Step Response Metric for Video Encoder Rate Control Characterisation

Ralf Globisch* and Keith L. Ferguson
CSIR Meraka Institute
Meiring Naude
Pretoria, South Africa
rglobisch@csir.co.za
kferguson@csir.co.za

Yago Sanchez and Thomas Schierl
Fraunhofer HHI
Einsteinufer 37
10587 Berlin, Germany
yago.sanchez@hhi.fraunhofer.de
thomas.schierl@hhi.fraunhofer.de

ABSTRACT

Adaptive streaming is a widely-used solution to avoid playback interruptions and compensate for insufficient bandwidth or momentary congestion. A temporary reduction in bitrate results in less or no video break-up and rebuffering. The rate control mechanism of a codec implementation allows the selection of a target bitrate and tries to meet this target within certain constraints e.g. number of frames, with a certain percentage overshoot. However the target bitrate is not always instantaneously achievable, and the number of bits required to encode a frame are highly content-dependent. In this paper the Codec Step Response metric is proposed to measure the effectiveness of a codec rate control at responding to instantaneous rate variations. This metric allows the measurements of the suitability of a rate control implementation for an application or access network.

CCS Concepts

•General and reference → Metrics; •Information systems → Multimedia streaming;

Keywords

media streaming;rate adaptation;metrics

1. INTRODUCTION

In real-time video delivery systems the encoder utilizes rate control (RC) techniques to adapt the average rate produced to match that of the communications network rate thereby ensuring continuous content flow suitable for video conferencing conversations [10] [11] [15] [16] [20]. Real-time systems require tight cooperation between the network behavior and the encoder to sustain the stream without dropping frames or re-buffering. Wireless networks, both cellular and 802.11 Wi-Fi standards, often exhibit significantly varying instantaneous effective throughput rates under congestion or radio channel fading conditions. Therefore the encoder RC for these real-time applications need to be selected or designed to respond appropriately.

It is common practice in the field to characterize the performance of RC mechanisms by rate-distortion (RD) analysis with the distortion or quality measured in peak signal-to-noise (PSNR). This metric is a suitable measurement for the steady state condition of the encoder through averaging the PSNR for a given average rate constraint. However, it does not characterize how the encoder adapts to (near) instantaneous disturbances in the network. The RD metric does not measure the transition capability from one steady state to another. Of particular importance is the adaptation to congestion where very fast rate back-off is required (multiplicative decrease). The most commonly used mechanism for RC during the transition is to discard video frames when the encoder cannot respond fast enough to the disturbance until a new steady state condition is established. The negative perceived quality of dropped frames is "stuttering" and/or the loss of visual object velocity perception in the video scene. Therefore a more appropriate measurement technique is required for the transition period, other than steady state RD.

The recovery from congestion (e.g. additive increase) is significantly less important as the perceptual difference between rapidly improving quality and steadily improving quality, is small, and is reserved for future work.

The scope of this paper is to propose using a natural log decay model as a generalized relative metric to measure and compare the response of different video codec rate controllers for their suitability to fast network adaptation under rate availability stress. The decay function models the rate during the transitional period between two steady state conditions. The typical scenario is the onset of congestion. It is desirable to determine the propensity of an encoder to drop frames under test conditions before it is deployed to a real-time application. The quicker the transition and the lower the rate cost of the transition, the lower the likelihood of dropped frames and therefore the higher the perceived quality. The purpose of the metric is to assist system designers in the selection of either the encoder type or the RC mode setting for a given encoder. It is proposed that the step response to a rate change (transition), together with the traditional RD metric (steady state), is sufficient to characterize an encoder RC capability. This paper focusses on the step response part and reveals both a time constant of a decay

*Ralf Globisch is a guest researcher at Fraunhofer HHI.

NOSSDAV'16, May 13 2016, Klagenfurt, Austria

DOI: http://dx.doi.org/10.1145/2910642.2910646

function and the added average rate required to switch from one steady state rate condition to another.

