Complexity-Driven Rate-Control for Parallel HEVC Coding

Juliette Carter[1], Saverio Blasi, and Marta Mrak

British Broadcasting Corporation, Research and Development Department, London, UK

[1]juliette.carter@bbc.co.uk

Abstract—**When using adaptive streaming, the content needs to be segmented so that clients can seamlessly switch to different rates depending on network conditions. On the video server each segment is stored in various bitrate representations, which are in practice provided by very fast encoders. Such encoders rely on parallelisation strategies to limit the encoder complexity. Parallelisation strongly affects the performance of rate-control (RC) algorithms, since different segments and parts of segments are encoded independently from each other. A new approach is proposed in this paper to tackle these issues, based on the optimisation of the initial parameters of a state-of-the-art RC model for inter-predicted frames in an HEVC/H.265 codec. The model makes use of an estimate of the texture complexity of the first frame in the segment to efficiently tune the parameters depending on the target rate. The approach is consistently improving the accuracy of RC schemes as well as the visual quality, with negligible impact on the encoding efficiency.**

***Index Terms*— Rate-control, HEVC, video streaming, DASH**

I. Introduction

The state-of-the-art H.265/High Efficiency Video Coding (HEVC) standard was developed to provide remarkable compression efficiency, necessary to enable new services such as delivery of content in Ultra High Definition (UHD) format [1][2]. HEVC provides high efficiency thanks to a variety of possible coding modes that can be adaptively selected to match underlying content. Hence, HEVC encoding can potentially be extremely complex, especially for UHD content, and therefore low complexity HEVC implementations are needed. To reduce complexity, many encoding processes are performed in parallel, either at a frame, or even at Coding Tree Unit (CTU, a square block of pixels of fixed size) level.

For the transmission of content over the internet, appropriate streaming technology is required. Dynamic Adaptive Streaming over HTTP (DASH) is an increasingly popular standard to enable adaptive streaming over HTTP [4]. The media content is segmented, encoded at different bitrates and resolutions, and stored on a server. The client adapts to dynamic network conditions and can switch to different versions of the same content at different bitrates. In order to meet the requirements of adaptive streaming technology as well as respond to constraints of storage space and transmission bandwidth, very high compression efficiency is required, as well as accurate rate-control (RC) schemes to allow the encoder to efficiently exploit the available resources [5]. However, processes to speed up the encoding such as parallel encoding of segments, frames and CTUs, can drastically affect the accuracy of an RC algorithm, as they prevent the encoder from collecting information for adaptation in case of inaccurate spending of bits. Moreover, the adaptive streaming framework imposes that no frame within a segment can be processed using information from other segments, and parameters cannot be adjusted from segment to segment. This can greatly affect the accuracy of an RC algorithm.

A novel approach is proposed in this paper to tackle these issues, based on the optimisation of the initial parameters of a state-of-the-art RC model for inter-coding using HEVC. The texture complexity is measured and used together with the target bitrate to model the initial values of RC parameters. This method was tested on a set of sequences at various bitrates under constraints dictated by adaptive streaming. The tests, show that the accuracy in reaching the target bitrate improves consistently with respect to RC schemes designed without considering parallelisation, reaching a satisfactory difference of 6.73 % of the target bitrate.

The rest of the paper is organized as follows. Section II presents the background and related work; the proposed adaptive RC model is described in Section III, and the experimental results are presented in Section IV. Finally, Section V concludes the paper.

II. Background and Related Work

RC theory is a well-studied problem, with numerous solutions proposed in the literature and successfully applied in practice. With wider availability of reliable network connections, higher bitrates can transmit large resolution content in high quality. Modern RC algorithms have to adapt to this changing landscape, while also performing the necessary decisions (in terms of setting the appropriate coding parameters to meet the target rate) with limited computational complexity.

In order to meet the given target bitrate, RC schemes must first perform an appropriate distribution of target bits, to correctly allocate target bitrates at a frame and CTU level. Then, given the target bits for a given CTU or frame, the RC scheme must tune the encoding parameters to meet the target. Many algorithms have been proposed based on the relationship between the rate and Quantisation Parameter (QP), referred to as R-Q models [6][7]. Other approaches investigate the relationship between rate and the percentage of zeros in quantised coefficients [8]. The utilisation of various encoding decisions in fast and efficient video coders, however, affect the reliability of models. For that reason, more accurate models have been proposed which take into account the rate-distortion (RD) decisions performed while encoding. In order to achievehigh compression efficiency, many encoder implementations select the optimal option based on RD optimisation. A cost J is computed for each option, typically as:

$$J = D + \lambda \cdot R \tag{1}$$

where D is the distortion between the original and reconstructed content when using the currently tested option, R is the corresponding rate needed to encode that option, and λ is the Lagrangian multiplier used in the optimisation process. Higher values of λ assign a larger weight to the rate component, which means options which result in smaller rates (at the cost of lower qualities) may be selected more often [9].

