# On Multiple Media Representations and CDN Performance

Yuriy Reznik
Brightcove, Inc.
Seattle, WA, USA
yreznik@brightcove.com

Thiago Teixeira
Brightcove, Inc.
Boston, MA, USA
tteixeira@brightcove.com

Robert Peck
Brightcove, Inc.
Scottsdale, AZ, USA
rpeck@brightcove.com

## ABSTRACT

This paper proposes a mathematical model describing the effects of using multiple media representations on CDN performance in HTTP-based streaming systems. Specifically, we look at cases of using multiple versions of the same content packaged differently and derive an asymptotic formula for CDN cache-miss probability considering parameters of the content's distribution and the distribution of formats used for packaging and delivery. We then study the validity of this proposed formula by considering statistics collected for several streaming deployments using mixed HLS and DASH packaging and show that it predicts the experimentally observed data reasonably well. We further discuss several possible extensions and applications of this proposed model.

## CCS CONCEPTS

• Multimedia information systems → Multimedia streaming

## KEYWORDS

HTTP Adaptive Streaming, HLS, DASH, CMAF, CDN performance, Caching Algorithms, Analysis of Algorithms

**ACM Reference format:**

Yuriy Reznik, Thiago Teixeira, and Robert Peck. 2021. On Multiple Media Representations and CDN Performance. *In Proceedings of ACM Mile – High Video (MHV'22).* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3510450.3517320

## 1 INTRODUCTION

As well known, modern-era HTTP-based adaptive streaming systems use multiple representations of the same media content for delivery. For instance, the same video can be encoded at different resolutions and bitrates [1-4]. With different codecs supported by different receiving devices, there may be further a need to encode some streams, e.g., using H.264 [5], and others using HEVC [6] or AV1 [7]. With multiple streaming formats (HLS [8], DASH [9], MSS [10]) and DRM systems (FairPlay [11], Widevine [12], PlayReady [13], etc.), there may further need to produce several versions of each stream with different file formats or encryption schema applied [17,18].

In other words, the use of multiple representations of the same content enables the broader reach and adaptivity of streaming media delivery. But on the other hand, the use of such multiple representations also makes the job of Content Delivery Networks (CDNs) much more difficult.

CDNs can be generally understood as distributed cache systems. They consist of multiple geographically dispersed edge caches, and where each edge cache is trying to keep content that is most popular/retrievable in each location. When there are multiple copies of the same content, and they are all being pulled by different devices, the CDN edge cache has to try to keep them all in its memory. However, when such memory is limited, these copies of the same content begin to compete against each other to stay in the cache. And in some cases, only a subset of them becomes cached.

When a content representation cannot be stored in the edge cache, the request to pull it results in a CDN cache miss and a request to pull it from another server. In the worst case, the request to pull can propagate all the way to the origin server, which is likely to cost more (e.g., if such server is operated by cloud platform, and each byte outgoing is subject to transfer costs), and put at risk the scalability and reliability of the entire streaming system. Pull requests from the origin server also typically result in significant delays, potentially causing streaming players to buffer or make abrupt rate switch decisions – in both cases diminishing the user experience.

The described effects are well-known for CDN vendors and DASH/HLS streaming systems operators. The fact that multiple streaming formats and DRMs can hurt CDN performance is also known and has been one of the key motivations for introducing the CMAF standard [16]. Several studies related to CDN performance for streaming have also been published [4,19-22]. However, to the best of the author's knowledge, the precise mathematical characterization of such effects has not yet been proposed or derived.

This paper is dedicated to the derivation of such a mathematical model. In Section 2, we will introduce some basic concepts, including the model of the popularity of media content items and performance limits of a cache system

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
*MHV '22, March 1–3, 2022, Denver, CO, USA*
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9222-8/22/03…$15.00
https://doi.org/10.1145/3510450.3517320

operating with items following such usage distribution. In Section 3, we will extend this model to account for the effects of multiple representations of the media content. In Section 4, we will conduct an experimental study to test and validate the proposed mathematical model. In Section 5, we will discuss several possible applications and extensions of the proposed model.