The contributions of this work are as follows: the Codec Step Response (CSR) metric, a binary composite metric, is proposed which allows encoder RC to be evaluated according to the time required for a rate change, and the bit overuse during this period. We evaluate the RC in widely used H.264 codecs x264 [1] and OpenH264 [7], as well as the Video Processing Project (VPP) [8] H264v2 codec, under stress and show how the metric is useful to characterise RC response to instantaneous rate variations allowing developers and researchers to evaluate and improve RC algorithms and implementations.

The structure of this paper is as follows: Section 2 discusses related work. In section 3 the decay function is formulated, and the rate cost is discussed. In Section 4 the experimental setup and results are discussed, followed by an analysis in Section 5 and the conclusion in Section 6.

2. BACKGROUND

2.1 Rate Control

The majority of video encoders in popular use achieve their compression efficiency through motion prediction, macroblock transform coding, coefficient quantization and variable length entropy coding [2] [3] [4]. The control of the total bits on a per picture frame basis functions at the macroblock level where the quantization step size is either increased for fewer bits (lower quality) or decreased for more bits (higher quality). However, the exact number of bits produced is not deterministic and is dependent largely on the motion and spatial activity within the video scene.

The objective of encoder RC is to minimize the distortion subject to a rate constraint. The rate constraint is typically defined as an average rate over some short time period and the distortion is often a mean absolute difference measure. The rate produced by a picture encoding is controlled by determining the fullness of a simulated buffer to give a desired rate for the next picture. An RD model from previous encodings is used to predict the distortion parameter (quantization step size) to use on the next encoding.

Where encoders differ is in the model that is used and the boundary conditions of those models. Further, the choice of distortion measure and rate measure in the model can differ. The most popular model used is the inverse quadratic model where the distortion variable is the average quantization parameter (related to the quantization step size). This is a very effective model due to both its simplicity and accuracy [6] [12] [13] [14] [18]. Other models that are used are the inverse natural log and power models where the distortion measure is the peak mean squared difference [8]. For accurate instantaneous rate constraint encoders, slower discrete gradient descent algorithms are used [8]. These differing RC mechanisms exhibit varying performances under real-time media communication conditions. Understanding their behavior assists in determining the wider system response times to significant disturbances in the network.

2.2 Adaptive Real-time Streaming

In a real-time communication session, the sender needs to adjust the outgoing bitrate to match the available bandwidth. Link overuse typically results in packet loss and increased delays. Adaptive streaming techniques rely on implicit or explicit feedback to reduce and increase the sending rate accordingly [5] [15] [16] [19]. As part of the ongoing RTCWeb standardisation effort various congestion control (CC) algorithms have been proposed which enable adaptive streaming: the Google Congestion Control [10] which is a mixture of loss and delay-based CC, NADA [20] which uses implicit or explicit feedback CC, and Scream, a self-clocked rate adaptation algorithm [11]. While these works focus on how the rate needs to be adapted, they do not take into account how the actual RC algorithm implementation is able to respond to the rate adjustment which is the focus of this work.

3. COMPARISON METHODOLOGY

3.1 Decay Function Formulation

To stress test the selected encoder RC, test sequences were encoded at a starting rate R_0, and some switch point, the rate was changed to the target rate R_1. To model the behaviour of the encoder while transitioning from one steady state to another, the decay function has been proposed as follows:

$$r_m(t) = R_1 + (R_0 - R_1)e^{-\frac{t-t_0}{\tau}}$$
$$\ln[\frac{r_m(t) - R_1}{R_0 - R_1}] = -\frac{t - t_0}{\tau}$$
$$y_m(t) = -(t - t_0)\tau^{-1}$$

where $r_m(t)$ is the rate r of the model m at time t, τ is the time constant, R_0 is the initial rate, R_1 is the target rate, t_0 is the switching point at which the rate is reduced from R_0 to R_1 as illustrated in Figure 2. $y_m(t)$ has been introduced as a linearisation substitution variable.

