Signal, Image and Video Processing (2020) 14:1107–1114
https://doi.org/10.1007/s11760-020-01646-y

ORIGINAL PAPER

SSIM-based adaptation for DASH with SVC in mobile networks

Mehmet Çalı[1] · Nükhet Özbek[2]

Received: 3 August 2019 / Revised: 4 December 2019 / Accepted: 21 January 2020 / Published online: 7 February 2020

Abstract
Dynamic Adaptive Streaming over HTTP (DASH) depends on adjustment of the quality of a video stream to the available network conditions. In order to increase Quality of Experience, average video quality should be maximized, while keeping the quality switching frequency at low levels. However, achieving high average quality with low switching frequency in highly fluctuating mobile network conditions is a tricky optimization problem. In order to overcome this problem, dynamic structure of Scalable Video Coding (SVC) is utilized in this paper. Another challenge in the quality adaptation algorithms is to proper assessment of the video quality. Most of the adaptation algorithms takes either bitrate or representation level as the input that is used to evaluate the quality of the video. However, bitrate is not strongly correlated with the quality, as it depends on the content of the video. Likewise, representation quality relationship entirely bound to encoding. In this paper, in order to have a more reliable adaptation input, SSIM is used while representing the quality of the video stream. The proposed adaptation is compared with a successful SVC DASH adaptation algorithm using both subjective and objective tests. As a result, considerably higher scores are achieved in terms of both switching frequency and average quality.

Keywords Dynamic Adaptive Streaming over HTTP (DASH) · Scalable Video Coding (SVC) · SSIM

1 Introduction

Video streaming constitutes the largest part of the internet traffic. According to Cisco, internet video traffic will be 80% of the overall internet traffic in 2021 [1]. Therefore, adaptive streaming algorithms have a significant role in the advancement of the efficient network resource utilization. In this regard, MPEG-DASH [2] is created to standardize the adaptive streaming of multimedia content. In DASH standard, video and audio contents are encoded in different bitrates which are called representation. In order to switch between representations depending on the network conditions, they are divided into a few-second-long time segments.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11760-020-01646-y) contains supplementary material, which is available to authorized users.

✉ Nükhet Özbek
nukhet.ozbek@ege.edu.tr

Mehmet Çalı
mehmetcali@iyte.edu.tr

1 Department of Electrical and Electronics Engineering, Izmir Institute of Technology, Izmir, Turkey

2 Department of Electrical and Electronics Engineering, Ege University, Izmir, Turkey

The information regarding representation, segment and other structures of DASH such as period and adaptation set is provided in the Media Presentation Description (MPD) file. The server stores the MPD file along with the segments of each representation. The client requests MPD file to get the URL of the segments and some data that can be used as an input for the adaptation such as bitrate. The structure of the MPD and the communication between the server and the client are specified by the DASH standard. However, the adaptation algorithm is not specified; hence, optimization of the adaptation process has received considerable attention recently.

While there are many studies regarding DASH using Advanced Video Coding (AVC), the amount of work focuses on DASH using Scalable Video Coding (SVC) [3] is limited. In SVC, videos are encoded in a layered structure. Each layer is used to enhance the information stored in the previous layers. SVC usage brings some advantages to DASH implementations. SVC utilizes the information stored in the lower layers while creating the higher layers. Therefore, the total storage space for all the representations of SVC-based DASH is less than the storage space of the AVC-based DASH [4]. Furthermore, the quality of the segments inside the buffer can be increased by streaming upper layers of these segments in the SVC-based DASH. As a result, SVC enables adaptation algorithm of DASH to respond more dynamically to the bandwidth fluctuations [5]. Considering these advantages, the adaptation algorithm is designed for an SVC dataset in the proposed work.

The major goal of the rate adaptation algorithms is to achieve the maximum Quality of Experience (QoE) for a given network condition. QoE of video stream can be expressed in four main criteria which are stalling duration, average video quality, quality variation and video start-up time [6]. Stallings and video start-up duration are the criteria that can be controlled by adjusting the desired buffer length. After setting an appropriate desired buffer length, adaptation algorithms focus on maximizing average video quality without getting a disturbing quality variation. At this point, the quality of each video chunk is required to control the average quality and quality variation. In most of the studies, bitrate of the representations or merely representation id is used to express the quality of the video chunks during the adaptation process. However, both methods are not reliable and consistent for assessment of the video quality, since they strongly bound to how successfully the input video is encoded and layered. Therefore, Structural Similarity Index (SSIM) [7], which is one of the most successful and prevalent video quality assessment metrics, is used as an input of the proposed adaptation algorithm.

