Direct Optimisation of λ for HDR Content Adaptive Transcoding in AV1

Vibhoothi[*a], François Pitié[a], Angeliki Katsenou[a], Daniel Joseph Ringis[a], Yeping Su[b], Neil Birkbeck[b], Jessie Lin[b], Balu Adsumilli[b], and Anil Kokaram[*a]

[a]Trinity College Dublin, Dublin, Ireland
[b]Google Inc., CA, USA

ABSTRACT

Since the adoption of VP9 by Netflix in 2016, royalty-free coding standards continued to gain prominence through the activities of the AOMedia consortium. AV1,[1] the latest open source standard, is now widely supported. In the early years after standardisation, HDR video tends to be under served in open source encoders for a variety of reasons including the relatively small amount of true HDR content being broadcast and the challenges in RD optimisation with that material. AV1 codec optimisation has been ongoing since 2020 including consideration of the computational load.[2] In this paper, we explore the idea of direct optimisation of the Lagrangian λ parameter used in the rate control of the encoders to estimate the optimal Rate-Distortion trade-off achievable for a High Dynamic Range signalled video clip. We show that by adjusting the Lagrange multiplier in the RD optimisation process on a frame-hierarchy basis, we are able to increase the Bjontegaard difference[3] rate gains by more than $3.98\times$ on average without visually affecting the quality.

Keywords: AV1, Adaptive Video Encoding, Rate-Distortion Optimisation, Lagrangian Optimisation

1. INTRODUCTION

In recent years, the growth in delivery of video at scale for broadcast and streaming applications (from Netflix, YouTube, Disney etc.) has inspired further research into content-adaptive transcoding. The goal is to deliver high-quality content at progressively lower bitrates by adapting the transcoder for each input at a fine-grained level of control. In 2013, YouTube was the first to adopt this strategy for its User-Generated-Content (UGC) by building a pipeline that is based on clip popularity by re-processing a clip with an enhanced pre-processor in combination with a different built-in transcoder. Around the same time, Netflix's seminal work on per-clip and per-shot encoding[4] for High Value Content (HVC) videos showed that an exhaustive search of the coding parameter space can lead to significant gains in Rate-Distortion (RD) tradeoffs per clip. These gains offset the high one-time computational cost of encoding as the same encoded clip may be streamed to millions of viewers across many different Content Delivery Networks (CDNs),[5] thus effectively saving bandwidth and network resources. That idea has since been revisited and has become more efficient by applying the Viterbi algorithm across shots and parameter spaces.[6] Over the past years a lot of researchers have focused on the optimisation of a high-level parameter (target bitrate or quantisation factor or objective quality) to generate an optimal bitrate ladder for a clip as part of a Adaptive Bitrate Streaming (ABR).[7–9]

In our previous work,[10] we showed that the RD tradeoff can be directly addressed by applying a numerical optimisation scheme to estimate the appropriate Lagrangian λ multiplier for a given clip (see Section 2) for standard dynamic range (SDR) videos. We observed an average BD-rate improvement of 1.9% for HEVC, 1.3% for VP9,[11] and 0.5% in AV1.[12] In our latest work,[12] we further demonstrated that additional BD-rate(%) gains, from 0.5% to 4.9%, for AV1 could be achieved by adopting a per frame-type optimisation.

In this paper, we explore the idea of λ optimisation on High-Dynamic Range (HDR)/Wide Color Gamut (WCG) material.[13] HDR/WCG systems can capture, process, and reproduce a scene conveying the full range of perceptible shadow and highlight details beyond normal dynamic range (SDR) video systems.

---

* vibhoothi@tcd.ie, anil.kokaram@tcd.ie

Similarly to our latest work,[12] we propose a content-adaptive transcoder optimisation at a global and a deeper frame-type level. The core new ideas are i) consideration of RD Optimisation (RDO) for HDR content, ii) optimisation of the RDO parameters on a frame-hierarchy basis, and iii) investigation of various convergence criteria that result in the minimisation of the computational load. Our experiments in Section 5 demonstrate that a frame-based tuning of video encoder can lead to an average gain of 1.63% of BD-rate (best recorded gain of 9.3%) compared to the standardised method. Moreover, the average gains per shot range between 0.58 and 3.43% for HDR video content in AV1.

Section 2 gives an overview of previous research work and λ definition in rate control. Section 3 explains the proposed methodology as well as the multi-dimensional optimisation. Section 4 then details the experimental set-up including the test sequences, the keyframe combination selection, the framework implementation. Section 5 reports on the experimental findings.