The curve fitting objective function is defined as:

$$S = \sum_i^N (y_s(i) - y_m(i))^2 = \sum_i^N (y_s(i) + (t_i - t_0)\tau^{-1})^2$$

where the subscript s denotes the actual sample values. Minimising the objective function with respect to τ yields:

$$\frac{\delta S}{\delta \tau} = \sum_i^N (y_s(i).(t_i - t_0) + \tau^{-1}(t_i - t_0)^2) = 0$$

Solving for τ and resubstituting y gives:

$$\begin{aligned} \tau &= -\frac{\sum_i^N (t_i - t_0)^2}{\sum_i^N y_s(i).(t_i - t_0)} \\ &= -\frac{\sum_i^N (t_i - t_0)^2}{\sum_i^N (\ln[\frac{r_s(t)-R_1}{R_0-R_1}]).(t_i - t_0)} \end{aligned} \quad (1)$$

Finally, the boundary condition $r_s(i) > R_1$ needs to be imposed for all i: $\ln[\frac{|r_s(i)-R_1|}{(R_0-R_1)}]$ which results in no loss of generality.

3.2 Time Constant Rate Cost

The time constant indicates the decay time but not the cost of this decay. If the RC is assumed to be under stress then a measure is required to show the stress level. The average rate above the target after the step function is such a measure that indicates the quality of the recovery or convergence. The ratio of the additional bits above the target bits relative to the target bits is defined as the time constant rate cost c which is defined in Equation 2. Measuring this for the time constant period provides a further understanding of what the rate stress level is with the time constant providing its duration. The added burden on the network based on the time constant and the cost of this period must be considered.

$$c = \frac{\left(\frac{1}{\tau}\sum_{i=t_0}^{\tau+t_0}[r_{s(i)} - R1]\right)}{R1} \tag{2}$$

Further, the rate cost ratio (RCR) is defined as the average bit overuse during the interval τ following the switch to the target rate R_1 in relation to R_1.

4. EXPERIMENTAL SETUP AND RESULTS

The application architecture is shown in Figure 1. Media from a live source is encoded in real-time by an encoder module which has an RC function. The transmission module sends encoded media over the network from which it receives reception feedback which is passed to the Cooperative Function to determine a target rate and configure the encoder accordingly. The Cooperative Function is a component that enables cooperation between the codec and the network.

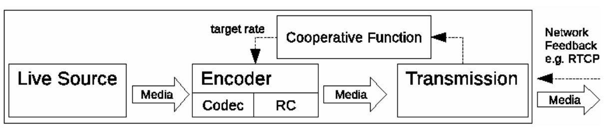

Figure 1: Experimental setup.

Four Common Intermediate Format (CIF) 352 x 288 pixel (pel) resolution video clips and two High Definition (HD) 1280x720 pel resolution clips were selected for the experiments. Three of the CIF clips were *Soccer*, *Foreman* and *Mobile* selected for their fast, medium and slow scene movement, respectively. The fourth CIF clip was a cropped and scaled version of the *Four People* HD sequence. The HD clips were *Four People* and *Mobile Calendar* selected for their low and high spatial complexity, respectively and *Four People* for its video conferencing applicability. The four CIF and the *Four People* HD were 10 sec video clips at 30 frames/s (300 frames). The *Calendar* HD clip was 10 sec at 25 frames/s (250 frames).

The rate measure is normalized in this paper to bits/pel (bpp) where the picture resolution (pel/frame), frame rate (frames/s) and bit rate (bits/s) are combined to generalize the results for all picture sizes and frame rates. The selection of HD and CIF in the experiments was designed to verify this assumption by selecting the same bpp rate constraint for both cases. All the encoders were expected to be under stress at 0.035 bits/pel, particularly with significant scene movement. They were also not expected to be under stress at 0.140 bits/pel. Therefore the experimental step size was selected to be from 0.140 bits/pel to 0.035 bits/pel. This equates to a bit rate step of approximately 426 kbits/s to 106 kbits/s for the CIF clips and 3.87 Mbits/s to 968 kbits/s for the HD clips at 30 frames/s.

Two switching points for some of the clips were selected to ensure that the results were independent of the switching point. Some care was required in selecting the switching points such that there was sufficient time for the encoders to be in a steady state condition prior and post the step for these relatively short 10 sec clips. To this end, the step (t_0) was applied at frame 100 and frame 150 for the 30 frames/s clips and at frame 125 for the 25 frames/s clip.

