Optimal Design of Encoding Profiles for ABR Streaming

Yuriy A. Reznik, Karl O. Lillevold, Abhijith Jagannath, Justin Greer, and Jon Corley

Brightcove, Inc., Seattle, WA, 98101, USA
{yreznik, klillevold, ajagannath, jgreer, jcorley}@brightcove.com

ABSTRACT

We discuss the problem of optimal design of encoding profiles for adaptive bitrate (ABR) streaming. We formalize this problem and show that it belongs to a class of non-linear constrained optimization problems, with several methods available for solving it numerically. We illustrate the effectiveness of our approach by several examples of optimal encoding ladders constructed for different sources and network models.

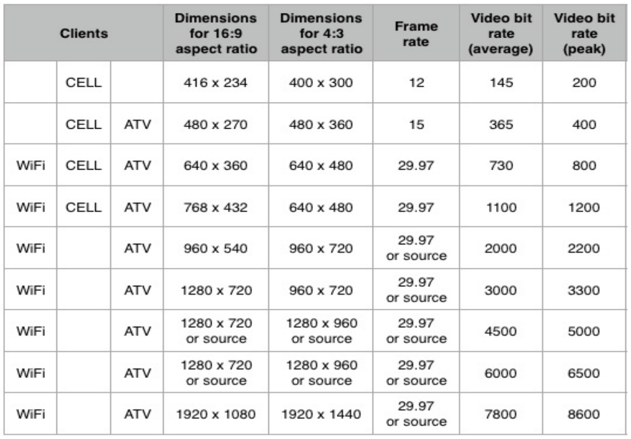
1. INTRODUCTION

During last two decades Internet streaming has evolved from a pioneering concept to mainstream technology used for delivery of media content [1-3]. The important step in this evolution was the invention of the concept of *adaptive bit-rate* (ABR) streaming [3-5].

In ABR streaming system, the content is encoded at several bitrates, and where each encoded stream incorporates random access points (e.g. IDR-frames), allowing switching between the streams. During the playback, the *streaming client* monitors the rate at which encoded content is arriving. If such rate becomes insufficient for continuous playback, the client switches to a lower bitrate stream. This prevents buffering. On the other hand, if such rate is greater than bitrate of the current stream, the client may switch to a higher bitrate stream. This maximizes quality of video delivered to end user. The first commercial product built on ABR principles was RealNetworks system G2, released in July 1998 [3]. The ABR mechanism has since become widely adopted, and is incorporated in most modern streaming protocols, such as HLS [6], MPEG DASH [7], etc.

The composition of characteristics of streams used for ABR streaming, such as their bitrates, resolutions, codec constraints, etc. is commonly called an *encoding profile* or *ladder.* When first ABR streaming systems were deployed, the encoding profiles were very simple: they typically included 28k, 56k, and 128k streams, corresponding to connection speeds achievable by dial-up and ISDN modems. When faster connections become available, the encoding profiles were extended to include few higher-bitrate streams. Examples of recent encoding profiles, as recommended for HLS streaming [8], are shown in Figure 1.

Most commonly, ABR encoding profiles are designed to be *universal* – intended for use for all media files, receiving devices, and delivery networks. However, there are at least two arguments that can be made to show that *universal ladder designs are sub-optimal.* First, rate-distortion characteristics are very different for different types of content. E.g. cartoons are more compressible than action movies. This suggests that "per-title" generated profiles can be more efficient [9]. Second, the networks and devices used for streaming are also very different. As shown in Figure 2, such differences may manifest themselves in different shapes of bandwidth PDFs as observed by different streaming clients. This suggests that encoding ladders designed for different categories of networks or receiving devices can also be more efficient that universal ladders.



Clients			Dimensions for 16:9 aspect ratio	Dimensions for 4:3 aspect ratio	Frame rate	Video bit rate (average)	Video bit rate (peak)
	CELL		416 x 234	400 x 300	12	145	200
	CELL	ATV	480 x 270	480 x 360	15	365	400
WiFi	CELL	ATV	640 x 360	640 x 480	29.97	730	800
WiFi	CELL	ATV	768 x 432	640 x 480	29.97	1100	1200
WiFi		ATV	960 x 540	960 x 720	29.97 or source	2000	2200
WiFi		ATV	1280 x 720	960 x 720	29.97 or source	3000	3300
WiFi		ATV	1280 x 720 or source	1280 x 960 or source	29.97 or source	4500	5000
WiFi		ATV	1280 x 720 or source	1280 x 960 or source	29.97 or source	6000	6500
WiFi		ATV	1920 x 1080	1920 x 1440	29.97 or source	7800	8600

Figure 1. Encoding profiles recommended for HLS [8].