2. BACKGROUND

The work of Sullivan et al[14] laid the foundations for optimising the RD tradeoff in modern video codecs. By taking a Lagrange multiplier approach the joint optimisation problem is posed as the minimisation of $J = D + \lambda R$, where λ is the Lagrange multiplier controlling the tradeoff. This idea is the basis of the RDO process used especially in making mode decisions in modern codecs. The independent variable in this optimisation is usually qp, a quantiser step size. Increasing qp reduces rate R but increases distortion D. Also, a different choice of λ yields different R, D pairs.

Different codecs devised different recipes to derive the optimal λ value for RDO through an empirical relationship with qp. In libaom-AV1[1] λ is empirically related to q_i (Quantizer Index: $\approx qp * 4$ in the AV1 codebase), as follows:

$$\lambda = q_{dc}^2 \times (A + 0.0035 \times q_i), \tag{1}$$

where A is a constant depending on the frame type ($3.2 \leq A \leq 3.3$) and $q_{dc} = f(q_i, A)$ is defined through a discrete valued Look Up Table (LUT) ($0 \leq q_i \leq 255$ for AV1). This $\lambda - qp$ relationship is not necessarily optimal for a particular clip because the empirical relationship was derived for optimality over an entire test corpus. To maximise gains, λ should be content dependent.

Per Clip λ Optimisation. The idea of adapting λ based on video content is not entirely new. Zhang and Bull[15] altered λ based on distortion statistics on a frame-basis for HEVC. In our previous work,[10, 11, 16] we introduced the idea of an adaptive λ on a clip basis, using a single modified $\lambda = k\lambda_o$ across all the frames in a clip. Here, λ_o represents the default value deployed from the relevant empirical relationship e.g. Eq. (1). In order to find the optimal λ value, we deployed numerical optimisers (Brent's method and Golden-search[17]) that minimised the BD-rate as a cost function. Later work,[10] considered the use of Machine Learning techniques to reduce the required computational load.

Per Clip, Per Frame-Type λ Optimisation. In our latest work,[12] we showed that this method of global λ tuning yields average BD-rate(%) gains of only 0.539% and 0.097% for AV1 and HEVC, respectively. These modest improvements are probably due to the fact that current modern video encoders are content-adaptive by nature and include many new improvements such as partition tools, Inter/Intra prediction tools and modern hierarchical reference frame-structure. Another important aspect to consider is that the heuristics and empirical shortcuts used in this content-adaptive implementation of the Lagrangian parameter, deviates from classical RDO theory, which normally requires λ to be constant across the sequences over which distortion is measured. For example in Eq.(1) λ changes with frame type. Motivated by this observation, in our latest work,[12] we studied the effect of isolating the optimisation of λ for different frame types. Results on SDR sequences showed that optimising λ purely for Keyframes (KF), Golden-Frames (GF), Alternate-reference Frames (ARF), leads to average BD-rate gains of 4.92% compared to global λ optimisation (only 0.54% BR-Rate gains).

HDR in AV1. The quality of compressed 4K and 8K HDR content was evaluated by Pourazad et. al.,[18] drone content from P. Topiwala et.al.,[19] and gaming (Nabajeet et. al.[20]). More relevant here is the work of Zhou et. al[21] for HEVC that expresses distortion D in terms of the HDR-VDP-2 quality metric.[22] They presented an algorithm for prediction of λ at the CTU level which resulted in 5% BD-rate improvement w.r.t. a reference implementation of HEVC (HM16.19).

Algorithm 1 Proposed Optimisation Workflow
_____
 1: Initialise video set, qp points, codec and associated options.
 2: [Run-A] Encode each clip in parallel with P threads, where P is a number of qp points.
 3: Compute RD-points and collect metrics using `libvmaf`.
 4: This establishes the RD curve corresponding to λ_o
 5: [Run-B] Deploy selected Numeric Optimiser with initial condition $\mathbf{k} = 1$
 6: On each successive iteration i of the Optimiser Do
 7:    Encode the given clip using the new value of $\mathbf{k} = \mathbf{k}_i$
 8:    Compute the cost function $C_m(\mathbf{k}_i)$ using reference RD curve for BDRATE calcuation from Run-A
 9: Terminate optimiser using default convergence and stopping criteria.
_____

3. DIRECT λ OPTIMISATION IN AV1

As noted in Section 2, the encoder is determining λ independently on a frame basis inside the encoder. In this work, we explore the impact of treating λ optimisation as a multi variable search problem at a frame basis w.r.t. HDR content. The following sections expand our main strategies for the direct optimisation of the λ parameter.