The proposed work focuses on achieving optimum QoE in mobile network condition, since the mobile data traffic has a large growth [1] and it requires more dynamic adaptation logic to stabilize its bandwidth fluctuations. In this regard, a novel layer and segment prioritization algorithm is proposed as well as a buffer quality-based desired buffer length to minimize quality variations. By using the SSIM as an adaptation input and designing a convenient adaptation algorithm to the features of SVC, high average quality and low quality variance are achieved in this work.

The organization of this paper is presented as follows. In Sect. 2 the related adaptation algorithms in the literature are explained. In Sect. 3, the proposed adaptation algorithm is presented with the structures used to implement it. Finally, the experiment setup and performance evaluation parts are provided in Sect. 4.

2 Related works

In order to achieve the optimum streaming performance, there are many studies which focus on quality adaptation algorithm related to DASH and other streaming standards. These video quality adaptation algorithms can be investigated in three categories which are bandwidth-based, buffer-based and both bandwidth- and buffer-based algorithms.

In the adaptation algorithms based on bandwidth only, representation with the most suitable bitrate to the estimated bandwidth is requested. As an example of the bandwidth-based algorithms, [8] uses bandwidth utilization as an adaptation input. The amount of representation level shift is determined according to the difference between the reference bandwidth utilization and the actual bandwidth utilization. In this process, they use fuzzy logic controller to manipulate continuous utilization data to obtain discrete representation level change output. Another bandwidth-based study is proposed in [9]. They also use SSIM as video streaming input; however, their solution does not include DASH standard or SVC. Instead, they use a simple control mechanism at the server side to make sure all the clients receive a video quality above an SSIM threshold. The adaptation algorithm also tries to lower the bitrate of the current representation until it reaches the measured bandwidth.

In the buffer-based algorithms, the strategy to maximize QoE is to keep the actual buffer size in desired buffer thresholds during streaming. If the actual buffer size rises, the adaptation logic increases the representation level to drop buffer size to the desired level again. In the opposite state, the level of representation is dropped to prevent any stalling that stem from buffer underflow. As a buffer-based adaptation algorithm, [10] tries to decrease the video quality switching frequency and prevent rapid quality drops. To achieve these goals, they propose separate algorithms for both of them. In the first algorithm, DASH segments are united into segment groups, and the quality switching is enabled only between the segment groups. Furthermore, two buffer thresholds are specified to decide a quality drop or increase. The next algorithm controls whether the first algorithm decision leads to a rapid quality drop or not. If the first algorithm estimates a quality drop, the action of the first algorithm is canceled. In [11], an algorithm named Bandwidth Independent Efficient Buffering (BIEB) is proposed. Their SVC-based algorithm consists of three phases, namely steady phase, growing phase and quality increase phase. The steady phase lasts until all the segments in the buffer threshold are streamed. The buffer threshold is specified as a function of index and bitrate of SVC layers. They increase the buffer threshold by a specific amount before the quality increase. Therefore, a possible buffer underflow is tried to be prevented after a quality increase. After the quality increase, the algorithm returns to steady phase. BIEB achieves successful results in terms of average quality and bandwidth utilization compared to [12], [13] and [14] according to their test results. Since it is a successful implementation of SVC-based DASH algorithm, the proposed work is compared with BIEB in the performance evaluation section. As adaptation algorithm is bound to both buffer and bandwidth inputs, [15–18] can be illustrated. In [15], they use motion activity and scene change information as content-related adaptation inputs. If the motion information of a segment is high, representation level with closer bitrate to the measured bandwidth is chosen. Furthermore,

more buffer size is allocated to segments that have higher motion intensity. They also have a control mechanism to prevent quality changes between the segments that belong to the same scene. Hence, the distinct quality changes are tried to minimized as much as possible. In the work proposed in [16], a state vector is defined with five elements which are current buffer status, the change in the buffer status, current bandwidth measurement, bitrate of the last N segments and a Boolean variable related to downloaded segment bitrates. QoE is represented with a defined reward function measured from the state vectors. As a result, the adaptation output that has the highest probability to maximize the reward function is computed using Markov chain. [17] and [18] are two recent studies that use SVC-based adaptation. Both studies determine the quality increase of already downloaded segments according to estimated bandwidth and available buffer length with heuristic approaches. In [17], quality increase occurs when the bitrate of the chunk is greater than estimated bandwidth or the buffer reaches a certain limit, whereas [18] uses bandwidth for adjustment of the quality only if the buffer length is between certain thresholds. Both approaches depend heavily on the accuracy of the bandwidth estimation and convenient for videos encoded with constant bitrate.