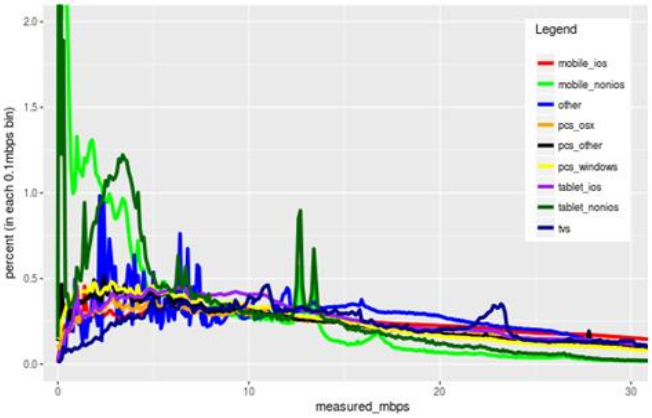


Figure 2. Bandwidth PDFs as observed by different clients. Source: Brightcove analytics, April 2017.

In this paper, we formulate and study the problem of optimal design of encoding profiles for ABR streaming considering R/D characteristics of the source, client model, and model of networks used for delivery. We show that this problem belongs to a class of known optimization problems and identify techniques suitable for solving it numerically. We also present several examples demonstrating effectiveness of our approach. Specifically, we show that optimal ladders designed for different sources and different networks are different. They have different numbers encoded streams, different placements of bitrates, and other parameters.

In passing, we must note the problem of ladder design accounting for characteristics of the source has already been considered in [9]. However, the approach presented in [9] still leaves several dimensions of the problem (such as number of streams to include in the ladder, or how to find best distribution of bitrates) being unsettled. Our problem setting is more general and more complete, leading to a unique optimal solution considering all ladder parameters. Our approach is also accounting for statistics of networks used for delivery. Perhaps closest prior attempts to look at both source and channel-related aspects were done in the context of optimizations done for packet-based streaming [2,10]. The problem we consider, however, is different and specific for HTTP-based streaming.

This paper is organized as follows. In next session we introduce all models and define performance parameters of ABR streaming systems. In Section 3, we formulate the problem of design of quality-optimal ABR ladders and show how it can be solved in few example cases. Extensions and concluding remarks are offered in Section 5.

2. PERFORMANCE OF ABR STREAMING

2.1. Quality-rate models

Given an encoder, range of bitrates, source content, and a quality metric (e.g. PSNR, SSIM, etc.), one can produce a sequence of encodings, resulting in pairs of values $(R_i, Q_i), i = 1,2, \dots$, where R_i denotes bitrate and Q_i denotes quality. For most codecs and content, we can further expect that such points form a monotonically increasing sequence $\forall i, j : R_j \geq R_i \Rightarrow Q_j \geq Q_i$, which can be approximated by a certain model function $Q(R)$, which we will call *quality-rate model* for a given codec and content.

We will assume that function $Q(R)$ is differentiable, monotonically increasing, and has range $[0,1]$, where 0 implies worst possible reproduction quality (e.g. nothing common with original content is delivered), and 1 implies that reproduction is perfect. The function $Q(R)$ should be selected such that $Q(0) = 0$, and $Q(\infty) = 1$.

2.2. Encoding ladder

By an *encoding ladder* we will understand an ordered set of n rate points: $R_1 < R_2 < \dots < R_n$ and associated quality levels $Q_1, \dots, Q_n$ characterizing encoded streams. We will say that ladder is *proper* if quality levels of encoded streams are coinciding with respective points of quality-rate function:

$$Q_i = Q(R_i),\ i = 1, \dots, n.$$

For convenience of notation, we will also assume that ladder can always be augmented by two extreme points: $R_0 = 0$, $Q_0 = 0$, and $R_{n+1} = \infty$, $Q_{n+1} = 1$.