BD-rate Optimisation. For the nth RDO decision in clip m, we propose that $\lambda_n = k_n\lambda_0$. We estimate $\mathbf{k} = [k_1, \ldots, k_N]$ (where we assume N RDO decisions) to maximise the BD-rate gain using MS-SSIM[23] as the quality metric (Q_m). The cost function $C_m(\mathbf{k})$ can therefore be formulated as:

$$C_m(\mathbf{k}) = \text{BD-rate}(R_m(\mathbf{k}\lambda_o, Q), R_m(\lambda_o, Q))$$
$$\propto \int_{Q_1}^{Q_2} \left(R_m(\mathbf{k}\lambda_o, Q) - R_m(\lambda_o, Q)\right) dQ \,, \tag{2}$$

where $R_m(\mathbf{k}\lambda_o, Q)$ is the bitrate of the mth clip at quality Q, using $\lambda = \mathbf{k}\lambda_o$ for the N RDO decisions and Q_1, Q_2 defined as usual.[3] $R_m(\cdot, \cdot)$ is derived from the MS-SSIM-based RD curve generated using P qp measurements. Here we use $P = 5$: $qp =\{27, 39, 49, 59, 63\}$.

The flow of the optimisation framework is reported in Algorithm 1. We can see that repeated computations of the BD-rate are required, and this incurs a huge computational cost. To address this, we deploy the idea of proxies for parameter selection as proposed by Ping H. et. al.[24,25] They observe that using different speed settings of the encoder at the same target quality/quantizer level results primarily in bitrate differences that are directly proportional to the content complexity (see section 5.1). That idea also extends to the use of lower resolution proxies. Therefore we can reduce computational load by performing optimisation using faster encoder presets 'em and lower resolution proxies.

Multi-Dimensional Optimiser. Our previous study for finding the optimal λ multiplier in AV1 used Brent's line search method.[12] The focus was on applying a modifier k to only one sub-module in the encoder. However, here we explore the scenario of a multiple dimensional search for finding the optimal λ for multiple frame types each associated with a different k. For this multi-dimensional search, we have various options like Nelder-Mead Simplex,[26] Conjugate Gradient,[27] Powell's method,[28] etc.

In order to select a suitable optimiser, we conducted a simple analysis by carrying out an exhaustive grid-search on a single video to study the surface of our objective function. For this study, we chose the clip NocturneRoom from the AOM Common-Testing-Configuration (AOM-CTC)[29] set, and optimised λ for two different frametypes: λ_{KF} for the KFs and $\lambda_{GF/ARF}$, for the GF/ARFs. For all other frame-types, λ was set to the default. The grid search range for both λ_{KF} and $\lambda_{GF/ARF}$ is 0.6 to 5.4, in steps of 0.1. This provided 2,401 anchor points resulting in a total of 12,005 RD points for analysis.

Figure 1 shows the contour plot of the BD-rate % (MS-SSIM) objective function for a sample clip. The surface is clearly smooth, and a gradient based method is expected to converge to a sub-optimal solution (local minimum) due to very low gradient. This observation was confirmed by testing gradient based methods such as the Nelder-Mead Simplex method[26] and the Conjugate Gradient.[27] Both of these methods converged erroneously after the first iteration as the gradient was very close to zero. Therefore, we explored line-search constrained
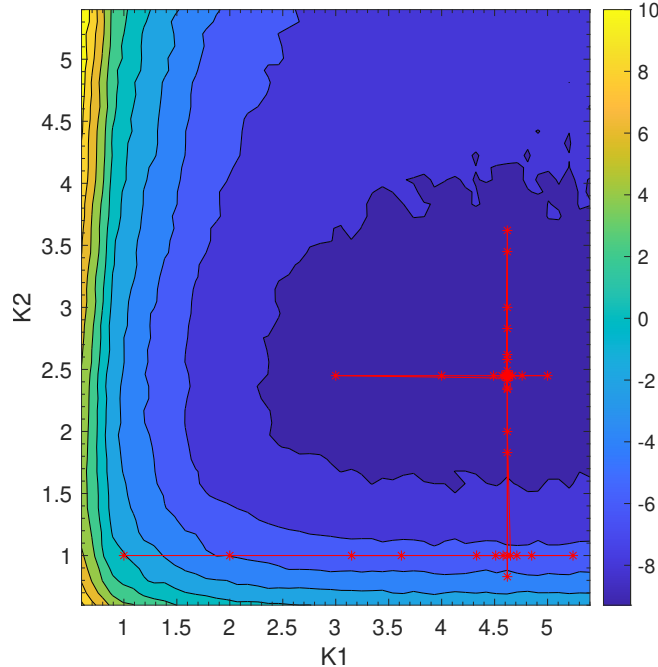
Figure 1: Plot of the optimisation surface, where X axis represents the λ_{KF}[K1] multiplier value and Y axis the $\lambda_{GF/ARF}$[K2] value, contour is $C_m(k_1, \ldots, k_n)$. The red line shows the variation of K1 and K2 values when Powell's method[28] is deployed for direct optimisation.

methods. One of the best performing was the modified Powell method,[28] which succeeded to reach the global minima for the test clip (see red-lines on the Figure 1).