The selected test sequences were encoded with OpenH264 [7], x264 [1], one of the most efficient H.264 implementations [17], and VPP [8] H264v2 codec in modes VPP-Cbr, Vpp-Log and VPP-Pow. Automatic IDR-frame insertion and frame skipping were disabled on all codecs. In the interest of repeatability, the codec configuration settings have been made available [9]. The five encoder outputs' bits/pel and the associated time for each frame for all the sequences were collected. Each of these was plotted and the convergence point (N) was determined by subtracting the steady state rate plot without the step function providing a normalizing reference. The subtraction of the reference leaves the approximate rate decay points as the residual from which the decay function is derived. This is the point at which the encoding results move from the transition phase into the steady state phase. This procedure provides limits to the curve fitting process for improved accuracy close to the switching point. The data was fitted to the decay function and the time constant was calculated using Equation 1, above. The average increase in rate within each time constant period from the switching point was determined and the rate cost calculated.

The high and low rate steady state behaviour was determined by averaging the rate and PSNR for approximately 120 frames prior to the step (high) and the final 100 frames (low), respectively. It was assumed that the RC would be in the high rate steady state just prior to the step. The last 100 frames appeared sufficiently far from the converged step response to be assumed to be in the low steady state.

The plot for the x264 encoder on the *Four People* HD video clip shown in Figure 2 below illustrates the experimental process that was applied to all the video samples. The fitted decay function and the target average rate is also included.

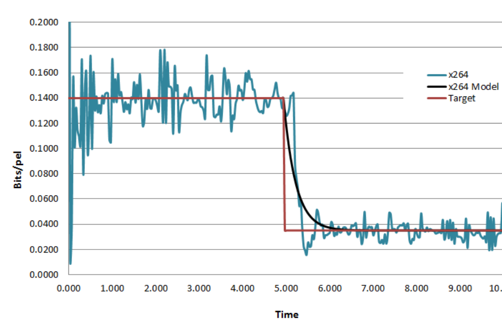

Figure 2: *Four People* Rate Decay Plots with x264 Response.

A summary of the rate decay time constants (τ) and the RCR for each encoder and video sequence for the mid-clip switching point (frame 150) is presented in Table 1 below. The results for switching point at frame 100 are given in Table 2. Note that the RCR in this table is the post step function period to the end of the video sequence.

A summary of the steady state RD performance of the encoders both prior to the step function (high rate) and after convergence (low rate) are shown in Table 3.

Table 1: Summary of Encoder Time Constants and Total Rate Cost Ratio (Mid point switch)

	Soccer		Foreman		Mobile		Four People CIF		Four People HD		Calendar HD	
Encoder	τ (s)	**RCR**	τ (s)	**RCR**	τ (s)	**RCR**	τ (s)	**RCR**	τ (s)	**RCR**	τ (s)	**RCR**
x264	0.29	3.21	0.26	3.14	0.16	2.95	0.32	2.71	0.28	2.68	0.23	2.41
OpenH264	11.07	2.52	0.35	2.21	-35.89	3.55	0.33	0.42	0.04	0.08	0.04	0.18
VPP-Log	0.60	1.59	0.27	2.15	0.23	0.72	0.17	1.13	0.19	1.04	0.19	0.95
VPP-Pow	0.58	1.62	0.22	0.85	0.12	0.00	0.09	-0.70	0.08	-0.57	0.04	-0.23
VPP-Cbr	0.01	-0.06	0.02	-0.02	0.01	-0.01	0.02	-0.02	0.01	-0.01	0.01	-0.01

Table 2: Summary of Encoder Time Constants and Total Rate Cost Ratio ($\frac{1}{3}$ point switch)

