# Statistical Characterization of the Chunk Size Distribution in DASH

Cristian J. Vaca-Rubio, Gerardo Gomez, M. Carmen Aguayo-Torres, F. Javier Lopez-Martinez
Dpto. de Ingeniería de Comunicaciones, Universidad de Malaga, 29071, Malaga, Spain
(e-mail: {cjvr,ggomez,aguayo,fjlopezm}@ic.uma.es)

***Abstract*—We present a statistical characterization of the segment size distribution of video streaming services based on dynamic adaptive streaming over HTTP (DASH). We first obtain the empirical distributions from traffic captures of available data sets encoded with different quality and segment size duration. Then, we determine the distributions that provide a better fit to the empirical data. We show that Weibull and truncated logistic distributions are adequate to modeling the chunk size distributions for a wide range of video qualities and segment size duration. Results are used to develop a source model for DASH traffic, implementing a synthetic generator of DASH-like video traces.**

## I. Introduction

Nowadays, multimedia traffic streaming is still under development in comparison with its market potential. Currently, each business platform is a closed system which develops its own transport protocol and content format. In other words, a proper inter-operability among the wide range of different devices and service providers is lacked. In a few years' time, it is expected that Internet video streaming traffic becomes even more predominant, reaching almost the 82% of the global traffic consumed by users by 2021 [1]. One way to provide the desired inter-operability among different servers and devices is enabled by MPEG Dynamic Adaptive Streaming over HTTP (DASH). DASH was developed to adapt the video contents to a wide range of devices and networks capabilities [2].

In a DASH server, video data sets are available in different qualities so that each player can request the bitrate that better suits the current network status [3]. For every available quality, videos are divided into chunks or segments in order to enable an adaptive progressive download of the video, as it is played back at the client. In general, shorter video chunks allow for a better adaptation to network conditions and a more efficient buffer management, at the expense of a worse encoding efficiency and a higher number of requests.

In the literature, different algorithms have been proposed to improve the streaming performance of DASH [4–7]. Because different scenes on a video may have a quite dissimilar behavior from a compressibility perspective, chunk sizes on a video may significantly vary – even within a given bitrate level. The effect of such variation over the bitrate adaptation techniques in DASH was recently evaluated in [8], by using a Gaussian approximation for the chunk size distribution. A proper characterization of the chunk size distribution is desirable in many ways: (*a*) it helps reducing the rebuffering probability and increasing the average bitrate level; (*b*) it can be used to accurately predict the sizes of future segments [9], which ultimately reduces rebuffering events and improves users' quality of experience (QoE); (*c*) it can be used to emulate different video streaming sources, in order to evaluate the system performance in real networks without the need to resort to real video traces [10]; (*d*) it enables the analytical characterization of system performance metrics.

In this paper, we provide a statistical characterization of the chunk size distribution in DASH systems based on empirical data from real video traces. Using a fitting procedure over a wide data set of videos with different qualities and segment size duration, we determine the choice of distribution parameters that provides the best fit to the empirical distributions. Results reveal that the duration of a video segment has an impact on which is the best distribution to model the segment sizes in DASH.

## II. DASH Overview

In a DASH system, as represented in Fig. 1, the video is encoded using different bitrates, known as *representation rates*. Each encoded video is fragmented into small video segments or chunks of a given size (i.e. segment/chunk size) that contain several seconds of the video content (i.e. segment/chunk duration). All the information related to representation rates, metadata, encoders, server IP address, etc. is specified in the Media Presentation Description (MPD).

DASH operation is controlled by the client, and can be described as follows:

1) The client makes a HTTP request to obtain the MPD.
2) Once the client has the MPD, the streaming session is started and the client requests video fragments to fill its video buffer. This is known as *Initial Burst* phase.
3) Afterwards, the client operates in a periodic mode in which it downloads new video chunks as the video is played back. The bitrates of these video segments will depend on the network status as well as the adaptation algorithm implemented by the client. This is known as *Throttling* phase, for which the video traffic follows an ON/OFF pattern [3].