4. EXPERIMENTAL SETUP

4.1 4K HDR Corpus

For our experimental studies, we formed a video corpus consisting of 50 video clips (6500 frames) curated from various public sources. All the videos are normalized to BT.2020 color primaries with SMPTE2084 Perceptual Quantizer (PQ) transfer function and represented in the YUV colourspace inside YUV-Y4M containers. All the conversions and normalization of the clips are implemented with HDRTools.[30] The configuration file for the conversion to YUV Space with PQ Signal is available in our project page*. The 50 clips contain 130 frames, a resolution of 3840x2160/4096x2160, and can be further grouped into 7 shot groups. Figure 2 illustrates sample frames of the dataset and Table 1 gives a short description of the content of these 7 shot groups. More information on the dataset, including computed Spatial and Temporal Information (SI and TI)[31] and Dynamic Range (DR) can be found in our project page*.

4.2 Keyframe Selection

Reference frames (henceforth called keyframes) in AV1, typically contain 5 to 10 times more bits than other frames. Therefore, we target the optimisation of the bit allocation of these keyframes. In AV1, in order to code an Inter frame, references up to 8 keyframes[35] are used. The encoder chooses from multiple frames in both forward and backward direction.[36] For the simulations presented, we consider multiple combinations of the 3 keyframe types: the reference Intra-coded frame *KF*, an *ARF_FRAME (ARF)* used in prediction but does not appear in the display, and an Inter coded frame which is coded at higher quality *GOLDEN_FRAME (GF)*.

---

*Project page: https://gitlab.com/mindfreeze/spie2022

Figure 2: Sample frames from the corpus.

Table 1: High-level description of the shots.

Shot Group	Description
Cosmos[32]	Vibrant animated sequence, high temporal complexity, 24fps
Meridian[29,32]	Natural sequence, high spatial complexity, 59.94fps
Nocturne[29,32]	Natural sequence, medium spatial complexity, 60fps
Sol Levante[29,32]	Animated sequence, medium-high temporal complexity, 24fps
Sparks[29,32]	Sequence with medium motion and wide dynamic range, 59.94fps
SVT[33]	Very high natural complexity sequences, 50fps
Cables 4K[34]	Outdoor sequences with moderate complexity, 59.94fps

4.3 Framework Implementation

For the simulations, Random-Access (RA) encoding mode was chosen as per AOM-CTC.[29] This mode is commonly used for streaming as it allows users to randomly seek into any frame of the clip. We deployed a stable release for AV1 (`libaom-av1-3.2.0, 287164d`) with modifications to allow k to propagate to the desired mode from a command-line argument.

The objective metrics for Quality and Rate (RD measurements) at the selected qp settings were computed using libvmaf,[37] a standard open-source video quality evaluation library. Our software framework for performing these experiments with AV1 is based on AreWeCompressedYet.[38]

5. EXPERIMENTS & RESULTS

5.1 Proxy Processing

As discussed earlier, optimisation requires many encodes, and we need to consider faster proxy presets to make these experiments practical. We have investigated three proxy modes for use during the optimisation: (4K S2-S2), the default non-proxy encode at the original 4K resolution using AV1 Speed 2 preset as in the final setting, (4K S2-S6), which encodes videos at 4K resolution using AV1 Speed 6 preset, and (1080p S2-S6), which operates at a (Lanczos 5) downsampled 1080p resolution with speed preset 6.

Table 2 presents the BD-rate gains on the 4 sequences of the *av2-g1-hdr-4k* set from AOM-CTC. The optimisation is performed in this case for a single global k using Brent's method.[17] It is clear that using the proxy method (1080p S2-S6) reduces the encoding complexity by an average of 4.8×, with negligible degradation in quality (BD-rate) when compared against full-resolution optimisation mode (4K S2-S2). We also note that BD-rate gains at (4K S2-S6) are almost identical to (1080p S2-S6), but with about 30% slower encoding speed. Given that processing time for our subsequent experiments takes in the order of 100's of hours, we therefore use this (1080p S2-S6) proxy setting in the rest of the study. Hence in results reported in Table 3 we use that proxy to estimate optimal values of $k_1, \ldots, k_n$; then evaluate BD-rate gains using the original material i.e. 4K with S2 preset. We observe of course some differences between this approach and using 4K S2 throughout but it is the more pragmatic approach validated by our results in Table 2.