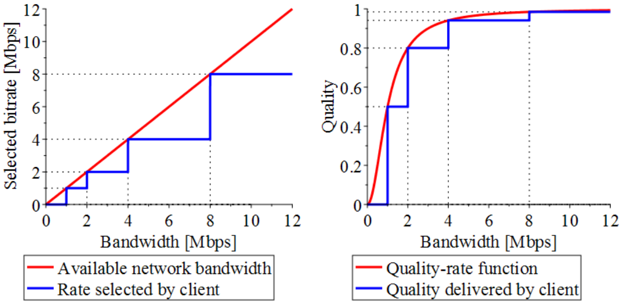


Figure 3. Model of a streaming client operating with ladder containing 1,2, 4, and 8-Mbps streams. Left figure shows rate selection logic. Right figure shows delivered quality.

2.3. Client model

We next define client model. As known from practice (see e.g. [11]), at certain points in time, the ABR streaming client estimates available bandwidth R, and then decides which of the encoded streams to pull next. The intent of such decisions is to enable continuous playback while utilizing most of the available bandwidth.

To model this behavior in a simplest possible way, we will assume that client always picks a stream with largest bitrate below or equal to the available bandwidth R:

$$R_{\text{selected}}(R) = \max_{i=0,\dots,n} R_i \leq R$$

This logic is illustrated in left subfigure in Figure 3. The right subfigure shows quality levels achievable by this model:

$$Q(R) = Q\big(R_{\text{selected}}(R)\big).$$

We will call this model – a *conservative client.* Note that when bandwidth is less that lowest ladder bitrate R_1, this model switches to 0, implying that client is buffering. This model is very simple, but sufficient for the purpose of average case analysis. We will discuss the use of some alternative models in Section 4.3.

2.4. Probabilities of loading of each stream

We next assume that network bandwidth can be modeled as a continuous random variable R with probability density function $p(R)$. Then, the probabilities of loading of each stream by conservative client can be obtained as follows:

$$p_i = \Pr(R_{selected}(R) = R_i) = \int_{R_i}^{R_{i+1}} p(R) dR.$$

Note, that same formula also produces p_0 – probability that client is buffering.

2.5. Average performance parameters

We summarize a set of parameters that can be defined for ABR systems in Table 1. Most of these parameters are well understood and commonly used in practice. However, the last two parameters are likely new. By *average quality limit* we will understand average quality theoretically achievable by streaming system using infinite ladder capturing all points of quality-rate function. Similarly, by *quality gap* we will understand a relative distance between delivered average quality and quality limit. These parameters are useful for understanding of the impact of limiting of the number of streams on the performance of the system.

Table 1. Performance parameters of ABR system

Parameter	Expression
Average bandwidth used for streaming	$\bar{R}(p, R_1, \ldots, R_n) = \sum_{i=i}^{n} p_i R_i$
Average network bandwidth	$\bar{B}(p) = \int_0^\infty R\, p(R)\, dR$
Bandwidth utilization	$\eta(p, R_1, \ldots, R_n) = \frac{\bar{R}(p, R_1, \ldots, R_n)}{\bar{B}(p)}$
Buffering probability	$p_0(p, R_1) = \int_0^{R_1} p(R) dR$
Average quality	$\bar{Q}(p, R_1, \ldots, R_n) = \sum_{i=1}^{n} p_i Q(R_i)$
Average quality limit	$Q^*(p) = \int_0^\infty Q(R)\, p(R)\, dR$
Quality gap	$\xi(p, R_1, \ldots, R_n) = \frac{Q^*(p) - \bar{Q}(p, R_1, \ldots, R_n)}{Q^*(p)}$

3. QUALITY-OPTIMAL LADDERS

3.1. The problem

We are now ready to pose the following problem: given quality-rate function $Q(R)$, bandwidth PDF $p(R)$, and rate limits $R_{\min}$, $R_{\max}$, and $R_{1,\max}$, first an n-point ladder $R_1^*, \ldots, R_n^*$, such that average quality delivered by ABR streaming system is maximal:

$$\bar{Q}(p, R_1^*, \ldots, R_n^*) = \max_{\substack{R_{\min} < R_1 \leq \cdots \leq R_n < R_{\max} \\ R_1 \leq R_{1,\max}}} \bar{Q}(p, R_1, \ldots, R_n).$$

We will call this problem *quality-optimal ladder* design problem. The constraint $R_{\min} < R_1 \leq \cdots \leq R_n < R_{\max}$ is imposed to ensure proper order and range of rate points that are selected. The upper limit on smallest rate $R_1 \leq R_{1,\max}$ is also imposed as it affects buffering probability.