A key aspect of DASH is that the client can request video chunks of different qualities in order to adapt to the instantaneous network status. Depending on the quality and duration of the video chunks, the encoder performance and the specific video scenes to be encoded, the size of the video segments in bytes will be different. Our goal is to provide a

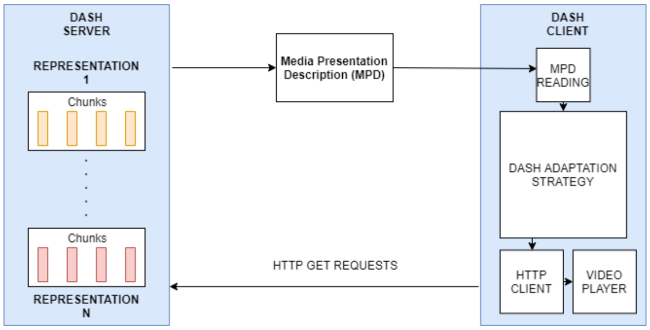


Fig. 1. DASH client-server architecture

statistical model for the chunk size distribution based on real video traces of different characteristics.

## III. Characterization of Chunk Size Distribution

### A. Experimental set-up

We first aim at attaining an empirical characterization of the chunk size distributions for a wide set of reference videos. We used the widely employed reference data set in [11][1], and more specifically, the Big Buck Bunny animation video available in twenty different quality levels, and segment size durations between 1-15 seconds.

The experimental set-up is described as follows: we used the DASH player provided by Akamai [12], and the MPD is retrieved for each video. Then, each of the twenty available representations is downloaded, for two different choices of segment duration $\tau_s$ (1s and 6s). We denote all combinations as TC-$X$-$Y$, where $X \in [1-20]$ denotes the different qualities and $Y \in \{1, 2\}$ denotes the selected segment size duration ($1 \to \tau_s = 1$s, $2 \to \tau_s = 6$s).

Note that the adaptive feature of DASH is turned-off throughout the experiment, as we want to characterize the chunk size distribution for every combination of quality and segment size duration. A very-high speed Gigabit Ethernet wired connection is used in order to avoid any potential rebuffering. Traffic is captured using Wireshark and processed with TShark to extract the packet sizes and arrival times. For every HTTP request, the segment size is computed as the sum of the payloads of all TCP packets within a given ON period. As observed in Fig. 2, the size of consecutive segments varies even for a fixed video quality. Once the empirical segment sizes are computed, we obtain the empirical cumulative distribution functions (CDFs) for each experiment.

### B. Chunk Size Distribution Fitting

After the set of forty empirical CDFs was estimated, we aim at obtaining the best fit for a set of target distributions.

[1]x264 codec for encoding video streams into the H.264/MPEG-4 AVC compression format is used.

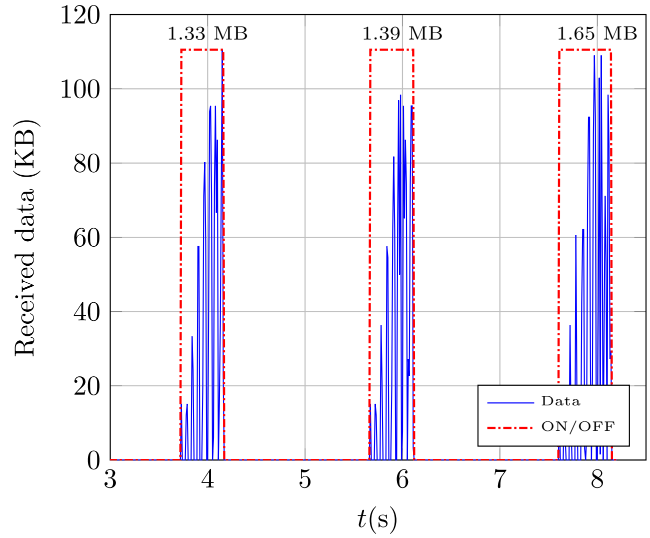


Fig. 2. Evolution of DASH traffic for 3 consecutive segments and fixed video quality, with segment duration of 2 seconds.