## 2 BASIC CONCEPTS

### 2.1 Content Popularity Model

Consider a set of *items*, such as encoded video representations or their segments:

$$S = \{s_x, x \geq 1\}$$

Let us also assume that these content items are ordered according to their usage / popularity, and that in such an ordered form they follow some canonical probability model. Specifically, in this work, we will assume that content popularity can be modeled by Zeta distribution [23]:

$$p(s_x) = \frac{x^{-\alpha}}{\zeta(\alpha)}, \quad (1)$$

where α is a shape parameter, and ζ(.) is a Riemann Zeta function [23]. In Figure 1, we show few example shapes of this distribution with various values of parameter α.

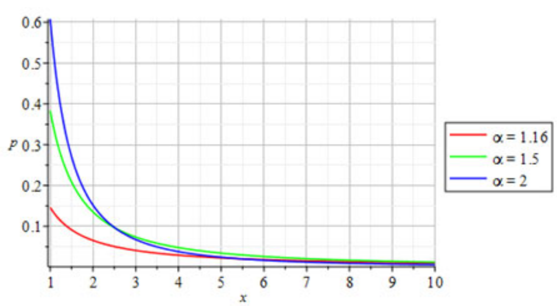


**Figure 1: Plots of Zeta distribution for several possible values of parameter α.**

This is a classic distribution, which can be understood as a discrete version of Pareto distribution [25], or generalization of so called 80/20 rule. It is broadly used in practice, and known to be adequate for modeling of many natural distributions, including distributions of popularity of video content in YouTube [26].

### 2.2 An Idealized Cache System

Let us next assume that our set of content items with probability distribution $p(s_x)$ is cached by a CDN edge cache with capacity of C items. If the caching algorithm is ideal, such cache will store C most popular items. The probability of hit in this case will be:

$$p_{hit}(C,\alpha) = \sum_{x=1}^{C} p(s_x) = \frac{H_{C,\alpha}}{\zeta(\alpha)}, \quad (2)$$

where

$$H_{C,\alpha} = \sum_{x=1}^{C} \frac{1}{x^{\alpha}} \quad (3)$$

is the generalized Harmonic number [27]. The probability of cache miss, consequently, will be:

$$p_{miss}(C,\alpha) = 1 - p_{hit}(C,\alpha) = 1 - \frac{H_{C,\alpha}}{\zeta(\alpha)}, \quad (4)$$

Naturally, the larger the capacity of the cache C, the lower is the cache miss probability $p_{miss}(C,\alpha)$.

### 2.3 Asymptotic Cache Hit / Miss Probabilities

It is known that generalized Harmonic numbers can also be expressed as a difference between Riemann and Hurwitz Zeta functions [28]:

$$H_{C,\alpha} = \zeta(\alpha) - \zeta(\alpha, C + 1).$$

When C is large, we can use known asymptotic expansion of Hurwitz Zeta function [27], leading to the following expression:

$$H_{C,\alpha} = \zeta(\alpha) + C^{-\alpha}\left(\frac{C}{1-\alpha} + \frac{1}{2} - \frac{\alpha}{12C} + O\left(\frac{1}{C^2}\right)\right). \quad (5)$$

Consequently, by plugging this expansion in formulae for cache hit/miss probabilities, we obtain:

$$p_{hit}(C,\alpha) \sim 1 - \frac{C^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right), \quad (6)$$

$$p_{miss}(C,\alpha) \sim \frac{C^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right). \quad (7)$$

The obtained expressions characterize the performance of an idealized cache system operating with a collection of items following Zeta distribution with decay parameter α.

## 3 CACHE SYSTEM WITH 2 FORMATS

### 3.1 Probabilities of Formats and Content Items

Next, let us now assume that all same content items are now packaged and stored in 2 different formats, effectively forming 2 sets:

$$S_1 = \{s_{1,x}, x \geq 1\}, \quad \text{and} \quad S_2 = \{s_{2,x}, x \geq 1\}.$$

As an example, the set with index 1 could be imply HLS-packaged media segments, and the set with index 2 - DASH-packaged segments.

We will assume that the packaging format does not influence the popularity of the content items per se, but it does limit the percentage of players that can pull and decode it.