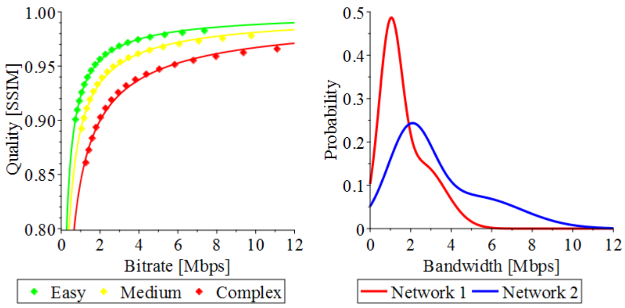


Figure 4. Quality-rate (left) and network (right) models used in our experiments.

As follows from definition, this problem belongs to a class of non-linear constrained optimization problems, and can be solved by existent techniques, such as sequential quadratic programming [12].

3.2. Examples of quality-optimal ladders

For the purpose of our experiments we use 3 video sequences, which we call “easy”, “medium”, and “complex” describing degrees of challenge that they present to the encoder. They were produced by catenating several 720p50 sequences available in [13]. In Figure 4, we show quality-rate functions obtained for these sequences by using x264 encoder [14] and SSIM quality metric [15]. The following model function is used:

$$Q_{\alpha,\beta}(R) = \frac{R^\beta}{\alpha^\beta + R^\beta}.$$

In Table 2, we show values of model parameters and accuracy achieved by such models.

Table 2. Parameters of quality-rate models.

Content	Model parameters		Model MSE
	α	β	
Easy	0.0555	0.8550	0.116e-5
Medium	0.0724	0.8016	0.371e-5
Complex	0.1015	0.7364	0.760e-5

Table 3. Parameters of network models.

Network	Model parameters				
	α	μ_1	σ_1	μ_2	σ_2
Network 1	0.584	0.996	0.564	2.554	1.165
Network 2	0.584	1.992	1.129	5.108	2.331

As network models, we used throughput measurements of LTE network [16], fitted to the following model:

$$p_{\alpha,\mu_1,\sigma_1,\mu_2,\sigma_2}(R) = \frac{\alpha f(R|\mu_1, \sigma_1) + (1-\alpha) f(R|\mu_2, \sigma_2)}{C}$$

where

$$f(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

is the probability density function of normal distribution, $\alpha, \mu_1, \sigma_1, \mu_2$, and σ_2 are model parameters, and C is the normalization constant accounting for the fact that R is non-negative. As shown Table 3 and Figure 4 (right), two models are obtained by scaling network throughput by two possible numbers of users in the LTE cell.

We next present quality-optimal ladders constructed for given models of content and networks. In all cases, we set rate limits $R_{\min} = 0.1$, $R_{\max} = 10$, and $R_{1,\max} = 0.4$, all in Mbps. The results are shown in Tables 4 and 5. Ladder bitrates (rounded to nearest kbps) are shown in column 3 of both tables. Last 3 columns show performance parameters of these ladders: quality achieved by n-th stream Q_n, average quality $\bar{Q}$, and quality gap ξ.

Table 4. Optimal ladders generated for network model 1.

Content	N	Ladder bitrates [kbps]	Q_n	$\bar{Q}$	ξ[%]
Easy	2	138, 803	0.909	0.867	6.58
	3	100, 512, 1209	0.931	0.888	4.35
	4	100, 411, 866, 1645	0.946	0.897	3.34
	5	100, 349, 694, 1155, 2087	0.955	0.902	2.76
Medium	2	175, 854	0.881	0.830	7.98
	3	100, 518, 1219	0.906	0.854	5.31
	4	100, 416, 876, 1663	0.924	0.866	4.00
	5	100, 354, 701, 1165, 2104	0.936	0.873	3.25
Complex	2	234, 931	0.825	0.769	10.2
	3	145, 590, 1304	0.867	0.797	6.96
	4	102, 431, 898, 1704	0.888	0.812	5.22
	5	100, 363, 716, 1183, 2134	0.904	0.821	4.16

Table 5. Optimal ladders generated for network model 2.