These target distributions are Weibull and truncated logistic[2] (TL) [13], whose CDFs are respectively given by

$$F_W(x) = 1 - e^{-\left(\frac{x}{\lambda}\right)^k} \qquad x \geq 0. \qquad (1)$$

$$F_{TL}(x) = \frac{1 - e^{-\frac{x}{s}}}{1 + e^{\frac{-(x-\mu)}{s}}} \qquad x \geq 0, \qquad (2)$$

where $\lambda, k \geq 0$ and $s, \mu \geq 0$ are the scale and shape parameters of the Weibull and TL distributions.

Tables I and II show the results of the fitting to Weibull and TL distributions, for test cases TC-$X$-1 and TC-$X$-2, respectively. For each of the test cases, which corresponds to a given video resolution and bitrate, the distribution that provides a best fit in the sense of minimizing the root mean square error (RMSE[3]) is highlighted in bold format. The average segment size for each test case is also included.

For the case of segment duration $\tau_s = 1$s, we observe that in most cases (and specially those corresponding to better video qualities) the truncated logistic distribution provides a better fit. However, Weibull distribution outperforms the TL distribution in some cases. We observe that the fitting is reasonably good for all video qualities, as observed in Fig. 3. We also see the RMSE is slightly reduced as the video bitrate is decreased.

If we now consider a longer segment duration $\tau_s = 6$s, we see that Weibull distribution always outperforms the TL distribution, regardless of the video quality. Similarly to the previous case, the fitting error is small in all test cases (see Fig. 4), although the RMSE is now reduced as the video bitrate

[2]The exponential distribution was also originally considered in the experiment. However, this choice yielded a poor fitting performance and was not relevant for being included in the analysis.

[3]Other goodness-of-fit (GOF) statistics such as R-square, adjusted R-square and the sum of squares due to error (SSE) were evaluated. For the sake of conciseness, we use the RMSE in the discussion, although equivalent conclusions are obtained using the other GOF statistics. A RMSE value closer to 0 indicates that the model has a smaller random error component and hence the fit is better.

TABLE I
Proposed model for different video qualities. Segment duration $\tau_s = 1$s.