Few of the adaptation algorithms use content-related objective metrics as their inputs. Among the described studies, [9] and [15] uses such metrics. However, the design in the [9] is for the network with smaller scale, since the adaptation process depends on the control mechanism in the server. Although the method proposed in [15] uses DASH structure, it uses only the motion and screen change information in the adaptation process. The effects of intraframe features such as distortions or data redundancy are not considered in their work. In this paper, SSIM is utilized to properly assess the contribution of each segment to overall stream quality. Using an objective metric that is correlated with Human Vision System (HVS), an adaptation algorithm which uses SVC dataset is designed.

3 Methodology

In this part, structures of designed adaptation algorithm are described. First of all, SVC dataset is encoded with JSVM (Joint Scalable Video Model) [19] using quality scalability option. Test videos are encoded in five layers whose average bitrates are in between 120 kbit/s and 1 Mbit/s. Two-second-long segments are used to enable flexible quality switches in mobile network conditions. In order to use SSIM in the adaptation algorithm, SSIM of each video chunk is inserted into MPD file of the dataset as [15] inserts motion activity in their work. Additionally, an MPD parser is designed using QT framework to extract the SSIM data as well as other related information used is streaming such as segment URLs. Libcurl [20] is used to send HTTP requests.

In AVC-based adaptations, video segments are streamed in increasing order, and adaptation mechanism only decides the representation id of the queued segment. In other words, segments are streamed starting from the lowest id, and only the representations that they belong are specified by the adaptation algorithm. On the other hand, representations correspond to a number of layers in SVC-based adaptation algorithms. In order to make more appropriate decision to available network condition, only lower layers of the queued segments are streamed first. After reaching a specified buffer length, the higher layers of the buffered segments can be streamed. This utility of SVC enables the option of selecting segments in non-sequential order. Moreover, upper layers of a scalable coded video require all of the lower layers for proper reconstruction. Thereby, the layers of a segment should be streamed in increasing order starting from the lowest id in SVC-based steaming. As a result, instead of representation id, segment id is the main output of the proposed adaptation algorithm. Since the layers are downloaded in increasing order, the next available layer of the segment that is decided by the adaptation algorithm is streamed in each loop.

The proposed adaptation algorithm can be described in two parts, which are adaptation loop and layer and segment selection algorithm. In the adaptation loop, base layer of the segments is streamed sequentially to prevent any stalling as a first step. Then, layer and segment selection algorithm is called to obtain the segment whose layer increment contributes the most to QoE. In the adaptation loop which is shown in Algorithm 1, B_0, B_{des}, D_{s0}, $D_{l,s}$, N_s and $V_{l,s}$ stand for current base layer buffer length, desired base layer buffer threshold, downloaded last segment index for base layer, the list of downloaded video chunks, number of segments in the stream and the video chunk with specified layer, and segment indexes, respectively. At first, if the actual buffer length of the base layer is less than B_{des}, base layer of the next segment is downloaded. If B_{des} is already reached, the layer and

```
B_des ⟵ b_min;
while S_play < N_s do
    if B_0 < B_des then
        s ⟵ D_s0 + 1;
        l ⟵ 0;
        download V_l,s;
        D_s0 ⟵ s
    else
        get l, s pair according to Algorithm 2;
        download V_l,s;
        append (l, s) to D_l,s;
    end
end
```

Algorithm 1: Adaptation loop

segment selection algorithm is called to get the video chunk to be streamed among enhancement layers. The adaptation loop runs until playback segment (S_{play}) reaches maximum segment index.

$$B_{\text{des}} = \frac{b_{\min}}{T_s} + \frac{(b_{\max} - b_{\min})(Q_{\text{buf}} - Q_{\text{base}})}{(Q_{\max} - Q_{\text{base}})T_s} \tag{1}$$

Desired buffer threshold is calculated according to Eq. 1. In this equation, $b_{\min}$ and $b_{\max}$ are constants that keep the specified buffer length in certain limits. Q_{buf} represents the average quality of buffered video chunks where quality of single chunk is calculated according to Eq. 2.