Hence, if by

$$\pi = \{\pi_1, \pi_2 | \pi_1 + \pi_2 = 1\}$$

we will denote the distribution of players supporting 1st and 2nd format respectively, then we can express full probabilities of content items as they will be pulled from CDN as

$$p(s_{1,x}) = \pi_1 \cdot p(s_x), \quad \text{and} \quad p(s_{2,x}) = \pi_2 \cdot p(s_x).$$

As earlier, $p(s_x)$ denotes content item probabilities, and $\pi_1, \pi_2$ denote support probabilities of 1st and second formats.

## 3.2 Structure of Cache Memory

Let us now consider a mixed set of all content items:

$$S_\Sigma = S_1 \cup S_2,$$

and try to order it according to full probabilities of such items. For simplicity, let's assume that $\pi_1 > \pi_2$, implying that the first format is more widely supported/used than the second one. Then, at the top of the cache, we may observe the structure shown in Table 1.

**Table 1: Content items in order of decreasing probability.**

| Item | Probability | Comments |
|---|---|---|
| $s_{1,1}$ | $\pi_1 p(1)$ | First go items packaged using more widely supported format |
| ... | ... | ... |
| $s_{1,x}$ | $\pi_1 p(x)$ | $x = \left\lfloor \left(\frac{\pi_1}{\pi_2}\right)^{\frac{1}{\alpha}} \right\rfloor$, solution of $\pi_1 p(x) = \pi_2 p(1)$ |
| $s_{2,1}$ | $\pi_2 p(1)$ | |
| $s_{1,x+1}$ | $\pi_1 p(x+1)$ | Then follow items from more widely supported content |
| ... | ... | ... |
| $s_{1,x_2}$ | $\pi_1 p(x_2)$ | $x_2 = \left\lfloor 2\left(\frac{\pi_1}{\pi_2}\right)^{\frac{1}{\alpha}} \right\rfloor$, solution of $\pi_1 p(x_2) = \pi_2 p(2)$ |
| $s_{2,2}$ | $\pi_2 p(2)$ | Next comes the second item packaged in less supported format |
| $s_{1,x_2+1}$ | $\pi_1 p(x_2+1)$ | Then again follow items from more widely supported format |
| ... | ... | ... |

As can be observed from Table 1, the sorted list will first include a chain of x items from more supported format, followed by single item from less supported content, then again x items from more supported format, etc.

The quantity x defining such interleaved / sorted order can be observed to satisfy

$$x \approx \left(\frac{\pi_1}{\pi_2}\right)^{\frac{1}{\alpha}}$$

subject to rounding to the nearest smaller integer.

## 3.2 Cache Hit/Miss Probabilities with 2 Formats

Considering the structure of the content in cache as shown in Table 1, we can now compute both cache-hit and miss probabilities.

We note that if C represents total cache memory limit, then with structure of data as described in Table 1, such cache should store approximately $\frac{C\cdot x}{x+1}$ elements of the content in format 1, and $\frac{C}{x+1}$ elements in the second format.

This simple argument produces:

$$p_{hit,2}(C,\alpha,\pi) = \sum_{x=1}^{\frac{Cx}{x+1}} \pi_1 p(s_x) + \sum_{x=1}^{\frac{C}{x+1}} \pi_2 p(s_x)$$
$$= \pi_1 \frac{H_{\frac{Cx}{x+1},\alpha}}{\zeta(\alpha)} + \pi_2 \frac{H_{\frac{C}{x+1},\alpha}}{\zeta(\alpha)}.$$

By plugging next asymptotic expansion for the generalized Harmonic numbers (5), we obtain:

$$p_{hit,2}(C,\alpha,\pi) \sim \pi_1 \left(1 - \frac{\left(\frac{Cx}{x+1}\right)^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right)\right)$$
$$+ \pi_2 \left(1 - \frac{\left(\frac{C}{x+1}\right)^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right)\right),$$

and consequently:

$$p_{hit,2}(C,\alpha,\pi) \sim 1 - \left(\pi_1 \left(\frac{x}{x+1}\right)^{1-\alpha}\right.$$
$$\left. + \pi_2 \left(\frac{1}{x+1}\right)^{1-\alpha}\right) \frac{C^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right),$$