	Soccer		Foreman		Mobile		Four People HD	
Encoder	τ (s)	**RCR**	τ (s)	**RCR**	τ (s)	**RCR**	τ (s)	**RCR**
x264	0.29	2.48	0.26	2.23	0.26	1.59	0.41	1.53
OpenH264	15.21	2.41	0.24	0.17	-33.76	3.78	0.07	-0.06
VPP-Log	1.96	2.29	0.27	0.41	0.31	0.70	0.42	0.57
VPP-Pow	0.78	1.30	0.19	0.02	0.26	0.12	0.19	-0.53
VPP-Cbr	0.02	-0.02	0.01	-0.01	0.01	-0.02	0.01	-0.01

Table 3: Steady State Encoder RD at High Rate (Target 0.140 bpp) and Low Rate (Target 0.035 bpp)

	Soccer		Foreman		Mobile		Four People HD		Calendar HD		Encoder Avg	
Encoder	**Rate**	**PSNR**	**Rate**	**PSNR**	**Rate**	**PSNR**	**Rate**	**PSNR**	**Rate**	**PSNR**	**Rate**	**PSNR**
x264	0.136	31.27	0.144	32.43	0.140	21.58	0.140	40.40	0.138	29.12	0.139	30.96
OpenH264	0.140	33.32	0.141	34.82	0.190	27.32	0.139	40.21	0.138	32.67	0.150	33.67
VPP-Log	0.149	30.26	0.145	34.27	0.141	25.34	0.141	40.82	0.141	32.00	0.143	32.54
VPP-Pow	0.146	30.21	0.148	34.38	0.142	25.26	0.142	40.98	0.145	32.29	0.144	32.63
VPP-Cbr	0.139	30.02	0.139	34.21	0.139	25.24	0.139	40.85	0.139	32.20	0.139	32.50
x264	0.035	27.03	0.033	25.79	0.034	18.84	0.034	37.11	0.034	28.35	0.034	27.43
OpenH264	0.114	31.83	0.048	29.51	0.156	27.01	0.035	38.28	0.050	32.05	0.081	31.74
VPP-Log	0.073	26.59	0.043	23.78	0.045	19.90	0.033	37.47	0.065	25.71	0.052	26.69
VPP-Pow	0.071	26.59	0.043	23.77	0.045	19.82	0.035	37.69	0.065	25.68	0.052	26.71
VPP-Cbr	0.033	22.51	0.034	22.73	0.035	19.67	0.035	37.86	0.034	22.66	0.034	25.08

From the tables above, the VPP-Cbr encoder reacts instantaneously to the stress step function in all cases, as expected, with no significant rate cost regardless of the video scene activity, the picture dimensions or frame rate. The τ variation of between 10-20 msec for all video sample cases verifies this behaviour. Further, it is designed not to exceed the target rate from frame to frame that can be observed by noting that it remains below the target rate throughout the 10 sec sequences both during the stress and stress free periods. All RCRs are less than or equal to zero. However, the trade-off that this accurate rate matching forfeits is the steady state quality at low rates. It was observed that the steady state PSNR was the lowest for all the encoders under test at 0.035 bpp.

The OpenH264 encoder exhibited instability in both the convergence in the transition period and in the steady state conditions. The tables indicate large τ values for the fast motion *Soccer* sequences pertaining to difficulty in converging to the lower rate steady state condition. The large negative τ values for the low motion *Mobile* sequence indicates divergence showing an increase in rate during the stress period! However, for the cases where it appears stable, as in the *Four People* and *Mobile Calendar* sequences it out-performs the other encoders as indicated by the low τ values coupled with the low rate cost values. Further, its steady state performance has significantly better quality in the higher PSNR values, although it has difficulty reaching the low rate target. The unstable results were discarded for the purposes of the comparison analysis.

Both the VPP-Log and Pow encoders showed consistently stable results but they both suffered from rate overshoot and undershoot immediately after convergence. This behaviour was also exhibited during the steady state stress period.

The x264 encoder steady state RD performance was better than the other encoders after the rate step. However, its response time to the rate step appeared delayed. This can both be observed in Figure 2 above and in the tables where small values of τ are accompanied by a relatively large RCR. The rate cost was significantly higher than any of the converged encoders.