$$Q_{i,j} = c_1 ssim_{i,j} + i \tag{2}$$

Q_{base} and $Q_{\max}$ indicate the lower and upper thresholds of Q_{buf} as given in Eqs. 3 and 4. T_s is the duration of segments in seconds. Equation 1 is created to establish linear relationship between desired buffer length and buffer quality. When the average quality of the buffered video chunks decreases, buffer length decreases as well to stabilize quality changes. In the opposite condition, buffer length is increased so that higher amount of buffer length can be utilized during possible future bandwidth drops. In this relation, the minimum and maximum buffer lengths ($b_{\min}$ and $b_{\max}$) are used to prevent buffer underflow and overflow, respectively.

$$Q_{\text{base}} = c_1 \sum_{j}^{N_s} \frac{ssim_{0,j}}{N_s} \tag{3}$$

$$Q_{\max} = N_l - 1 + c_1 \tag{4}$$

Quality of each chunk is assessed using SSIM and layer index. The constant c_1 specifies the effect of SSIM in the assessment of input video chunks, while N_l stands for the number of layers. This constant c_1 is selected as 2 for a dataset encoded with constant QP. In constant bitrate videos where layer index performs worse in representing the quality, c_1 should be selected higher. The lower and upper limits of Q_{buf} are 0 and 6 for $N_l = 5$ and $c_1 = 2$ according to Eq. 3 and Eq. 4. However, the linear relationship starts from Q_{base} instead of $Q_{\min}$. This is preferred since SSIM of an encoded video segment rarely gets values below 0.7 as observed in the designed dataset and other DASH datasets such as [21]. Likewise, Q_{buf} rarely drops below 1.4 for $c_1 = 2$. Therefore, average base layer quality of the video (Q_{base}) is selected as the minimum threshold of Q_{buf}.

In Algorithm 2, $P_{i,j}$ and $P_{\max}$ stand for priority factor of the specified video chunk and maximum priority factor of the chunk that can be downloaded in the adaptation loop, respectively. This algorithm specifies the layer and segment ids of the video chunk that will be streamed in the adaptation loop. The layer and segment ids of the video chunk with the highest priority factor are selected. The algorithm is only called for enhancement layers, since the base layer is streamed sequentially for each segment. The priority factor comparison is applied to layers with the lowest possible id which are not streamed yet. Moreover, it is applied to segments inside the buffer whose id is higher than playback segment with a margin of S_{margin}. S_{margin} allows SVC multiplexer and video decoder to have time for their processes. In other words, the algorithm selects among the video chunks that requires sufficient time for layer multiplexing and decoding.

Layer and segment selection algorithm aims at selecting the video chunks that will contribute to the average quality at most without causing excessive quality fluctuations. In this respect, priority factor is calculated according to Eq. 5.

$$P_{l,s} = ssim_{l,s} - ssim_{l-1,s} + \frac{c_2}{l} \tag{5}$$

```
s ⟵ 0, l ⟵ 0, c2 ⟵ 0.2;
P_{l,s} ⟵ 0, P_max ⟵ 0, P_margin ⟵ 0.001;
for j ← S_play + S_margin to D_s0 do
    for i ← 0 to N_l − 1 do
        if V_{i,j} is not downloaded then
            P_{i,j} is computed using Eq. 5;
            if P_{i,j} > P_max + P_margin then
                P_max ⟵ P_{i,j};
                l ⟵ i;
                s ⟵ j;
            end
            break;
        end
    end
end
```

Algorithm 2: Layer and segment selection

In this equation, both the layer index and SSIM difference are used for prioritization. The video chunks with lower layer index have higher priority to decrease quality fluctuations. Since the SSIM is capped at 1, when the quality increases, the SSIM difference between the successive layers decreases. Therefore, both SSIM difference and layer index in the $P_{i,j}$ lead downloading the segments with lower quality first. In other words, this relation prevents any segment getting much higher quality than the neighboring segments. Prioritization of chunks with higher SSIM difference from the previous layer also enables selection of the chunk that contributes the overall quality more than the others. The constant c_2 represents the success of layer index in picturing the representation quality. For a well-layered, constant QP input video, c_2 constant is selected as 0.2. On the other hand, it can be selected even lower for datasets with constant bitrate videos whose layers have high quality variation. In Fig. 1, a streaming state is given to illustrate the prioritization of the enhancement layers. In this state, base layer buffer threshold is reached, so