| Test Case | Resolution | Video Bitrate | Scale Factor ($\lambda$) | Shape Factor ($K$) | RMSE_1 | Mean ($\mu$) | Scale Factor ($s$) | RMSE_2 | Av. Chunk Size (kB) |
|---|---|---|---|---|---|---|---|---|---|
| TC-1-1 | 320x240 | 47.0 kbps | 0.006596 | 3.861 | 0.07004 | **0.005979** | **0.001177** | **0.0669** | 6.14 |
| TC-2-1 | 320x240 | 92.0 kbps | **0.0124** | **11.42** | **0.02971** | 0.01197 | 0.0007527 | 0.03843 | 11.78 |
| TC-3-1 | 320x240 | 135.0 kbps | 0.02098 | 1.848 | 0.07797 | **0.01593** | **0.006948** | **0.07733** | 17.47 |
| TC-4-1 | 480x360 | 182.0 kbps | 0.02463 | 5.675 | 0.0769 | **0.02306** | **0.003025** | **0.0743** | 23.15 |
| TC-5-1 | 480x360 | 226.0 kbps | 0.03454 | 1.308 | 0.06309 | **0.01244** | **0.01942** | **0.0600** | 28.79 |
| TC-6-1 | 480x360 | 270.0 kbps | 0.03626 | 6.716 | 0.0511 | **0.03426** | **0.003637** | **0.0498** | 34.30 |
| TC-7-1 | 480x360 | 353.0 kbps | **0.04773** | **8.675** | **0.04668** | 0.04566 | 0.003722 | 0.0486 | 45.16 |
| TC-8-1 | 480x360 | 425.0 kbps | **0.05829** | **7.546** | **0.04655** | 0.0553 | 0.005275 | 0.0561 | 53.42 |
| TC-9-1 | 854x480 | 538.0 kbps | 0.08211 | 1.157 | 0.0552 | **0.0007553** | **0.05539** | **0.0523** | 69.2 |
| TC-10-1 | 854x480 | 621.0 kbps | **0.08265** | **8.76** | **0.0445** | 0.07909 | 0.006409 | 0.0467 | 78.33 |
| TC-11-1 | 1280x720 | 808.0 kbps | **0.1064** | **10.18** | **0.03723** | 0.1024 | 0.007054 | 0.0380 | 101.18 |
| TC-12-1 | 1280x720 | 1.1 Mbps | **0.1429** | **8.967** | **0.0273** | 0.1368 | 0.01079 | 0.0332 | 134.12 |
| TC-13-1 | 1280x720 | 1.3 Mbps | 0.1891 | 2.32 | 0.0407 | **0.1586** | **0.04912** | **0.0356** | 165.5 |
| TC-14-1 | 1280x720 | 1.7 Mbps | 0.2477 | 2.318 | 0.0261 | **0.2094** | **0.06756** | **0.0197** | 217 |
| TC-15-1 | 1920x1080 | 2.2 Mbps | 0.3021 | 5.262 | 0.0400 | **0.2811** | **0.03796** | **0.0388** | 282 |
| TC-16-1 | 1920x1080 | 2.6 Mbps | 0.3584 | 4.966 | 0.0411 | **0.3319** | **0.04813** | **0.0409** | 326 |
| TC-17-1 | 1920x780 | 3.3 Mbps | 0.483 | 1.884 | 0.0266 | **0.3732** | **0.1608** | **0.0205** | 419 |
| TC-18-1 | 1920x780 | 3.8 Mbps | 0.5474 | 2.956 | 0.0269 | **0.4811** | **0.1161** | **0.0265** | 484 |
| TC-19-1 | 1920x780 | 4.2 Mbps | 0.6154 | 1.742 | 0.0326 | **0.4524** | **0.2253** | **0.0252** | 529 |
| TC-20-1 | 1920x780 | 4.7 Mbps | 0.6314 | 1.825 | 0.0130 | **0.4753** | **0.2204** | **0.0129** | 565 |

TABLE II
Proposed model for different video qualities II. Segment duration $\tau_s = 6$s.