5. ANALYSIS

The proposed step response characterization put forward in this paper requires verification of consistency across different video sequences (content), their picture size and frame rates. The point of interest is in the short time neighborhood of the rate step. The steady state RD performance in the pre and post rate step periods is not considered here.

The *Four People* HD video sequence was scaled to CIF with some prior cropping from the left and right boundaries to maintain the correct aspect ratio. Some scene information was lost in the cropping and therefore it is not identical but is considered sufficient for the test. Comparing the results of these two sequences in Table 1, the differences observed in the τ-rate cost pairs for all but one encoder are consistent. The x264 encoder has a τ difference of 0.04 with a RCR

Table 4: Averaged Encoder τ and Total Rate Cost Ratio

	All Sequences	
Encoder	**τ (s)**	**RCR**
x264	0.27	2.47
OpenH264	0.15	0.51
VPP-Log	0.50	1.16
VPP-Pow	0.27	0.29
VPP-Cbr	0.01	-0.02

difference of 0.03 between the CIF and HD cases. Similarly, VPP-Log has a 0.02 and 0.09 τ and rate cost difference, respectively. The OpenH264 encoder was the exception with indications of convergence instability for the CIF sequence. This was also observed in one HD sequence not presented in this paper due to space contraints.

Table 1 and Table 2 provide results for the same video clip but with the rate step stimulus shifted in time to another short time neighborhood. It is reasoned that if the encoded information is sufficiently different (e.g different direction motion or object occlusion, etc.) around one switch point to that of another in the same video then they may be considered as entirely different test cases. However, the ranking of the encoders should remain nearly the same as would be expected across different clips. This is because it is surmised that each encoder RC technique has different step responses depending on scene content and time position in the sequence but their relative performance is consistent. If one encoder is better than another then it is always better regardless of time position or scene.

Comparing the switching points of frame 150 (in Table 1) with its clip counterpart for frame 100 (in Table 2), it is noted that the ranking remains consistent. For example, in the *Soccer* clip the ranking of τ results in ascending order are VPP-Cbr, x264, VPP-Pow and VPP-Log. The rank is the same for both switching points.

Having empirically established some confidence in the consistency of the CSR metric across several video sequence parameters, the key issue is then to determine the relative performance of the encoders under the experimental conditions described here. The τ and rate cost values were averaged across all the tests for each encoder. These are presented in Table 4 below. It must be noted that the OpenH264 anomalies were removed from the averaging.

It is instructive to plot these two-dimensional characterization points graphically to determine their relative step response performance. The scatter plot is shown in Figure 3 below.

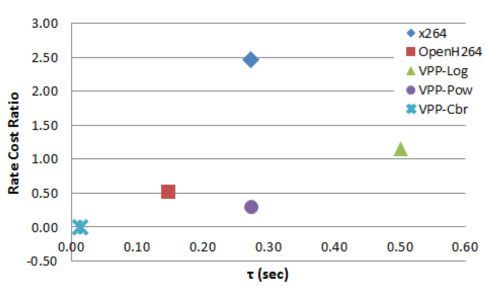

Figure 3: Transition Encoder Response Comparison.

The closer the plotted point is to the y-axis, the faster the encoder's ability to react to disturbances (small τ). Simultaneously, the closer the point is to the x-axis, the smaller the additional bit rate consumed to achieve convergence to a new rate (small RCR). Both these parameters are simultaneously responsible for whether or not encoder buffers will overflow and require frame dropping to re-establish a steady state condition. Therefore, the closeness to the origin of the graph is the optimal solution. However, it is reiterated that although VPP-Cbr is close to optimal in terms of its step response, it has sacrificed significant steady state distortion performance at the low rates to achieve this.

The OpenH264 encoder is the next closest to the origin but often has unstable behavior after the step function is applied. Recall that this point was determined by excluding the unstable results.

The VPP-Pow encoder and the x264 encoder have similar average response times but the x264 encoder consumes significantly more bits during the decay. Hence the x264 encoder is further from the origin than the VPP-Pow result. The behavior of the RC observed with the data collected provides evidence for this. The x264 encoder does not respond immediately to the step and continues at the higher steady state rate for a short period therefore producing the high bit cost. On the other hand, the VPP-Pow tends to undershoot the target average rate after converging. The x264 descent is faster but delayed and VPP-Log is slower but compensates for the extra descent bit rate by undershooting.