The correlation between rate and λ is robust, as shown in [5] and can be expressed as:

$$\lambda = \alpha \cdot R^{\beta} \tag{2}$$

where α and β are model parameters which are dependent on visual characteristics of the content and are updated online during the encoding. The update uses information on the actual rate that was spent on given CTUs or frames, to adapt to changing visual characteristics of the sequence. The optimal quantization parameter QP to meet the target bits can then be computed using a fixed relationship with λ as in [10]:

$$QP = 4.2005 \cdot \ln(\lambda) + 13.7122 \tag{3}$$

The R-λ model proposed in [5] considers fixed initial values for α and β. The authors suggest that these values are not decisive, as both parameters are updated during the encoding. Unfortunately, this is not the case when dealing with practical constraints such as segmentation or parallel processing. Under these challenging conditions, less information is available to update the parameters while encoding. Additionally, parameters must be initialised when starting each new segment.

The importance of determining optimal initial parameters in RC has already been investigated. In the context of R-Q models, an algorithm [11] was presented using spatial frame complexity to determine a content-dependent initial QP for real-time applications. Temporal complexity was also used [12]. In the context of R-λ models, a two-pass RC scheme was proposed [13], highlighting the importance of having content-dependent initial parameters. Unfortunately, two-pass approaches may not be suitable for some applications in which low complexity is crucial. In [14], the R-λ model is extended by using a content-dependent value of α. This method, however, is only applied to intra frames.

III. RC Parameter Scheme

Various distribution scenarios use coding of segments, typically a few seconds in length [4]. There must be no interdependency between frames of different segments, so that each segment can be encoded in parallel. When parallelisation is used within a segment, the information that can be collected and used by the RC algorithm is restricted. For example, when starting the encoding of a frame, it may not be possible to know how many bits were spent on previous frames (as they may not have finished encoding). Similarly, as a CTU can start encoding as soon as the necessary portion of the reference frame has been encoded, it may not be possible to extract information on the bits spent in neighbouring CTUs. Therefore, the initial values required for RC to be efficient (α and β from (2)) have a stronger impact, as the encoder is slower to adapt.

Common HEVC implementations make use of a fixed Structure of Pictures (SOP), periodically used throughout the encoded sequence. The SOP specifies the hierarchical relationship among frames and their encoding order (which may be different to the display order, identified in HEVC by the Picture Order Count, POC), as shown in Fig. 1. For each frame, α and β are updated, making use of information extracted from previously encoded frames at the same hierarchical layer. However, when parallel processing is used, the initial α and β cannot be updated during a single SOP encoding (with the exception of the initial intra-frame, which is treated separately). In order to evaluate the effects of parallelisation and segmentation on the accuracy of the RC scheme, a number of sequences with different visual characteristics were encoded, all in the UHD format (with 3840 × 2160 luma samples), 4:2:0 chroma subsampling, 8 bit-depth and 50 or 60 frames per second (fps) [15]. The algorithm in [5] was used for RC, where initial values of 3.2001 and -1.3670 were used for α and β, respectively. For the purpose of this test, the open-source Turing codec [3], an HEVC implementation specifically designed for speed, quality, and usability, was used. In tests parallelisation at both frame and CTU level were used.

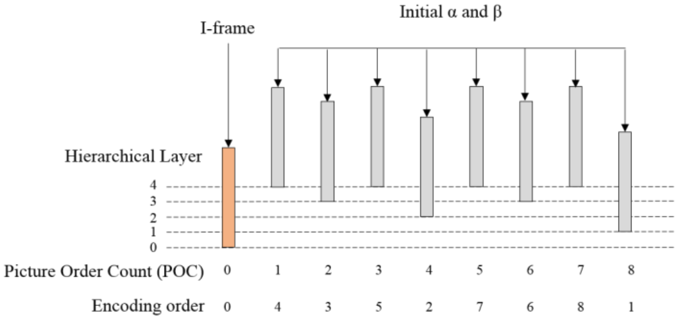

Fig. 1 SOP structure commonly used in HEVC codecs

The RC scheme was designed without consideration for parallelisation and segmentation and accurately reaches the target bitrate in that context. However the accuracy drops significantly when the encoder is constrained with parallelisation and segmentation. Table I shows some results comparing the accuracy in meeting the target bitrate of this RC scheme, by first disabling and then enabling parallelisation and segmentation. The overall difference between the target bitrate and the achieved bitrate is 5.2 % when parallelisation and segmentation are disabled, compared to 11.9 % when enabled (figures obtained by averaging the difference in percentage between the target bitrate and actual bitrate achieved by the coder, for several target bitrate points). These results show that these RC algorithms are not ideal when enabling parallelisation and segmentation. Table I also shows that the encoder time is significantly shorter when parallelisation is enabled (an average speed up of 4 times), which motivates this work.