Finally, by plugging expression for $x = \left(\frac{\pi_1}{\pi_2}\right)^{\frac{1}{\alpha}}$, noting that $\pi_2 = 1 - \pi_1$, and by using some basic algebra we arrive at:

$$p_{hit,2}(C,\alpha,\pi) \sim 1 - \left(\pi_1^{\frac{1}{\alpha}} + \pi_2^{\frac{1}{\alpha}}\right)^{\alpha} \frac{C^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right), \quad (8)$$

and:

$$p_{miss,2}(C,\alpha,\pi) \sim \left(\pi_1^{\frac{1}{\alpha}} + \pi_2^{\frac{1}{\alpha}}\right)^{\alpha} \frac{C^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{C}\right)\right). \quad (9)$$

The obtained expressions (8,9) represent generalizations of the earlier expressions for cache hit/miss probabilities (6,7) for a case when media items are stored in two formats.

## 3.2 Relative Effect on Cache Miss Probability

Let us now define a ratio between cache miss probabilities in the single and 2-format systems:

$$\xi(C,\alpha,\pi) = \frac{p_{miss,2}(C,\alpha,\pi)}{p_{miss}(C,\alpha)}.$$

By plugging the respective asymptotic expressions (7 and 9), and disregarding the vanishing $O(C^{-1})$ terms, we arrive at:

$$\xi(\alpha,\pi) \sim \left(\pi_1^{\frac{1}{\alpha}} + \pi_2^{\frac{1}{\alpha}}\right)^{\alpha}. \quad (10)$$

We first immediately notice that this formula (10) is no longer dependent on cache size C, implying that asymptotically this ratio should converge to a constant! This implies that we no longer need to know precises values of CDN edge cash sizes or exact CDN topology in order to assess relative effect of uses of 2 formats on its performance!

We also note, that this ratio $\xi(\alpha,\pi)$ can be understood as a measure of the asymmetry of format support distribution π. Effectively, it is a classic $\ell_p$-norm over this distribution with norm parameter $p=\frac{1}{\alpha}$.

When either of the formats has high support probability ($\pi_1 \to 1$ or $\pi_2 \to 1$), then

$$\xi(\alpha,\pi) \to 1,$$

implying that cache miss probability in 2-format case will be the same as in case of a single format. This holds regardless of CDN cache size or the parameter α of the content popularity distribution.

The worst-case situation will be when both formats are about equally well supported. In this case:

$$\pi_1 = \pi_2 = 1/2,$$

and consequently:

$$\xi(\alpha, 1/2) \sim 2^{\alpha-1}.$$

We note that, in this case, the cache miss probability grows exponentially as a function of the content popularity distribution parameter α. The higher is the α or the sharper is the decay in the popularity distribution, the higher is the impact of the second format on the performance of the system.

We show overall behavior of a function $\xi(\alpha,\pi)$ in Figure 2.

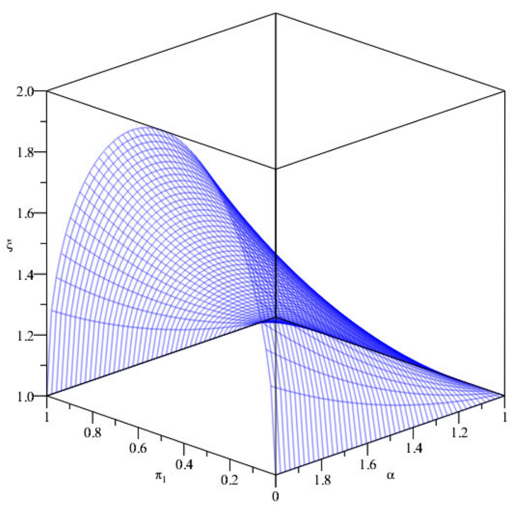


**Figure 3: Plot of function $\xi(\alpha,\pi)$. Parameter $\alpha$ determines the slope of the content popularity distribution, and $\pi_1$ is the support probability of the 1st format.**

# 4 EXPERIMENTAL VALIDATION

## 4.1 Data Collection and Processing

To conduct the study, we used Brightcove VideoCloud streaming delivery system [29], in which some accounts used only HLS format for delivery, while others used a combination of HLS and DASH formats. In the latter case, the CDN was effectively supplied with 2 copies of the same content – one was the set of TS-packaged segments for HLS v3/v4 delivery, and the other – ISOBMFF-packaged segments for DASH delivery.