Content	n	Ladder bitrates [kbps]	Q_n	$\bar{Q}$	ξ[%]
Easy	2	232, 1457	0.940	0.906	5.14
	3	116, 811, 2124	0.955	0.924	3.27
	4	100, 589, 1421, 2803	0.964	0.932	2.40
	5	100, 486, 1107, 1974, 3577	0.971	0.937	1.92
Medium	2	293, 1549	0.920	0.878	6.23
	3	158, 893, 2216	0.939	0.899	4.04
	4	100, 601, 1438, 2828	0.949	0.909	2.97
	5	100, 495, 1123, 1995, 3615	0.958	0.915	2.35
Complex	2	391, 1685	0.887	0.833	7.98
	3	232, 1018, 2358	0.910	0.857	5.29
	4	156, 712, 1569, 3001	0.924	0.869	3.94
	5	114, 537, 1179, 2060, 3727	0.935	0.877	3.11

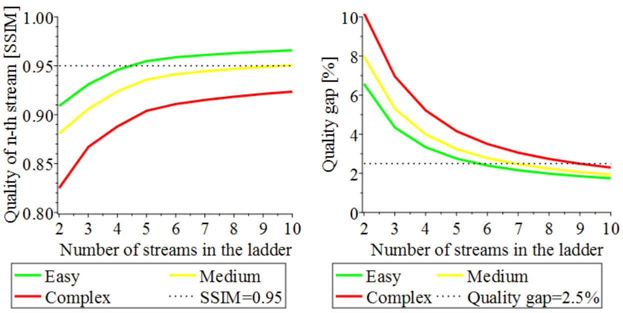


Figure 6. Impact of the number of streams on quality. Left: quality of n-th stream in the ladder. Right: quality gap.

3.3. Discussion

Considering ladders presented in Tables 4 and 5, we note that they are quite different for two network models that have been used. With first network model we see that most points are placed below or around 1Mbps. However, with second network model rate points are placed more sparsely, with most placed below or around 2Mbps.

We also note, that quality-optimal ladders are also different for different types of content. Ladders designed for more complex content use higher bitrates for respective ladder points. We also notice that at same number of streams, ladders designed for easy content deliver higher quality than ones for more complex content. This suggests that *the number of streams in ladders designed for complex and easy content should be different.*

To study this phenomenon more, in Figure 6, we present plots of quality achieved by n-th stream as well quality gap as functions of number of streams in the ladder. These plots are produced using 1st network model. It can be seen that our "easy' content can reach 0.95 SSIM at last level by using 5 streams, while "medium" content needs about 9, and with complex content many more are needed.

Quality gap can be used as an additional or alternative criterion for deciding how many ladder points to use. For example, it shows that 9-point ladder for "complex" content is just about 2.5% away in average quality from ladder with infinite number of points. Hence the adding more streams to the ladder is not going to help much.

4. VARIATIONS AND EXTENSIONS

The proposed set of models and optimization framework can be altered or extended in many ways.

For example, to account for multiple possible resolutions one can first compute quality-rate models $Q(S,R)$ for each specific resolution S, and then take upper boundary

$$Q(R) = \sup_{S\in\mathcal{S}} Q(S,R)$$

as final quality-rate function.

The client model can also be modified. Figure 7 shows an alternative model of a client that switches rates at some earlier decision points $T_i \leq R_i$. We call this model an "aggressive client".

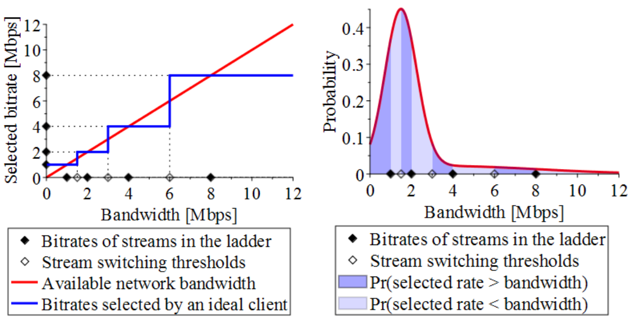


Figure 7. Aggressive client model. Left: rate selection logic. Right: probabilities when selected rates are above and below the available bandwidth.