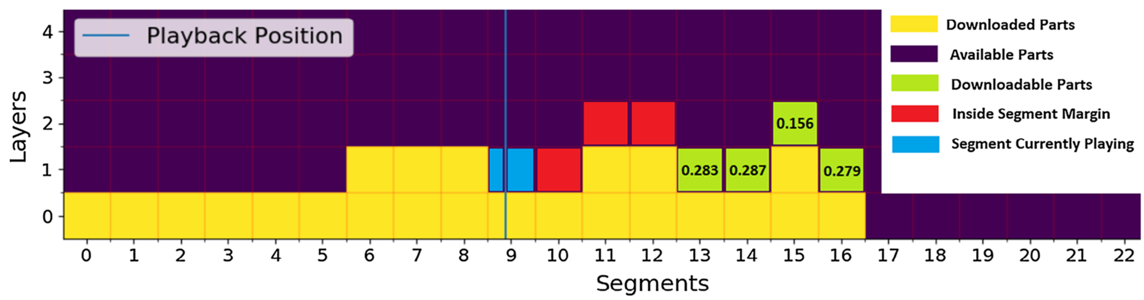

Fig. 1 Prioritization of video chunks in a streaming state

enhancement layer of the selected segment will be streamed. As described in Algorithm 2, the nested for loop specifies only the downloadable chunks shown in the streaming state. Using the SSIM differences and the layer indexes, priority factors are calculated and shown in the downloadable parts of Fig. 1. Among these chunks, the maximum priority factor corresponds to layer, segment pair of [1,14].

4 Performance evaluation

The performance of the designed adaptation algorithm is compared with BIEB adaptation algorithm which is proposed [11]. The minimum desired buffer size of BIEB is selected as 16 s which is the same value chosen in their experiment. For the proposed method, the constants of $b_{\min}$, $b_{\max}$, c_1 and c_2 are selected as 14, 32, 2 and 0.2, respectively. The comparison is performed using objective and subjective methods. The experiment is conducted for Big Buck Bunny [22], Sintel [23] and Tears of Steel [24] source videos which are encoded with 2-s-long segments. The algorithms are tested using two different network scenarios which are described in the following section.

4.1 Experiment setup

The experiment setup consists of mainly three parts which are client, server and network traffic shaping tool. Both client and server run on Ubuntu 18.04. Apache 2.4 is installed on the server. For network traffic shaping, Token Bucket Filter (tbf) and Linux Traffic Control command (tc) are used.

Firstly, the designed dataset with 2-s-long segments is downloaded on the Apache server with the corresponding MPD files. Shell scripts are created for emulation of the mobile network conditions using tbf and tc. Two different network waveform scenarios are realized with the shell scripts. Bandwidth waveforms have duration of 100 s. The first waveform has single peek around 900 kbit/s and 1 Mbit/s interval since 1 Mbit/s is the maximum bandwidth limit that can be set using tbf. The second waveform has two peaks at the same interval. The minimum bandwidth values are around 150 kbit/s in both scenarios. The streaming bitrates are also measured for both of the network scenarios and provided with the bandwidth waveforms as shown in Fig. 2.

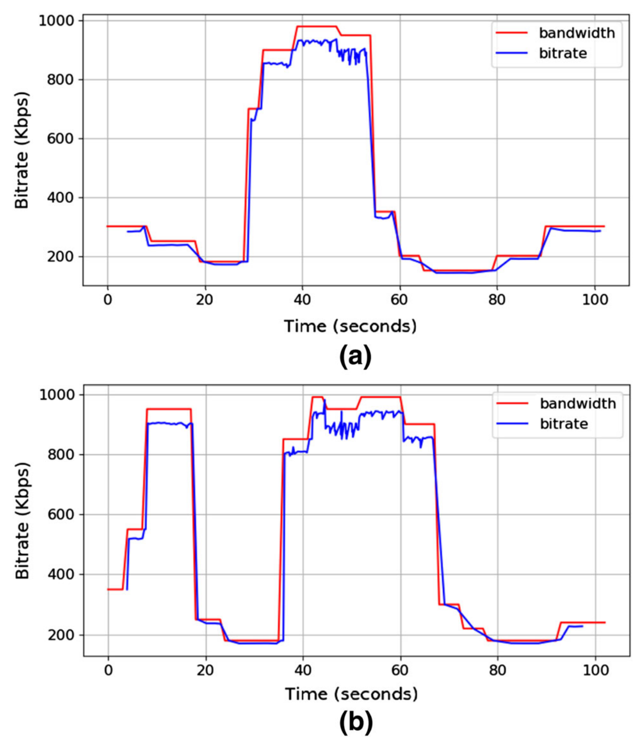

Fig. 2 Measured bitrate and bandwidth graph of Sintel for **a** single-peak and **b** double-peak scenario