| Test Case | Resolution | Video Bitrate | Scale Factor ($\lambda$) | Shape Factor ($K$) | RMSE_1 | Mean ($\mu$) | Scale Factor ($s$) | RMSE_2 | Av. Chunk Size (kB) |
|---|---|---|---|---|---|---|---|---|---|
| TC-1-2 | 320x240 | 46.0 kbps | **0.03535** | **18.53** | **0.0158** | 0.03455 | 0.001284 | 0.0277 | 34 |
| TC-2-2 | 320x240 | 89.0 kbps | **0.06925** | **13.85** | **0.0193** | 0.06721 | 0.00339 | 0.0334 | 66 |
| TC-3-2 | 320x240 | 128.0 kbps | **0.1021** | **9.711** | **0.0397** | 0.09784 | 0.007205 | 0.0533 | 95 |
| TC-4-2 | 480x360 | 177.0 kbps | **0.138** | **13.87** | **0.0251** | 0.1339 | 0.006761 | 0.0400 | 132 |
| TC-5-2 | 480x360 | 218.0 kbps | **0.1718** | **11.29** | **0.0371** | 0.1656 | 0.01046 | 0.0514 | 163 |
| TC-6-2 | 480x360 | 255.0 kbps | **0.2034** | **8.71** | **0.0430** | 0.1941 | 0.01593 | 0.0544 | 189 |
| TC-7-2 | 480x360 | 321.0 kbps | **0.2609** | **5.713** | **0.0464** | 0.2432 | 0.03048 | 0.0537 | 237 |
| TC-8-2 | 480x360 | 374.0 kbps | **0.3077** | **4.245** | **0.0379** | 0.2809 | 0.0468 | 0.0434 | 276 |
| TC-9-2 | 854x480 | 506.0 kbps | **0.4045** | **8.619** | **0.0394** | 0.3857 | 0.03181 | 0.0498 | 378 |
| TC-10-2 | 854x480 | 573.0 kbps | **0.4633** | **6.799** | **0.0456** | 0.4364 | 0.04597 | 0.0542 | 428 |
| TC-11-2 | 1280x720 | 780.0 kbps | **0.6143** | **11.31** | **0.0317** | 0.5924 | 0.03668 | 0.0445 | 582 |
| TC-12-2 | 1280x720 | 1.0 Mbps | **0.8081** | **7.43** | **0.0420** | 0.7649 | 0.07312 | 0.0508 | 749 |
| TC-13-2 | 1280x720 | 1.2 Mbps | **0.1452** | **5.354** | **0.0393** | 0.911 | 0.1205 | 0.0455 | 899 |
| TC-14-2 | 1280x720 | 1.5 Mbps | **1.209** | **3.655** | **0.0334** | 1.088 | 0.2089 | 0.0373 | 1100 |
| TC-15-2 | 1920x1080 | 2.1 Mbps | **1.674** | **6.943** | **0.0428** | 1.58 | 0.1615 | 0.0519 | 1500 |
| TC-16-2 | 1920x1080 | 2.4 Mbps | **1.959** | **5.241** | **0.0402** | 1.817 | 0.2451 | 0.0469 | 1770 |
| TC-17-2 | 1920x780 | 2.9 Mbps | **2.41** | **3.502** | **0.0313** | 2.16 | 0.4348 | 0.0352 | 2150 |
| TC-18-2 | 1920x780 | 3.3 Mbps | **2.72** | **2.891** | **0.0314** | 2.376 | 0.5808 | 0.0352 | 2430 |
| TC-19-2 | 1920x780 | 3.6 Mbps | **2.947** | **2.585** | **0.0343** | 2.521 | 0.6976 | 0.0376 | 2640 |
| TC-20-2 | 1920x780 | 3.9 Mbps | **2.995** | **2.272** | **0.048** | 2.879 | 0.806 | 0.0519 | 2900 |

is increased. Hence, the fitting quality is now better for lower video qualities.

We also observe that even though the segment duration is increased by a factor of 6 between both sets of test cases, the average chunk size is increased by a smaller factor. This is due to the fact that a larger segment duration allows for a more efficient video compression of the video scene because of temporal correlation.

## IV. Application: Source Model for DASH

We now illustrate how the previous statistical characterization of the chunk size distributions for different video bitrates and segment duration can be leveraged to implement a source model for generating synthetic DASH-like video traces. In Fig. 5, we generated different segments with random sizes according to the distribution parameters in Table II. For simplicity, we assumed that the sizes of consecutive segments are independent, which is a valid approximation for longer segments. Indeed, the inter-segment correlation can also be estimated and incorporated into the source model [14]. Real video traces (corresponding to the DASH throttling phase) are also included, together with the average segment size for each test case as a reference. We see that the synthetic and real video traces exhibit a similar behaviour.

