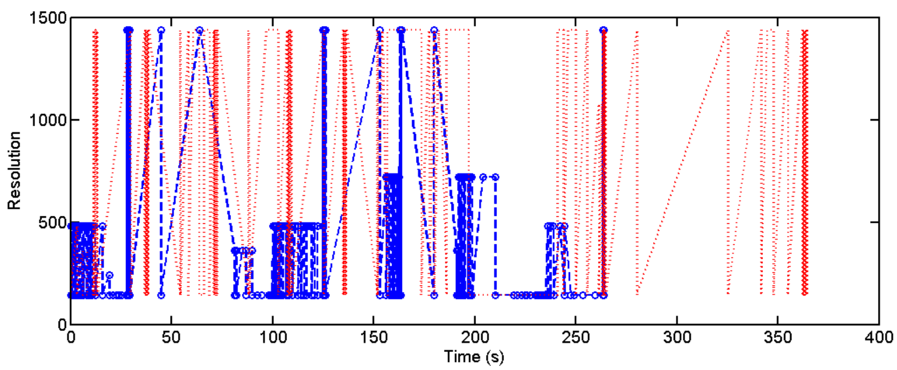

Figure 3. Variation of video representations in adaptive streaming.

In Figure 1, the red line is bandwidth estimation of the CrystallBall. The blue line depicts the bandwidth estimation of our proposal method. In this experiment, we only anticipated the average bandwidth one step ahead. We can predict the value of bandwidth by more than one step. The values will be predicted based on the predicted values. As shown in Figure 1, the green line does not vary much compared to the CrytallBall shown in the red line, and it closes to the average bandwidth. In addition, in Figure 2, the green line depicts buffer length of the proposed method. It is higher than the red line which is the buffer length of compared method. The red line even reaches zero value more than the blue line. Furthermore, in Figure 3, our approach has also reduced the variability of video representations compared to the CrystallBall method.

5. Conclusion

In this paper, we presented the fuzzy-based method to reduce the mobile video quality fluctuation of the high-quality video-streaming in adaptive streaming over HTTP. We exploited two mathematic techniques, which are moving average and fuzzy logic, to anticipate bandwidth values. The experiment results showed that our proposal reduced the fluctuation of bandwidth compared to the previous research. For future work, we plan to extend the method with some additional parameters such as buffer length, crowdsourcing. Furthermore, we intend to investigate more in the ultra-high definition multimedia transmission for a mobile device over the network by cooperating the proposed method with the most recent advantage video compression techniques.

ACKNOWLEDGEMENTS

This work was supported by 'The Cross-Ministry Giga KOREA Project' grant from the Ministry of Science, ICT and Future Planning, Rep. of Korea(GK16P0100, Development of Tele Experience Service SW Platform based on Giga Media).

References

[1] Stockhammer T. Dynamic adaptive streaming over HTTP--: standards and design principles. Proceedings of the second annual ACM conference on Multimedia systems, (**2011**) Feb 23-25; San Jose, CA, USA.
[2] Toni L, Aparicio-Pardo R, Simon G, Blanc A, Frossard P. Optimal set of video representations in adaptive streaming. Proceedings of the 5th ACM Multimedia Systems (**2014**) March 19; Singapore.
[3] Mangla T, Theera-Ampornpunt N, Ammar M, Zegura E, Bagchi S. Video through a crystal ball: effect of bandwidth prediction quality on adaptive streaming in mobile environments. Proceedings of the 8th International Workshop on Mobile Video, (**2016**) May 10-13; Klagenfurt, Austria.
[4] Dubin R, Dvir A, Pele O, Hadar O, Katz I, Mashiach O, Adaptation logic for HTTP dynamic adaptive streaming using geo-predictive crowdsourcing for mobile users. Multimedia Systems. (**2016**)
[5] Pugachev VS, Probability theory and mathematical statistics for engineers, Elsevier (**2014**)
[6] Gottwald S, Fuzzy sets and fuzzy logic: The foundations of application—from a mathematical point of view, Springer-Verlag, (**2013**)
[7] Gerla G, Fuzzy logic: mathematical tools for approximate reasoning, Springer Science & Business Media (**2013**)
[8] Mueller C, Lederer S, Poecher J, Timmerer C. Demo paper: Libdash-an open source software library for the mpeg-dash standard. Multimedia and Expo Workshops (ICMEW), 2013 IEEE International Conference, (**2013**) July 15-19; San Jose, California, USA
[9] Cantelon M, Harter M, Holowaychuk TJ, Rajlich N, Node. js in Action, Manning (**2014**)

Linh Van Ma

Linh Van Ma is currently a M.S candidate at the Smart Mobile and Media Computing Laboratory, School of Electronics and Computer Engineering, Chonnam National University, South Korea.

Sanghyun Park

Sanghyun Park received his B.S. Degree in Computer and Information from the Korea Nazarene University. He is now studying his Doctoral Degree in Electronics and Computer Engineering at Chonnam National University.

Jong-Hyun Jang

Jong-Hyun Jang received his Ph.D. degree in computer science and engineering from Hankuk University of Foreign Studies, Yongin, Rep. of Korea, in 2004. Since 1994, he has been with ETRI, where he is currently a principal member of the research staff as well as a team leader of the Real and Emotional Sense Platform Research Section.

Jiseung Nam

Jiseung Nam received his Ph.D. degree in computer science and engineering from the University of Arizona, the USA in 1992. Currently, he is a professor in Chonnam National University, Gwangju, S.Korea.

JinSul Kim

JinSul Kim received the B.S. Degree in computer science from the University of Utah, Salt Lake City, Utah, the USA in 2001, and the M.S. and Ph.D. degrees in digital media engineering, Department of Information and Communications of the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea in 2005 and 2008. Currently, he is a professor in Chonnam National University, Gwangju, Korea.