4.2 Objective evaluation

In this part, proposed adaptation and BIEB are compared in terms of average quality and degree of quality fluctuations. In order to represent the average quality with an objective method, average SSIM and average PSNR are used. Both quality switching frequency and the degree of quality changes in these shifts affect the QoE negatively. Therefore, the amount of quality shifts is not accurate by itself to represent the effect of quality switches. In this regard, [25] uses variation of bitrate to represent degree of shifts from the

Table 1 Objective test results in terms of SSIM

		Big Buck Bunny		Sintel		Tears of Steel	
		Single peak	Double peak	Single peak	Double peak	Single peak	Double peak
Average SSIM	BIEB	0.8965	0.9054	0.8752	0.8810	0.9096	0.9225
	Proposed	0.9081	0.9262	0.8854	0.8990	0.9221	0.9337
Variance of SSIM	BIEB	0.00519	0.00394	0.00303	0.00374	0.00155	0.00137
	Proposed	0.00291	0.00165	0.00222	0.00157	0.00101	0.00068

Table 2 Objective test results in terms of PSNR

		Big Buck Bunny		Sintel		Tears of steel	
		Single peak	Double peak	Single peak	Double peak	Single peak	Double peak
Average PSNR	BIEB	32.88	33.49	31.69	32.05	32.86	33.91
	Proposed	33.44	34.33	32.13	32.93	33.71	34.65
Variance of PSNR	BIEB	13.69	16.54	11.37	12.16	10.73	8.56
	Proposed	11.24	10.27	10.11	9.09	8.81	6.93

average bitrate. Being quality assessment metrics, SSIM and PSNR are more reliable to indicate video quality compared to bitrate. As a result, variance of SSIM and PSNR is used to represent the contribution of both quality switching rate and the effect of each shift. The objective test results for both algorithms are provided in Tables 1 and 2. The proposed algorithm has higher average quality and lower quality variation for all of the test conditions.

4.3 Subjective evaluation

As subjective evaluation, paired comparison method is used. In the paired comparison method, the test data are provided in pairs for evaluation. Subjects present their preference among each pair. In the designed subjective test, three possible preferences are presented to subjects which are 'left is better,' 'right is better' and 'same.'

The subjective test is conducted according to standards presented by ITU-T recommendation P.910 [26]. According to ITU recommendation, the pairs can be provided in sequential order or simultaneously. Simultaneous presentation (SP) has some advantages over sequential presentation. Firstly, SP significantly reduces test duration of a pair. Therefore, longer test sequences can be used in SP to increase the reliability. Another benefit of SP is that the subjects can distinguish the differences between the pairs more easily. However, SP is recommended by ITU if the input video resolutions are low. Since the created dataset consists of only videos that have 640 × 360 resolution, SP is used in the paired comparison.

The subjective test algorithm is coded using Python. OpenCV is used to display the test sequences and the voting screen. As suggested in ITU standard, the pairs are presented for approximately 10 s before the voting screen. The start and end points of these intervals are selected using crossing points of SSIM waveforms that belong to proposed method and BIEB. This configuration leads only one algorithm to perform better at any time during the selected interval. As a result, the decision process becomes easier for the subjects. The pairs are presented in the horizontally positioned slots. The selection of proposed algorithm and BIEB to right or left slot changes randomly between each interval in the sequence. The placement of stimuli also changes randomly between each test. On the other hand, the intervals in the video sequence are presented in the streaming order. By presenting the sequence in the steaming order, abrupt context changes which can cause focusing problems are prevented. The background color of both video sequence and voting screen is gray as suggested in ITU-T recommendation.

The voting screen has 'left is better,' 'right is better,' 'same,' 'replay' and 'next' buttons. After the selection of the subject, the preference is appended to a log file, and the next video interval starts playing. The experiment is conducted with the help of 20 subjects with an average age of 28. Twelve of the subjects are male, and 8 of them are female. Twelve of the subjects have at least MS degree, while 8 of the subjects have undergraduate degree.