## V. Conclusion

A statistical model for the segment size distribution in DASH has been developed, using a large set of real video traces with different characteristics (quality, resolution and segment size duration). We have seen that Weibull distribution

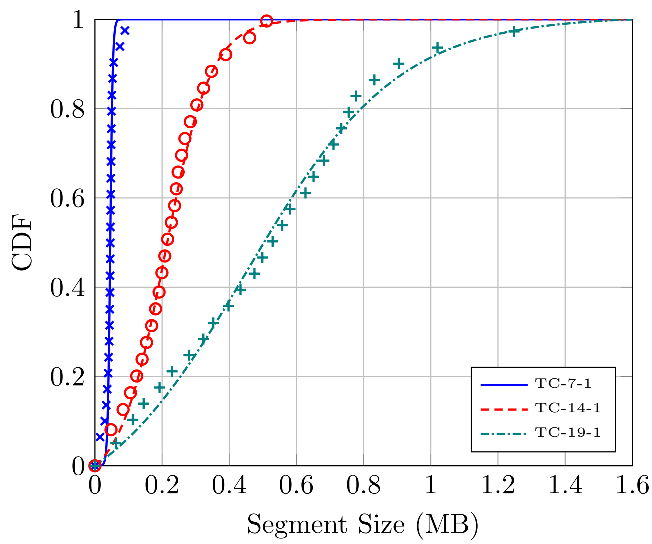


Fig. 3. Empirical (markers) vs fitted (lines) CDFs for the segment size distribution. Parameter values for the fitted CDFs are extracted from Table I and correspond to the truncated logistic distribution.

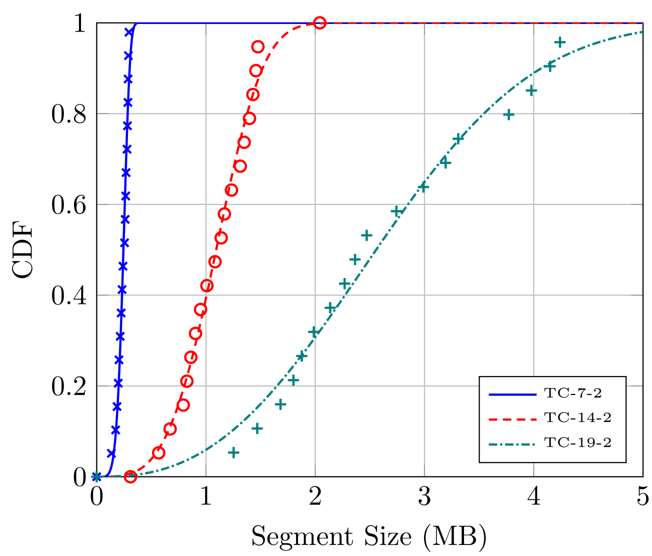


Fig. 4. Empirical (markers) vs fitted (lines) CDFs for the segment size distribution. Parameter values for the fitted CDFs extracted are from Table II and correspond to the Weibull distribution.

is more suited to model the segment size distribution for longer video segments, whereas the truncated logistic distribution provides a better fit in most situations corresponding to shorter video segments. The developed model facilitates the generation of synthetic video traces of configurable characteristics. Future extensions include studying the effect of the choice of video codec in the segment size distribution, and the use of different data sets corresponding to other video types.

## ACKNOWLEDGMENT

This work has been supported by the Spanish Government (Ministerio de Economia y Competitividad) under grant TEC2016-80090-C2-1-R, and Universidad de Malaga through Plan Propio de Investigacion (ayudas para la iniciacion a la investigacion).