For each account we then produced a capture of CDN and origin server logs corresponding to the same time interval and the same delivery region. The total number of records in CDN logs captured for this study was around 40M.

Then, for each account, we've computed the total volume of requests V, and also parameters $\alpha$ achieving best fits of the observed per-segment load distributions to the Zeta distribution model. Using both volume V and shape parameters $\alpha$, we have then selected 30 pairs of best matching accounts with HLS only – and mixed HLS+DASH delivery modes.

In other words, we have identified 30 test cases where volume- and content popularity distribution- wise the delivery conditions were similar, but in terms of formats they were not: HLS-only accounts used one packaging format, while HLS+DASH accounts used two.

Then, using such selected pairs of accounts, we have computed the effective CDN cache miss probabilities $\hat{p}_{miss}$ and $\hat{p}_{miss,2}$ based on data exchanges reported CDN and origin server logs, and then computed their ratios: $\hat{\xi}=\frac{\hat{p}_{miss,2}}{\hat{p}_{miss}}$. Such experimentally measured ratios we then compared to our model's suggested values $\xi(\alpha,\pi)$, where parameters $\alpha$ and $\pi$ were selected to match the observed parameters content and formats distributions in mixed HLS+DASH accounts.

## 4.2 The Results

In Figure 4 we plot experimentally obtained parameters $\hat{\xi}$ for each pair of accounts under the test, superimposed with our model function $\xi(\alpha,\pi)$.

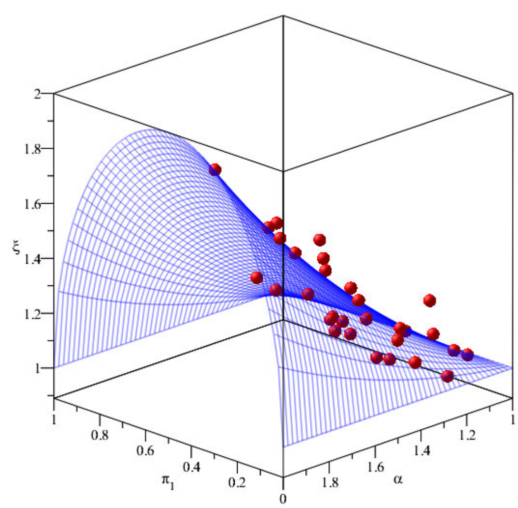


**Figure 4: Plot of our model function $\xi(\alpha,\pi)$ with superimposed ratios of cache miss probabilities $\hat{\xi}$ measured experimentally.**

Based on Figure 4, it can be observed that the experimentally measured values $\hat{\xi}$ and the values $\xi(\alpha,\pi)$ computed by using our model are reasonably well aligned. The overall RMS achieved between experimental and predicted values in our experiments was around 0.098.

For convenience, the overall statistics collected in our experiment are summarized in Table 2.

**Table 2. Summary statistics of the experiments.**

| Characteristic | Value |
|---|---|
| The total number of records in CDN logs used for analysis | 40M |
| The number of pairs of HLS and HLS+DASH systems selected for comparative analysis | 30 |
| RMS of fit of $\alpha$ parameters (Zeta distribution models for content popularity), average across all cases | 0.051 |
| RMS of fit of $\xi$ values as predicted by the model | 0.098 |

## 5 EXTENSIONS AND CONCLUDING REMARKS

The presented analysis of caching system can be extended in a variety of ways to describe additional effects that may be of interest in studies of CDNs and HTTP-based streaming systems.