In order to describe the scores of the paired comparison tests, the Bradley–Terry model [27] is used commonly. In the Bradley–Terry model, if two stimuli and two options (better and worse) are used, their scores (u_1 and u_2) can be calculated using probability of choosing stimuli i over j (P_{ij}) and probability of choosing stimuli j over i (P_{ji}) as shown in Eq. 6.

$$u_1 = \frac{P_{12}}{P_{12} + P_{21}}, \quad u_2 = \frac{P_{21}}{P_{12} + P_{21}} \tag{6}$$

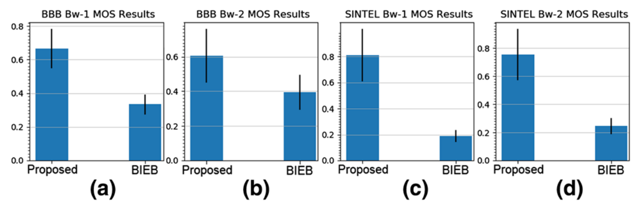

Fig. 3 Subjective scores of **a** BBB single-peak, **b** BBB double-peak, **c** Sintel single-peak and **d** Sintel double-peak scenarios

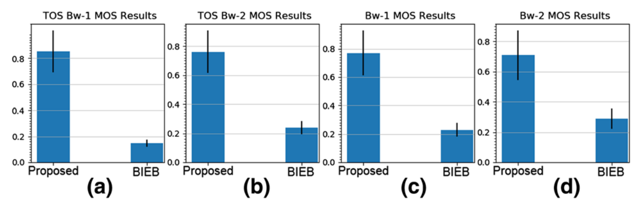

Fig. 4 Subjective scores of **a** TOS single-peak, **b** TOS double-peak, **c** overall single-peak and **d** overall double-peak scenarios

There are many methods which utilizes the Bradley–Terry model to evaluate the results of paired comparison with three options. In this paper, the method which is proposed in [28] is used to calculate the scores. This method calculates the individual scores directly from the Bradley–Terry model and adds confidence intervals using third option which is 'same.' The upper and lower limits of the confidence interval that belongs to the score of stimulus i can be calculated using Eq. 7 and Eq. 8.

$$l_{upp} = \frac{P_{12} + P_{same}}{P_{12} + P_{21} + P_{same}} - u1 \quad (7)$$

$$l_{low} = u1 - \frac{P_{12}}{P_{12} + P_{21} + P_{same}} \quad (8)$$

The subjective test scores of BIEB and the proposed method are demonstrated in Figs. 3 and 4. According to subjective scores, the proposed method performs better than compared algorithm with a high margin.

5 Conclusion

In this paper, adaptive streaming of SVC videos is implemented according to DASH standard. An adaptation algorithm is designed such that it enables utilization of favorable characteristics of SVC as much as possible. Moreover, SSIM of individual video chunks is used as adaptation algorithm input to make the adaptation process more reliable. Therefore, the proposed method has an advantage over the previous works in terms of ability to work stably for various datasets. The proposed work is compared with a state-of-the-art SVC DASH adaptation algorithm using different input videos and bandwidth scenarios. According to objective and subjective evaluation, the proposed algorithm performs better than the compared algorithm in terms of average video quality and quality variation.