5.2 Multiple Frame Types λ Optimisation Performance

One of the objectives of this work is to compare different combinations of frame-type dependent λ optimisations with the ultimate aim to find the best λ optimisation strategy. First, we considered 4 modes, as in our previous work.[12] Particularly, we set $\lambda = k\lambda_o$ for some grouped frame types, and kept $\lambda = \lambda_o$ for all other frame types. These four modes are: i) *All Frames*, which refers to the global λ tuning method where we set the same

Table 2: Proxy encoding time(hours), estimated λ multiplier value, and BD-rate gains (%) MS-SSIM for the different proxy encoding speeds and video resolutions.

Proxy Settings	Encoding Time (hrs)			λ Multiplier			BD-rates (%)		
	4K S2-S2	4K S2-S6	1080p S2-S6	4K S2-S2	4K S2-S6	1080p S2-S6	4K S2-S2	4K S2-S6	1080p S2-S6
MeridianRoad	332.78	107.05	102.71	1.02	1.02	1.16	0.07	0.07	-0.2
NocturneDance	598.07	189.32	134.46	0.75	1.01	1.01	0.45	-0.08	-0.08
NocturneRoom	932.528	252.87	159.29	3.79	3.86	2.81	-9.19	-9.24	-7.86
SparksWelding	1065.35	249.94	179.91	1.59	2.25	1.56	-1.47	-1.81	-1.39

multiplier k for all frames, as previously proposed by Ringis et al.,[11] ii) *KF*, which refers to the scenario where we set $\lambda = k\lambda_o$ tune for KF, iii) *GF-ARF*, which refers to λ optimisation for GF and ARF frames, iv) *KF-GF-ARF*, which refers to optimising λ for KF, GF, and ARF frames.

In addition to the above four modes, we introduced a new search method *Powell [KF-GF/ARF]*, which is a multidimensional joint search, where we deploy two k values, with k_1 for *KF* and k_2 for *GF* and *ARF* frames. Thus, $\lambda_{KF} = k_1\lambda_o$ and $\lambda_{GF/ARF} = k_2\lambda_o$, and for others we use the default λ value.

Table 3 reports the BD-rate (MS-SSIM %), MS-SSIM (dB), VMAF[39] gains for these different optimisation strategies for the whole corpus which is divided into 7 different shots groups (see Table 1). The underlined result shows the highest gains in terms of BD-rate that the proposed tuning brings compared to (4K S2). The results presented are averaged on a shot group basis. Also, the minimum and average BD-rate in the shot group are recorded. Another perspective of the resulting BD-rates from all tested clips is illustrated in the histograms of the BD-rate values in Figure 3.

First, we observe that the new values of k are, on average, significantly different from $k = 1$, hence verifying our initial hypothesis of a "better" λ value. Inspecting the Brent optimiser results from both Table 3 and Figure 3, it transpires that the *KF-GF-ARF* method is achieving the best BD-rate gains on average (1.12%). Analysing the results on per-shot basis, we were able to see significant improvements in BD-rates for certain shots. Best average gains recorded were for Sol-Levante, where the BD-rate improved from 0.79% to 2.94%, and this was followed by Cosmos with an improvement from 1.44% to 2.74%.

Another important observation is that the multidimensional search method *Powell [KF, GF/ARF]*, appears to be the overall top performer, with average BD-rate gains of 1.63%. The histogram in Figure 3e also shows that the improvement is consistent for the majority of the videos. This also evident in terms of bitrate, as we achieve a bitrate reduction of 0.64% with *KF, GF, ARF* and 4.25% with *Powell [KF, GF/ARF]* at QP39. We also observed that for this improved bitrate savings, we have a minimal loss of MS-SSIM with 0.06dB and 0.35 for VMAF on average.

From Table 3, we notice that clips within the same shot group respond differently. These deviations can be partly explained by variations in spatial and temporal information (See project page*). Analysing the distribution of results obtained with the Powell method, we observed that the higher BD-rate gains (> 4%) were acquired for clips with high spatial information/complexity (5 clips). Videos with lower temporal information, exhibited higher gains. Videos with low temporal and low spatial complexity had no improvements. Different optimisation modes exhibited very different distributions, for instance, in the classic tuning method, the BD-rate improvements were noticed with clips with SI between 400-500 while for tuning mode of λ for KF/GF gave large loss for the same range of videos.