For example, if we consider a system with $k > 2$ formats with support probabilities: $\pi = \{\pi_1, \dots, \pi_k\}$, then the increase in the cache miss probability in such a system relative to the system with same content and only one format will be:

$$\xi(C,\alpha,\pi) = \frac{p_{miss,k}(C,\alpha,\pi)}{p_{miss}(C,\alpha)} \sim \left(\pi_1^{\frac{1}{\alpha}} + \dots + \pi_k^{\frac{1}{\alpha}}\right)^{\alpha}.$$

This is a simple and natural extension of our formula (10) and other results derived in Section 4. It confirms all same effects, including prediction that the worst-case impact on the cache performance will happen in the case when all k formats are about equally well supported by the receiving devices:

$$\pi_1 = \dots = \pi_k = \frac{1}{k} \;\; \Rightarrow \;\; \xi \sim k^{\alpha-1}.$$

This is clearly a scenario to avoid in practice.

Another extension of our model is possible by considering a case when different formats have different compression performance or storage requirements. This may be helpful for studies of mixed codec streaming deployments, or even standard ABR streaming, considering multiple representations or renditions of the same content used for streaming.

Specifically, if we consider a case when the media content is encoded by using same encoding ladder with k renditions, and bitrates $R = \{R_1, \dots, R_k\}$, and when $\pi = \{\pi_1, \dots, \pi_k\}$ denotes probabilities at which such different renditions are being pulled during streaming, then we can establish the following:

1) cache miss probability of a system using 1 encoding at bitrate $R_1$ and with cache capacity of S [kbits]:

$$p_{miss}(S,R_1,\alpha) = \frac{(S/R_1)^{1-\alpha}}{(\alpha-1)\zeta(\alpha)}\left(1 + O\left(\frac{1}{S}\right)\right)$$

2) cache miss probability in a system using k encodings with rate ladder R and with cache capacity of S [kbits]:

$$p_{miss,k}(S,R,\pi,\alpha) = \left(\pi_1^{\frac{1}{\alpha}} + \dots + \pi_k^{\frac{1}{\alpha}}\right)^{\alpha} \frac{(S/R^*)^{1-\alpha}}{(1-\alpha)\zeta(\alpha)}\left(1 + O\left(\frac{1}{S}\right)\right)$$

where

$$R^* = \frac{1}{\pi_1^{\frac{1}{\alpha}} + \dots + \pi_k^{\frac{1}{\alpha}}}\left(\pi_1^{\frac{1}{\alpha}} R_1 + \dots + \pi_k^{\frac{1}{\alpha}} R_k\right)$$

is the average rate of the content as it stays in the cache,

3) The relative increase of the cache miss probability in a system using ABR rate ladder R vs. system using encoding at single rate $R_1$:

$$\xi(S,\alpha,\pi,R_1,R) = \frac{p_{miss,k}(S,R,\pi,\alpha)}{p_{miss}(S,R_1,\alpha)} \sim \left(\pi_1^{\frac{1}{\alpha}} + \dots + \pi_k^{\frac{1}{\alpha}}\right)^{\alpha}\left(\frac{R^*}{R_1}\right)^{\alpha-1}$$

The last few formulae are indeed quite interesting, as they show that cache misses in ABR system are functions of not only of the content popularity distribution and the load probabilities of each rendition $\pi_i$, but also of average bitrate $R^*$, where the averaging is performed by using the power of $1/\alpha$ – transformed load probabilities. This form of averaging is not immediately obvious, but it is a consequence of probability-based re-orderings and selective storage of different renditions by the cache systems.

Many additional modifications and extensions of this analysis can also be proposed to take into account specifics of implementations of particular caching algorithms that may be used by CDN edge caches. Adding such specifics may undoubtedly improve accuracy of predictions of such models.

## REFERENCES


[1] G. Conklin, G. Greenbaum, K. Lillevold, A. Lippman, and Y. Reznik, "Video coding for streaming media delivery on the internet," *IEEE Trans. CSVT*, vol. 11, no. 3, 2001, pp. 269-281.

[2] Yuriy A Reznik, Karl O Lillevold, Abhijith Jagannath, Justin Greer, and Jon Corley. 2018. Optimal design of encoding profiles for ABR streaming. In *Proceedings of the 23rd Packet Video Workshop*. 43–47.

[3] Yuriy A Reznik, Xiangbo Li, Karl O Lillevold, Abhijith Jagannath, and Justin Greer. 2019. Optimal Multi-Codec Adaptive Bitrate Streaming. In *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 348–353.