References

1. Cisco, Cisco visual networking index: forecast and methodology 2016–2021 (2016)
2. Stockhammer, T.: Dynamic adaptive streaming over HTTP: standards and design principles. In: Proceedings of the Second Annual ACM Conference On Multimedia Systems, pp. 133–144 (2016)
3. Schwarz, H., Marpe, D., Wiegand, T.: Overview of the scalable video coding extension of the H. 264/AVC standard. IEEE Transactions on circuits and systems for video technology, 17(9), pp. 1103-1120 (2007)
4. Kalva, H., Adzic, V., Furht, B.: Comparing MPEG AVC and SVC for adaptive HTTP streaming. In: 2012 IEEE International Conference on Consumer Electronics (ICCE), pp. 158-159. IEEE (2012)
5. Zhao, M., Gong, X., Liang, J., Wang, W., Que, X., Cheng, S.: QoE-driven cross-layer optimization for wireless dynamic adaptive streaming of scalable videos over HTTP. IEEE Trans. Circuits Syst. Video Technol. **25**(3), 451–465 (2015)
6. Kua, J., Armitage, G., Branch, P.: A survey of rate adaptation techniques for dynamic adaptive streaming over HTTP. IEEE Commun. Surv. Tutor. **19**(3), 1842–1866 (2017)
7. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
8. Xiong, P., Shen, J., Wang, Q., Jayasinghe, D., Li, J., Pu, C.: NBS: A network-bandwidth-aware streaming version switcher for mobile streaming applications under fuzzy logic control. In: 2012 IEEE First International Conference on Mobile Services, pp. 48-55. IEEE (2012)
9. Chiariotti, F., Pielli, C., Zanella, A., Zorzi, M.: QoE-aware Video Rate Adaptation algorithms in multi-user IEEE 802.11 wireless networks. In: 2015 IEEE International Conference on Communications (ICC), pp. 6116-6121 (2015)
10. Mori, S., Bandai, M.: QoE-aware quality selection method for adaptive video streaming with scalable video coding. In: 2018 IEEE International Conference on Consumer Electronics (ICCE), pp. 1-4. IEEE (2018)
11. Sieber, C., Hoßfeld, T., Zinner, T., Tran-Gia, P., Timmerer, C.: Implementation and User-centric comparison of a novel adaptation logic for DASH with SVC. In: 2013 IFIP/IEEE International Symposium on Integerated Network Management (IM 2013), pp. 1318-1323. IEEE (2013)
12. Miller, K., Quacchio, E., Gennari, G., Wolisz, A.: Adaptation algorithm for adaptive streaming over HTTP. In: 2012 19th International Packet Video Workshop (PV), pp. 173–178. IEEE (2012)
13. Muller, C., Lederer, S., Timmerer, C.: An evaluation of dynamic adaptive streaming over HTTP in vehicular environments. In: Proceedings of the 4th Workshop on Mobile Video, pp. 37-42. ACM (2012)
14. Oechsner, S., Zinner, T., Prokopetz, J.,Hoßfeld, T.: Supporting scalable video codecs in a P2Pvideo-on-demand streaming system. In: Proceedings of 21st ITC SS Multimedia Application–Traffic, Perform. QoE, pp. 48–53 (2010)
15. Hu, S., Sun, L., Gui, C., Jammeh, E., Mkwawa, I.-H.: Content-aware adaptation scheme for QoE optimized DASH applications. In: Global Communications Conference (GLOBECOM), pp. 1336–1341. IEEE (2014)

16. Zhou, C., Lin, C.-W., Guo, Z.: mDASH: a Markov decision-based rate adaptation approach for dynamic HTTP streaming. IEEE Trans. Multimed. **18**(4), 738–751 (2016)
17. Ozcan, S. G., Kivilcim, T., Cetinkaya, C., Sayit, M.: Rate adaptation algorithm with backward quality increasing property for SVC-DASH. In: 2017 IEEE 7th International Conference on Consumer Electronics-Berlin (ICCE-Berlin), pp. 24-28. IEEE (2017)
18. Lekharu, A., Kumar, S., Sur, A., Sarkar, A.: A qoe aware svc based client-side video adaptation algorithm for cellular networks. In: Proceedings of the 19th International Conference on Distributed Computing and Networking, pp. 27. ACM (2018)
19. Reichel, J., Schwarz, H., Wien, M.: Joint scalable video model 11 (JSVM 11). Joint Video Team, pp. 23 (2007)
20. Stenberg, D.: Libcurl—the multiprotocol file transfer library. https://curl.haxx.se/libcurl/. Accessed 1 April 2019
21. Kreuzberger, C., Posch, D., Hellwagner, H.: A scalable video coding dataset and toolchain for dynamic adaptive streaming over HTTP. In: Proceedings of the 6th ACM Multimedia Systems Conference, pp. 213-218. ACM (2015)
22. Blender Foundation: Big Buck Bunny. https://peach.bl-ender.org/. Accessed 1 Feb 2019
23. Blender Foundation: Sintel, the Durian Open Movie Project. https://durian.blender.org/. Accessed 1 Mar 2019
24. Blender Foundation: Tears of Steel. https://mango.bl-ender.org/. Accessed 1 Feb 2019
25. Wang, C., Rizk, A. Zink, M.: SQUAD: A spectrum-based quality adaptation for dynamic adaptive streaming over HTTP. In: Proceedings of the 7th International Conference on Multimedia Systems, pp. 1, ACM (2016)
26. ITU-T Recommendation P.910: Subjective video quality assessment methods for multimedia applica-tions. International telecommunication union (1999)
27. Bradley, R.A., Terry, M.E.: Rank analysis of incomplete block designs: I. Method paired comparisons. Biometrika **18**(3), 324–345 (1952)
28. Lee, J.-S., De Simone, F., Ebrahimi, T.: Subjective quality evaluation via paired comparison: application to scalable video coding. IEEE Trans. Multimed. **13**(5), 882–883 (2011)