TABLE I
DIFFERENCE IN TARGET BITRATE, AND PSNR FOR A TARGET OF 10 MBPS, PARALLELISATION AND SEGMENTATION DISABLED, AND ENABLED.

Sequences	(Parallel. + seg.) disabled			(Parallel. + seg.) enabled		
	diff [%]	PSNR [dB]	Encode time (s)	diff [%]	PSNR [dB]	Encode time (s)
Boxing	8.51	41.97	7159	25.70	40.27	1703
ParkDancers	1.57	35.50	5027	10.01	35.55	1224
Sedof	2.26	29.42	4060	13.72	28.48	1165
TapeBlackRed	9.39	44.42	7106	29.31	44.40	1685

The central parameter in RC is λ [5]. Due to the fact that β has a higher impact on the selection of λ, it has been used to further research a content-dependent initialisation for the R-λ RC model. A training set consisting of 16 sequences were encoded with a set of initial values for β, hardcoded between -0.2 and -2.0 in steps of 0.1. Four target bitrates were tested: 3, 10, 18 and 25 Mbps. For each test point (each sequence at each target bitrate), initial values of β were discarded if the sequence presented too abrupt a variation in PSNR, (over 15% difference in PSNR between two segments) or if jumps in visual quality could be observed between the segments. From the remaining initial values of β, the best value was identified, for which the target bitrate is most accurately reached.

Optimised initial values for β, shown in Table II, were observed to be dependent on the content, particularly on the texture of the sequence. *Sedof*, for instance is spatially very complex and requires a low initial β of -2 at 3 Mbps, and *TapeBlackRed* is spatially smoother and requires for the same target bitrate a much higher initial β of -0.7.

A measure of the texture complexity of the first intra-frame can be computed following the approach from [16] which uses the Sum of Absolute Transformed Differences (SATD) on 8×8 blocks of pixels. SATD is defined as the sum of the absolute values of coefficients obtained after applying the Hadamard transform to the original block of pixels. This metric was first proposed to estimate the bits to spend on intra-

TABLE II
SELECTED INITIAL BETA VALUE AND CPP

Sequence	**Selected β value**				**CPP**
	3Mbps	**10Mbps**	**18Mbps**	**25Mbps**	
Boxing	-1.1	-0.7	-0.7	-0.6	5.0
CandleSmoke	-0.7	-0.7	-0.7	-0.7	4.8
Discus	-1.4	-1.0	-1.0	-0.9	8.0
Hurdles	-1.1	-0.9	-0.7	-0.7	6.6
LongJump	-1.4	-0.9	-0.9	-0.8	9.1
Manege	-2.0	-1.2	-1.2	-1.1	16.7
Netball	-1.1	-0.8	-0.8	-0.7	4.8
NingyoPompoms	-1.1	-0.8	-0.8	-0.8	4.5
ParkAndBuildings	-1.4	-0.9	-0.9	-0.9	8.1
ParkDancers	-1.8	-1.1	-0.9	-0.9	9.7
Petitbato	-1.4	-0.9	-0.9	-0.8	9.1
Sedof	-2.0	-1.1	-1.1	-1.1	16.1
ShowDrummer	-0.8	-0.7	-0.7	-0.7	5.9
Somersault	-0.6	-0.5	-0.5	-0.5	3.4
TableCar	-1.8	-1.1	-1.0	-0.9	11.0
TapeBlackRed	-0.7	-0.4	-0.4	-0.3	2.4

predicted frames, and is therefore suitable for application in scenarios with parallelised encoding where the complexity needs to be measured on the first frame (an intra frame). For each sequence, a cost measure C is assigned to be the SATD of the first I-frame of the sequence and a Cost Per Pixel (*CPP*) value is obtained by dividing C by the number of luma samples within the frame, or:

$$CPP = C \div (W \cdot H) \tag{5}$$

where W and H are width and height of the frame, respectively. The values of *CPP* for the sequences in Table II are shown on the right-most column. It can be observed that generally, the higher the *CPP*, the lower the value of the selected initial β value. Additionally, for each sequence, it can be observed that the higher the target bitrate, the higher the initial β. For these reasons, both *CPP* and target bitrate R_T are used to determine the best initial β value. Therefore, by studying the selected values for the initial β against CPP in Table II, a relationship between β and CPP can be built, modelled by fitting a curve.

$$\beta_{init} = -P_0 \cdot \left(CPP^{P_1} - P_2\right) \tag{6}$$

where P_i (i = 0, 1, 2) can be expressed as

$$P_i = U_i \cdot R_T^{\,V_i} + K_i \tag{7}$$

and U_i, V_i and K_i (i = 0, 1, 2) are determined by fitting a curve through the selected β_{init} points, for each R_T.