The aggressive client model is feasible if

$$\int_{T_{i-1}}^{R_i}(R_i - R)p(R)dR \le \int_{R_i}^{T_i}(R - R_i)p(R)dR,$$

implying that the network bandwidth consumed by the client stays below one that is available.

Finally, we may be given *a set of networks* $\mathcal{W}$ with densities $p_w(R), w \in \mathcal{W}$, and asked to design a ladder considering delivery to any network in this set. One way to pose optimization problem in this case would be to find a ladder that delivers *best average quality in the worst case*:

$$\bar{Q}(R_1^*,\dots,R_n^*) = \max_{\substack{R_{\min}<R_1\le\dots\le R_n<R_{\max}\\ R_1\le R_{1,\max}}} \min_{w\in\mathcal{W}} \bar{Q}(p_w,R_1,\dots,R_n).$$

However, if one also knows relative *usage probabilities* across networks u_w, $\Sigma_{w\in\mathcal{W}}u_w = 1$ then the problem reduces to an earlier case by considering compound density:

$$p(R) = \sum\nolimits_{w\in\mathcal{W}} u_w p_w(R)\,.$$

5. REFERENCES

[1] D. Wu, Y.T. Hou, W. Zhu, Y-Q. Zhang, and J.M. Peha. Streaming video over the Internet: approaches and directions, *IEEE Transactions on Circuits and Systems for Video Technology*, 11(3): 282-300, 2001.

[2] B. Girod, M. Kalman, Y.J. Liang, and R. Zhang. Advances in channel-adaptive video streaming, *Wirel. Commun. Mob. Comput.*, 2: 573–584, 2002.

[3] G.J. Conklin, G.S. Greenbaum, K.O. Lillevold, A.F. Lippman, and Y.A. Reznik. Video coding for streaming media delivery on the Internet, *IEEE Transactions on Circuits and Systems for Video Technology*, 11(3): 269-281, 2001.

[4] R. Agarwal, et al. System and method for providing random access to a multimedia object over a network, *US Patent 6,314,466*, granted November 6, 2001, filed October 6, 1998.

[5] G. Greenbaum, et al. System and method for generating multiple synchronized encoded representations of media data, *US Patent 7,885,340*, granted February 8, 2011, priority Apr. 27, 1999.

[6] R. Pantos, and W. May. HTTP Live Streaming, RFC 8216, August 2017. https://tools.ietf.org/html/rfc8216

[7] ISO/IEC 23009-1:2012, Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats, ISO/IEC, 2012.

[8] Apple, Inc. Best Practices for Creating and Deploying HTTP Live Streaming Media for Apple Devices, Technical Note TN2224, August 02, 2016. https://developer.apple.com/library/content/technotes/tn2224/_index.html

[9] A. Aaron et al. Per-Title Encode Optimization, Netflix technology blog, Dec 14, 2015. https://medium.com/netflix-techblog/per-title-encode-optimization-7e99442b62a2

[10] J. Chakareski. *Rate-Distortion Optimized Packet Scheduling for Video Streaming: Optimizing video delivery in packet networks*, VDM Verlag, 2009.

[11] S. Hesse. Design of scheduling and rate-adaptation algorithms for adaptive HTTP streaming, In *Proc. SPIE 8856, Applications of Digital Image Processing XXXVI*, 88560M, 2013.

[12] J. Nocedal and S.J. Wright. *Numerical Optimization*, Springer, 2006.

[13] YUV video sequences. https://media.xiph.org/video/derf/

[14] x264 open source encoder project.
https://www.videolan.org/developers/x264.html

[15] Z. Wang, L. Lu, and A. C. Bovik. Video quality assessment based on structural distortion measurement, *Signal Processing: Image Communication*, 19(2): 121-132, 2004.

[16] J. Karlsson, and M. Riback. Initial field performance measurements of LTE, *Ericsson review*, 3, 2008.