[4] Nabajeet Barman and Maria G Martini. 2019. QOE modeling for HTTP adaptive video streaming – a survey and open challenges. *IEEE Access* 7 (2019), 30831–30859.

[5] ISO/IEC 14496-10:2020 MPEG-4 AVC | ITU-T Recommendation H.264, 2021, https://www.iso.org/standard/75400.html, https://www.itu.int/rec/T-REC-H.264

[6] ISO/IEC 23008-2:2013 MPEG-H HEVC | ITU-T Recommendation H.265, 2021, https://www.iso.org/standard/35424.html, https://www.itu.int/rec/T-REC-H.265

[7] AV1 Video Codec, Alliance for Open Media, 2021, https://aomedia.org/av1/

[8] HTTP Live Streaming 2nd Edition, 2019, https://tools.ietf.org/html/draft-pantos-hls-rfc8216bis-08

[9] ISO/IEC 23009-1:2014, DASH, 2014. https://www.iso.org/about-us.html

[10] Microsoft Smooth Streaming. https://www.iis.net/downloads/microsoft/smooth-streaming

[11] FairPlay Streaming: https://developer.apple.com/streaming/fps/

[12] PlayReady. https://www.microsoft.com/playready/overview/

[13] Widevine. https://www.widevine.com/solutions/widevine-drm

[14] ISO/IEC 13818-1:2019. MPEG-2 TS. https://www.iso.org/standard/75928.html

[15] ISO/IEC 14496-12:2015. ISOBMFF. 2015. https://www.iso.org/standard/68960.html

[16] ISO/IEC 23000-19:2018. Common Media Application Format (CMAF). 2018. https://www.iso.org/standard/71975.html

[17] Y. Reznik, X. Li, K. Lillevold, R. Peck, T. Shutt and P. Howard, "Optimizing Mass-Scale Multi-Screen Video Delivery," in SMPTE Motion Imaging Journal, vol. 129, no. 3, pp. 26-38, April 2020

[18] Y. Reznik, J. Cenzano and B. Zhang, "Transitioning Broadcast to Cloud," in SMPTE Motion Imaging Journal, vol. 130, no. 9, pp. 18-32, Oct. 2021

[19] Danny H Lee, Constantine Dovrolis, and Ali C Begen. 2014. Caching in http adaptive streaming: Friend or foe?. In *Proceedings of Network and Operating System Support on Digital Audio and Video Workshop*. 31–36.

[20] Parikshit Juluri and Deep Medhi. 2015. Cache'n DASH: Efficient Caching for DASH. *SIGCOMM Comput. Commun*. Rev. 45, 4 (October 2015), 599–600

[21] W. Li, S. M. A. Oteafy and H. S. Hassanein, "StreamCache: Popularity-based caching for adaptive streaming over information-centric networks," *2016 IEEE International Conference on Communications (ICC),* 2016, pp. 1-6.

[22] He Ma, Beomjoo Seo, and Roger Zimmermann. 2014, Dynamic scheduling on video transcoding for MPEG DASH in the cloud environment. In Proceedings of the 5th ACM Multimedia Systems Conference (MMSys '14). Association for Computing Machinery, New York, NY, USA, 283–294.

[23] Zeta distribution https://en.wikipedia.org/wiki/Zeta_distribution

[24] Riemann Zeta function http://mathworld.wolfram.com/RiemannZetaFunction.html

[25] Pareto distribution https://en.wikipedia.org/wiki/Pareto_distribution

[26] Cha, Meeyoung & Kwak, Haewoon & Rodriguez, Pablo & Ahn, Yong-Yeol & Moon, Sue. (2009). Analyzing the Video Popularity Characteristics of Large-Scale User Generated Content Systems.

[27] Generalized harmonic numbers http://functions.wolfram.com/GammaBetaErf/HarmonicNumber2/introductions/DifferentiatedGammas/ShowAll.html

[28] Hurwitz Zeta function http://mathworld.wolfram.com/HurwitzZetaFunction.html

[29] Brightcove VideoCloud system. https://www.brightcove.com/en/products/online-video-platform/