5.3 Convergence Speed of Optimisation Methods

The Powell method is computationally expensive, with a roughly twice computational cost compared to the Brent optimisation. Practically this means 203 hours on average over 118 hours per clip. Furthermore, the Brent search for optimal $k1$ and $k2$ determinations required an average of 15 hours, while Powell method's joint search for $(k1, k2)$ needed around 87 hours. The final optimisation step, i.e. the non-proxy encode at slower speed preset, took around 107 hours per clip on average for all three modes. It is interesting that for the Powell method and for most of the clips, we were able to achieve results very close to the final iteration by only using

Table 3: Per-frame-type λ-optimisation results for multiple combination of frame types. k values are obtained using (1080p S6) proxy settings. BD-rates (BDR) are calculated using (4K S2) as an anchor.

Shot Group	Frame Type	Avg. k value	Avg. BDR (%)	Max. BDR (%)	Min. BDR (%)	Avg. Iters	Avg. Bitrate. Savings (%)	Avg. Q39 Bitrate Savings (%)	Avg. MS-SSIM Change (dB)	Avg. VMAF Change
Cosmos	*All Frames*	*1.49*	*-1.44*	*-2.91*	*-0.41*	*9.00*	*-15.71*	*-14.19*	*0.32*	*1.52*
Meridian	All Frames	1.07	0.12	-0.29	0.36	9.71	-5.32	-3.45	0.04	0.24
Nocturne	All Frames	1.65	-0.16	-1.33	3.53	8.63	-26.33	-25.43	0.49	2.87
Sol Levante	All Frames	1.23	-0.79	-2.20	0.33	9.20	-7.47	-7.07	0.22	1.48
Sparks	All Frames	1.26	-0.37	-1.31	1.01	8.22	-11.33	-10.33	0.24	1.11
SVT	All Frames	0.97	0.04	-0.36	1.04	8.83	3.12	3.04	-0.14	-0.24
Cables 4K	All Frames	1.56	-0.36	-1.05	0.16	10.00	-16.69	-15.81	0.44	1.74
Cosmos	KF	3.31	-0.85	-1.88	0.10	12.29	-2.10	-2.21	0.03	0.14
Meridian	KF	4.00	-0.37	-10.45	7.86	11.00	-2.03	1.41	0.04	0.19
Nocturne	*KF*	*5.08*	*-2.39*	*-6.98*	*-0.86*	*11.88*	*-4.28*	*-6.89*	*0.07*	*0.37*
Sol Levante	KF	4.49	-1.02	-2.43	0.00	11.20	-2.58	-2.39	0.08	0.33
Sparks	KF	2.54	-0.44	-2.75	2.21	11.33	-0.44	-0.78	-0.01	-0.03
SVT	KF	6.20	0.36	-0.59	3.01	10.50	-1.83	-1.74	0.13	0.38
Cables 4K	KF	2.49	0.16	-2.65	6.19	11.00	1.13	0.56	-0.04	-0.06
Cosmos	GF, ARF	2.00	-1.63	-3.88	0.00	8.00	-2.69	-2.53	0.04	0.17
Meridian	GF, ARF	2.07	1.01	-0.19	3.50	9.14	-6.26	-7.15	0.12	0.78
Nocturne	GF, ARF	1.51	0.13	-3.32	4.23	10.75	5.24	0.54	0.06	0.33
Sol Levante	*GF, ARF*	*2.71*	*-2.15*	*-3.42*	*-0.16*	*10.40*	*-4.87*	*-4.85*	*0.14*	*0.72*
Sparks	GF, ARF	1.60	0.34	-2.09	2.15	8.78	-3.54	-4.36	0.13	0.66
SVT	GF, ARF	2.38	-1.35	-4.77	0.16	11.50	-2.09	-2.10	0.16	0.78
Cables 4K	GF, ARF	1.66	0.40	-0.59	2.23	9.50	-5.21	-6.15	0.20	0.63
Cosmos	KF, GF, ARF	3.36	-2.74	-6.89	0.55	10.86	-5.58	-5.66	0.10	0.43
Meridian	KF, GF, ARF	0.96	1.37	-0.53	6.92	10.86	1.01	4.41	0.02	0.02
Nocturne	KF, GF, ARF	1.56	-2.05	-7.86	1.43	9.25	-3.00	-5.19	0.05	0.38
Sol Levante	*KF, GF, ARF*	*3.54*	*-2.94*	*-5.54*	*-0.25*	*11.20*	*-7.99*	*-7.48*	*0.24*	*1.27*
Sparks	KF, GF, ARF	0.95	-0.42	-1.39	0.14	9.56	1.82	1.85	-0.07	-0.20
SVT	KF, GF, ARF	0.91	-0.97	-3.81	0.12	10.67	4.81	4.55	-0.27	-0.45
Cables 4K	KF, GF, ARF	0.93	-0.09	-0.81	0.37	9.75	1.56	1.49	-0.06	-0.17
Cosmos	Powell (KF, GF/ARF)	(3.99, 4.09)	-3.02	-6.91	0.24	61.71	-6.36	-6.57	0.13	0.52
Meridian	Powell (KF, GF/ARF)	(3.80, 1.12)	-1.13	-9.30	3.92	40.71	-4.42	-5.45	0.03	0.12
Nocturne	Powell (KF, GF/ARF)	(4.18, 1.27)	-2.80	-7.84	0.00	52.63	-3.64	-5.73	0.07	0.43
Sol Levante	*Powell [KF, GF/ARF]*	*(4.01, 3.61)*	*-3.43*	*-5.43*	*-2.28*	*100.40*	*-8.60*	*-8.22*	*0.25*	*1.32*
Sparks	Powell (KF, GF/ARF)	(0.88, 1.58)	-0.33	-2.17	1.68	60.33	-0.15	-0.82	-0.01	0.11
SVT	Powell (KF, GF/ARF)	(6.42, 0.91)	-0.76	-3.62	0.01	47.00	0.88	0.83	-0.09	0.03
Cables 4K	Powell (KF, GF/ARF)	(2.00, 1.78)	-0.58	-2.89	1.06	50.13	-4.14	-5.37	0.08	0.30
	All Frames	1.34	-0.41	-2.91	3.53	9.06	-12.24	-11.27	0.25	1.30
	KF	3.88	-0.66	-10.45	7.86	11.34	-1.64	-1.71	0.04	0.17
Average	GF, ARF	1.92	-0.32	-4.77	4.23	9.64	-2.62	-3.78	0.12	0.57
	KF, GF, ARF	1.64	-1.02	-7.86	6.92	10.20	-0.76	-0.64	-0.01	0.13
	Powell (KF, GF/ARF)	*(3.27, 1.98)*	*-1.63*	*-9.30*	*3.92*	*57.32*	*-3.46*	*-4.25*	*0.06*	*0.35*