By using Eq. (7) for (i = 0, 1, 2) with Eq. (6), it is possible to determine a content-dependent and target bitrate dependent initial β value, by using R_T and the *CPP* of the first I-frame of the segment. The predicted initial β using the method described is plotted in Fig 2.

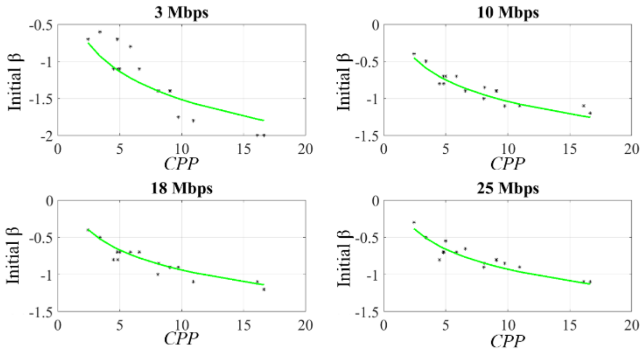

Fig. 2 Predicted initial β using proposed model (full curve) and manually selected initial β (dots), against average CPP.

IV. EXPERIMENTAL RESULTS

The proposed method was implemented in the context of the state-of-the-art algorithm in [5] within the HEVC Turing codec and compared to using fixed initial RC parameters. Full parallelisation was used to achieve the highest compression speed, and sequences were segmented at intervals of two seconds. Tests were performed using the 16 training sequences from Section III and on additional 8 test sequences in the UHD format at target bitrates of 3, 8, 10, 14, 18, 22, 25 and 30 Mbps. The tests were run on Linux machines with Intel Xeon X3450, 2.67 GHz clock frequency and 8 GB of RAM. For each test point, from both the training sequences and the new set of 8 sequences, the difference between the target and actual bitrate was calculated. Finally, the average accuracy among all tested bitrates was computed and compared with the method in [5].

On average, the proposed method achieves a difference of 6.45 % between target and actual bitrate on the training sequences, against 11.92 % when using the RC in [5]. Similar results are obtained on the test sequences. Table III presents overall accuracy for the test sequences. Clearly, accuracy improves considerably; overall a 6.73 % difference is obtained, against 12.45 % in [5]. Table IV shows a selection of results at three representative bitrates where the quality of the encoded sequences is also presented in terms of average PSNR. The table shows that the proposed approach consistently increases the accuracy of the RC algorithm across most of the sequences in the test set, with virtually no impact on the quality in terms of average PSNRs. Informal subjective viewings suggest that the visual quality is also improved particularly between two segments, whereas common RC algorithms may result in quality jumps.

TABLE III

AVERAGE DIFFERENCE BETWEEN TARGET RATE AND ACTUAL RATE

Sequence	**Difference [%], β = -1.367 [5]**	**Difference [%], adaptive initial β**
Badminton	17.13	2.07
CentralLineCrossing	23.58	16.37
MenAndPlants	14.26	4.83
Oban	11.57	5.72
Parakeet	5.36	10.29
SpinningObjects	5.29	5.59
TruckCyclist	4.43	2.27
YoungDancers1	17.99	6.71
Average	*12.45*	*6.73*

V. CONCLUSIONS

The use of fast parallelisation and segmentation present difficulties for RC algorithms to accurately reach a target bitrate. Under these constraints, we presented a content and bitrate dependent adaptive β initialisation for a R-λ model RC, using the complexity of the first I-frame of the segment.

This adaptive initial β algorithm demonstrated a consistent improvement in reaching the target bitrate for a set of training and test sequences over a range of different target bitrates, compared to using a fixed initial value for β. On average, the proposed model reaches a much better accuracy, with a difference of 6.73 % between the target bitrate and the bitrate reached by the encoder on the test sequences, compared to a difference of 12.45 % for a fixed initial value of β.

ACKNOWLEDGEMENTS

The work described in this paper has been conducted within the project COGNITUS. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 687605.

REFERENCES

[1] G. J. Sullivan, J. Ohm, W.-J. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, Dec. 2012.