Table 5: Averaged Steady State RD

	High Rate		Low Rate	
Encoder	**Rate**	**PSNR**	**Rate**	**PSNR**
x264	0.139	30.96	0.034	27.43
OpenH264	0.150	33.67	0.081	31.74
VPP-Log	0.143	32.54	0.052	26.69
VPP-Pow	0.144	32.63	0.052	26.71
VPP-Cbr	0.139	32.50	0.034	25.08

The steady state rate and distortion for the high pre step function rate and the low post convergence rate were determined provide further insight into the RC behaviour of the encoders under test. The averaged values (excluding the OpenH264 unstable results) are presented in Table 5. This table is plotted with the low and high target rates indicated as dotted vertical lines in Figure 4 below.

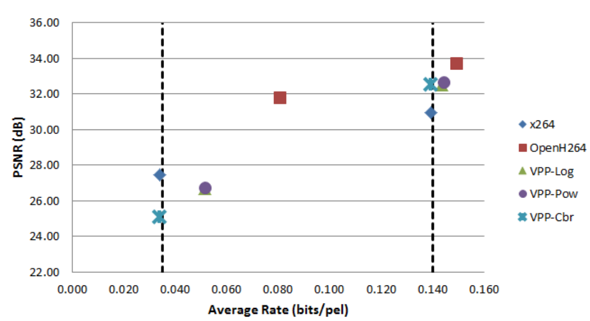

Figure 4: Encoder Steady State RD Comparison.

Firstly, it should be noted that the impact of small rate and quality differences of the encoders at high rates is not significant from a viewer perception perspective. In the light of this, at the higher rate of 0.140bpp the relative performance of the RC is largely defined by their ability to accurately track the target rate. With the encoders not under stress, the x264 and VPP-Cbr encoder demonstrate greater average rate accuracy than the others. The VPP-Cbr encoder had an average quality advantage of approximately 2.5dB over the x264 encoder.

Of particular interest is the relative performance of the encoders under stress after a network disturbance. At the low rate 0.035bpp it is evident from Figure 4 above that the OpenH264 and both VPP-Log and Pow encoders were unable to achieve a steady state condition close to the target rate. The poor average performance of OpenH264 can be attributed to the influence of its instability. It performed very well when stable (see Table 5), but on average, it cannot be relied upon to converge under all conditions. Again, x264 and VPP-Cbr are able to meet the low rate steady state requirement under stress. The more than 2dB quality improvement of the x264 encoder over the VPP-Cbr encoder is perceptually significant at these low rates.

The overall performance of the x264 encoder is considered to have the best average steady state performance of all the encoders under test. Although the VPP-Cbr encoder has a greater high rate quality, it is less significant than the substantial low rate advantage of the x264 encoder.

The CSR metric applied to the given set of encoders under test resulted in the performance ranking for transient disturbances to be from the best performance downwards; VPP-Cbr, OpenH264 (when stable), VPP-Pow, x264, VPP-Log and finally OpenH264 (on average). The steady state performance ranking order was the x264, VPP-Cbr, VPP-Log and Pow and then OpenH264 (on average). These results indicate that the steady state capabilities of encoder RC mechanisms do not reflect their performance under network disturbances and therefore a different measure, as proposed here, is required for this case.

It is the context of a design in the intended operating environment and the constraints of the system application that determines what the trade-offs are between the transient switching behaviour and the steady state RD performance. The perceived quality impact to the viewer is dropped frames if the response to network disturbances is too slow or too many bits are consumed in the response. The impact is poor picture quality if the RD performance outside of the disturbances does not meet the application requirements.

6. CONCLUSION

In this paper we have proposed the Codec Step Response metric to evaluate the performance of rate control implementations in their capability to meet instantaneous rate requirements as prevalent during network congestion or disturbances. We showed how effective the CSR metric is at evaluating state of the art codecs x264, OpenH264 and the VPP H264v2 codec. Evidence has been provided showing the CSR metric to be consistent across a range of video content, switching points, picture resolutions and frame rates.