one iteration of the algorithm. With respect to the BD-RATE at iteration 1, the BD-rate at the final iteration (Powell method) was only on average 9% different with a median of 2.4%. Hence in practice, just one iteration of Powell is good enough and certainly for at least 50% of the clips. Taking all the above into consideration, we can reduce the total computational cost by half just by reducing the iterations of Powell search.

5.4 HDR vs. SDR λ Optimisation

It is educational to consider whether there is any difference in gains between equivalent HDR and SDR material. We therefore curated a subset of 39 clips from the current dataset, for which a SDR version of the sequence was released along with the HDR data from the content producer. Table 4 presents these SDR/HDR results for these particular subset sequences. The anchor for the BD-rate computations is the (4K S2) case. Overall, the average gains for HDR and SDR are comparable, with average BD-rate gains of 2.47% for SDR vs. 1.89% for HDR. Comparing directly the distributions of BD-rate between SDR and HDR, we notice that for 82% of the clips we have very similar BD-rate gains ($\pm 1\%$).

We need to note that this comparison between HDR and SDR must however be nuanced by the fact that the same distortion metric (MS-SSIM) is applied in both HDR and SDR. Using the same distortion metric for both allows us to make a direct comparison, but MS-SSIM has not been designed for HDR. To the best of our knowledge, there is no objective quality metric available at the moment that could facilitate a direct comparison