[2] T. K. Tan et al., "Video Quality Evaluation Methodology and Verification Testing of HEVC Compression Performance," *IEEE Trans. Circuits Syst. Video Technol*.,, vol. 26, no. 1, Jan. 2016.

[3] S. G. Blasi, M. Naccari, R. Weerakkody, J. Funnell and M. Mrak, "The open-source Turing codec: towards fast, flexible and parallel HEVC encoding," *International Broadcasting Conference*, Sep. 2016.

[4] I. Sodagar, "The MPEG-DASH standard for multimedia streaming over the Internet," *IEEE MultiMedia*, vol. 18, no. 4, Apr. 2011.

[5] B. Li, H. Li, L. Li, and J. Zhang, "λ domain rate control algorithm for high efficiency video coding," *IEEE Trans. Image Process.*, vol. 23, no. 9, Sep. 2014.

[6] S. Ma, W. Gao, and Y. Lu, "Rate-distortion analysis for H.264/AVC video coding and its application to rate control," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, Dec. 2005.

[7] D.-K. Kwon, M.-Y. Shen, and C. Kuo, "Rate control for H.264 video with enhanced rate and distortion models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 5, May 2007.

[8] Z. He and S. K. Mitra, "Optimum bit allocation and accurate rate control for video coding via ρ-domain source modeling," IEEE Trans. Circuits Syst. Video Technol., vol. 12, no. 10, Oct. 2002.

[9] B. Li, H. Li, L. Li, and J. Zhang, "Rate control by R-lambda model for HEVC," JCTVC-K0103, *11th JCTVC Meeting*, Shanghai, CN, Oct. 2012.

[10] B. Li, D. Zhang, H. Li, and J. Xu, "QP determination by lambda value," JCTVC-I0426, *9th JCTVC Meeting*, Geneva, CH, Apr. 2012.

[11] H. Wang and S. Kwong, "Rate-distortion optimization of rate control for H.264 with adaptive initial quantization parameter determination," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 1, Jan. 2008.

[12] W. Wu, H. Kim, "A novel rate control initialization algorithm for H.264", *IEEE Trans. Consumer Electron.*, vol. 55, May 2009.

[13] I. Zupancic, M. Naccari, M. Mrak, and E. Izquierdo, "Two-pass rate control for improved quality of experience in UHDTV delivery," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no 1, Feb. 2017.

[14] L. Li, B. Li, D. Liu, and H. Li, "λ-domain rate control algorithm for HEVC scalable extension," *IEEE Trans. Multimedia*, vol. 18, no. 10, Oct. 2016.

[15] M. Naccari, R. Weerakkody, J. Funnell and M. Mrak, "Enabling Ultra High Definition Television services with the HEVC standard: The THIRA project," *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, Turin, 2015, pp. 1-4.

[16] M. Karczewicz and X. Wang, "Intra frame rate control based on SATD," JCTVC-M0257, *13th JCTVC Meeting*, Incheon, KR, Apr. 13.

TABLE IV

SELECTION OF RESULT: DIFFERENCE BETWEEN TARGET RATE AND ACTUAL RATE AND PSNR

	Target bitrate = 10 Mbps				Target bitrate = 22 Mbps				Target bitrate = 30 Mbps			
	β = -1.367 [5]		Adaptive init. β		β = -1.367 [5]		Adaptive init. β		β = -1.367 [5]		Adaptive init. β	
	diff [%]	PSNR	diff [%]	PSNR	diff [%]	PSNR	diff [%]	PSNR	diff [%]	PSNR	diff [%]	PSNR
Badminton	18.21	38.03	1.54	37.82	18.98	39.73	2.94	39.39	18.31	40.31	0.21	39.95
CentralLineCrossing	11.83	34.13	5.44	34.05	7.45	35.25	4.42	34.95	5.17	35.57	5.45	35.51
MenAndPlants	16.74	40.66	4.89	40.86	14.05	43.14	4.64	42.36	14.78	43.73	2.50	43.16
Oban	7.46	37.56	6.48	37.44	15.02	38.69	5.29	38.51	11.84	38.82	2.58	38.89
Parakeet	1.62	38.73	14.89	38.82	0.29	38.96	0.07	38.99	1.39	39.04	0.87	39.02
SpinningObjects	8.01	41.44	5.81	41.28	3.96	44.01	4.18	43.41	3.02	44.50	1.40	44.17
TruckCyclist	5.67	40.42	0.98	40.11	4.63	41.83	3.29	41.58	0.80	42.15	2.52	42.10
YoungDancers1	19.65	40.05	8.16	39.79	10.92	40.52	5.45	40.28	14.50	40.82	5.99	40.60