The results indicate that the VPP-Cbr is the most effective for stringent instantaneous rate requirements but sacrifices picture quality to achieve this. The OpenH264 codec did not converge for all tests but had the next best step response when converged. These were followed by the VPP-Pow, x264 codec and VPP-Log, in performance order. It was further shown that the steady state RD performance of the same encoders is different to their network disturbance performance justifying an alternative metric. VPP-Cbr had the best overall step response performance and x264 the best overall steady state performance.

7. REFERENCES

[1] x264. http://www.videolan.org/developers/x264.html.

[2] Video coding for low bit rate communication. ITU-T H.263, 2005.

[3] Advanced video coding for generic audiovisual services. ITU-T H.264, 2014.

[4] High efficiency video coding. ITU-T H.265, 2015.

[5] N. Baldo, U. Horn, M. Kampmann, and F. Hartung. RTCP feedback based transmission rate control for 3G wireless multimedia streaming. In *Personal, Indoor and Mobile Radio Communications, 15th IEEE International Symp. on*, 2004.

[6] T. Chiang and Y.-Q. Zhang. A new rate control scheme using quadratic rate distortion model. *Circuits and Systems for Video Technology, IEEE Trans.*, 1997.

[7] Cisco. Openh264, 2015. https://github.com/cisco/openh264.

[8] CSIR Meraka Institute. Video Processing Project. http://sourceforge.net/projects/videoprocessing/.

[9] CSIR Meraka Institute. Codec Step Response, 2016. https://github.com/CSIR-RTVC/CodecStepResponse.

[10] S. Holmer, H. Lundin, G. Carlucci, L. D. Cicco, and S. Mascolo. A Google Congestion Control Algorithm for Real-Time Communication. draft-alvestrand-rmcat-congestion-03, 2015.

[11] I. Johansson. Self-clocked Rate Adaptation for Conversational Video in LTE. In *Proc. of the 2014 ACM SIGCOMM Workshop on Capacity Sharing Workshop*. ACM, 2014.

[12] H.-J. Lee, T. Chiang, and Y.-Q. Zhang. Scalable rate control for mpeg-4 video. *Circuits and Systems for Video Technology, IEEE Trans. on*, Sep 2000.

[13] Z. Li, F. Pan, K. Lim, X. Lin, and S. Rahardja. Adaptive rate control for h.264. In *Image Processing, 2004. ICIP '04. 2004 Int. Conf. on*, Oct 2004.

[14] J. Ronda, M. Eckert, F. Jaureguizar, and N. Garcia. Rate control and bit allocation for mpeg-4. *Circuits and Systems for Video Technology, IEEE Trans.*, 1999.

[15] T. Schierl, T. Wiegand, and M. Kampmann. 3GPP compliant adaptive wireless video streaming using H.264/AVC. In *Image Processing, 2005. ICIP 2005. IEEE Int. Conf. on*, Sept 2005.

[16] V. Singh, J. Ott, and I. D. D. Curcio. Rate Adaptation for Conversational 3G Video. In *Proc. of the 28th IEEE Int. Conf. on Computer Communications Workshops*, INFOCOM'09, Piscataway, NJ, USA, 2009. IEEE Press.

[17] D. Vatolin, D. Kulikov, and A. Parshin. MPEG-4 AVC/H.264 Video Codecs Comparison. Technical report, 2010.

[18] A. Vetro, H. Sun, and Y. Wang. Mpeg-4 rate control for multiple video objects. *Circuits and Systems for Video Technology, IEEE Trans. on*, Feb 1999.

[19] C. Wanxiang and L. Zhenming. An modified RTP adaptive algorithm. In *Info-tech and Info-net, 2001. Proc. ICII 2001. Int. Conf. on*, 2001.

[20] X. Zhu and R. Pan. NADA: A Unified Congestion Control Scheme for Low-Latency Interactive Video. In *Packet Video Workshop (PV), 20th Int*, Dec 2013.