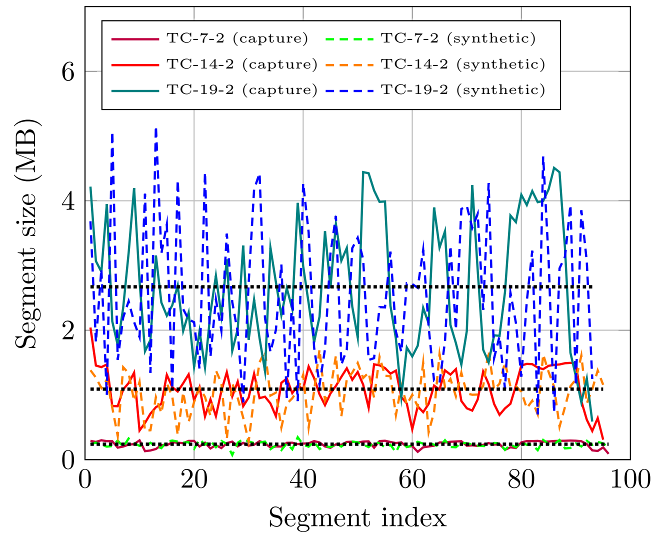


Fig. 5. Real vs. synthetic video traces for different video bitrates. Thick dotted black lines indicate the average chunk sizes in Table II as a reference. Segment duration $\tau_s = 6$s.

## REFERENCES


[1] Cisco, Cisco visual Networking index. Forecast and methodology, 2016 to 2021. https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html, 2017.

[2] MPEG, I. "Information technology-dynamic adaptive streaming over HTTP (DASH)-part 1: Media presentation description and segment formats." ISO/IEC MPEG, Tech. Rep (2012).

[3] J. Kua, G. Armitage and P. Branch. "A survey of rate adaptation techniques for dynamic adaptive streaming over HTTP," IEEE Commun. Surveys Tuts., 19:1842–1866, 2017.

[4] T. C. Thang, H. T. Le, A. T. Pham and Y. M. Ro, "An Evaluation of Bitrate Adaptation Methods for HTTP Live Streaming," IEEE J. Sel. Areas Commun., vol. 32, no. 4, pp. 693-705, April 2014.

[5] Yin, X., Jindal, A., Sekar, V., Sinopoli, B., "A control-theoretic approach for dynamic adaptive video streaming over HTTP," ACM SIGCOMM Comput. Commun. Rev., vol. 45, no. 4, pp. 325-338, Aug. 2015.

[6] Akhshabi, S., Begen, A. C., and Dovrolis, C, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP," Proc. 2nd ACM Conf. on Multimedia systems, pp. 157-168, Feb. 2011.

[7] J. Mongay Batalla, P. Krawiec, A. Beben, P. Wisniewski and A. Chydzinski, "Adaptive Video Streaming: Rate and Buffer on the Track of Minimum Rebuffering," in IEEE Journal on Selected Areas in Communications, vol. 34, no. 8, pp. 2154-2167, Aug. 2016.

[8] T. Zhang, F. Ren, W. Cheng, X. Luo, R. Shu and X. Liu, "Modeling and analyzing the influence of chunk size variation on bitrate adaptation in DASH," IEEE INFOCOM 2017 - IEEE Conf. on Computer Commun., Atlanta, GA, 2017, pp. 1-9.

[9] A. Kalampogia and P. Koutsakis, "H.264 and H.265 Video Bandwidth Prediction," in IEEE Trans. Multimedia, vol. 20, no. 1, pp. 171-182, Jan. 2018.

[10] J. Yang, Y. Ran, S. Chen, W. Li and L. Hanzo, "Online Source Rate Control for Adaptive Video Streaming Over HSPA and LTE-Style Variable Bit Rate Downlink Channels," IEEE Trans. Veh. Technol., vol. 65, no. 2, pp. 643-657, Feb. 2016.

[11] S. Lederer, C. Muller, and C. Timmerer. "Dynamic adaptive streaming over HTTP dataset"; Proc. 3rd Multimedia Systems Conf. (MMSys'12); ACM, New York, NY, USA, 89-94.

[12] Akamai DASH player. Available at: https://players.akamai.com/dash/

[13] Ahmadpour, S., Wan, T. C., Toghrayee, Z., and HematiGazafi, F. "Statistical Analysis of Video Frame Size Distribution Originating from Scalable Video Codec (SVC)". Complexity, vol. 2017, art. ID 8098574, 12 pages, 2017.

[14] L. M. Novak, "Generating correlated Weibull random variables for digital simulations," IEEE Conf. on Decision and Control; 12th Symposium on Adaptive Processes, San Diego, CA, USA, 1973, pp. 156-160.