(a) All Frames (b) KF (c) GF, ARF
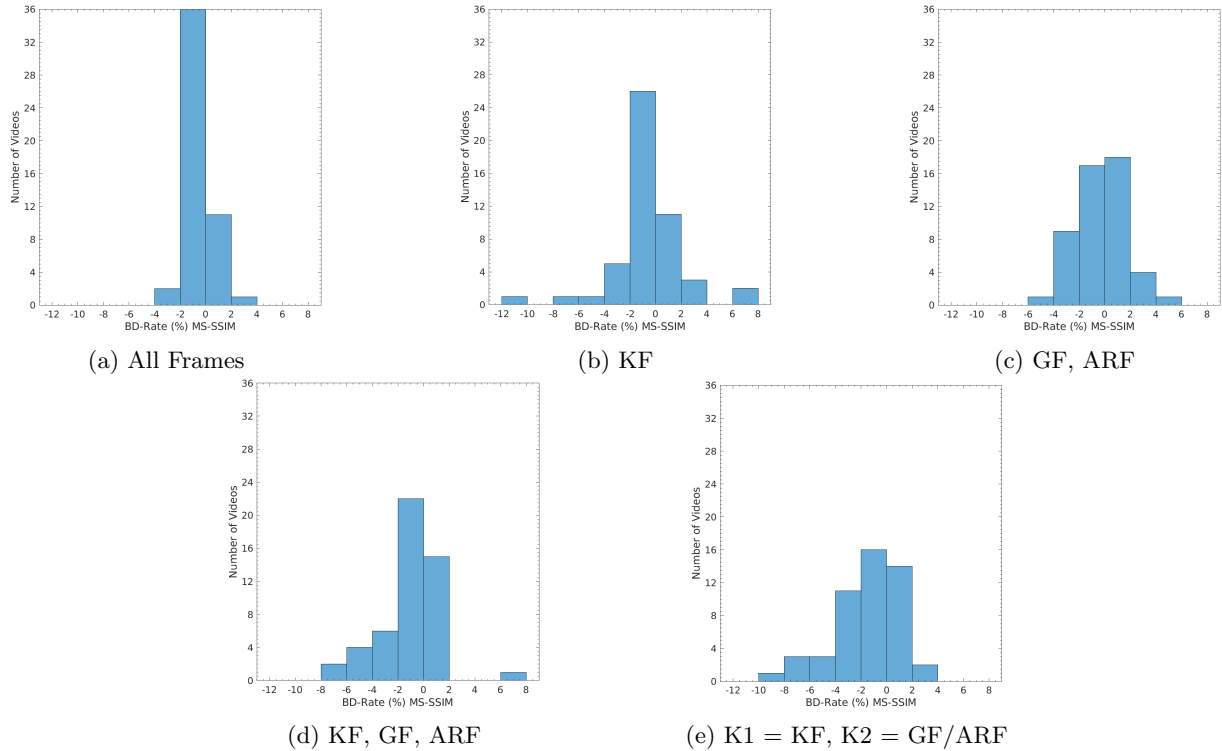
(d) KF, GF, ARF (e) K1 = KF, K2 = GF/ARF

Figure 3: Histogram distribution of BD-rate(%) MS-SSIM for various frame-level tuning methods.

of SDR and HDR. Nevertheless the point is here that HDR optimisation and SDR optimisation is quite different. In particular Table 4 shows that the average estimated value of λ is quite different for SDR and HDR.

Table 4: λ optimisation result for same set sequences represented in SDR and HDR domain, where the k values are obtained using (1080p S6) proxy settings. BD-rates (BDR) are calculated using (4K S2).

Dynamic Range	Shot Group	Avg. λ value	Avg. BDR (%)	Max. BDR (%)	Min. BDR (%)	Avg. Iters	Avg. Bitrate Savings (%)	Avg. Q39 Bitrate Savings (%)	Avg. MS-SSIM Change (dB)	Avg. VMAF Change
SDR	*Cosmos*	*(3.42, 4.41)*	*-3.26*	*-9.34*	*0.09*	*55.71*	*-5.98*	*-6.72*	*0.09*	*0.38*
	Meridian	(3.79, 0.89)	-2.78	-9.9	4.18	59.33	-3.32	-5.52	-0.01	-0.13
	Nocturne	(4.11, 1.42)	-2.86	-7.06	0.94	87.71	-6.07	-10.68	0.09	0.38
	Sol Levante	(5.36, 3.16)	-3.14	-5.55	-1.47	71	-8.44	-8.53	0.24	1.11
	Sparks	(0.82, 1.14)	-1.12	-2.48	0.37	55	1.57	1.73	-0.08	-0.24
	SVT	(2.96, 1.60)	-2	-10.69	0.14	53.17	-1.41	-1.42	0	0.33
	Average	(3.24, 2.07)	-2.47	-10.69	4.18	63.44	-3.65	-4.93	0.04	0.26
HDR	Cosmos	(3.99, 4.09)	-3.02	-6.91	0.24	61.71	-6.36	-6.57	0.13	0.52
	Meridian	(4.26, 1.15)	-1.32	-9.3	3.92	43.33	-5.15	-6.35	0.04	0.13
	Nocturne	(3.60, 1.31)	-2.95	-7.84	0	56.71	-3.66	-6.08	0.06	0.39
	Sol Levante	*(4.01, 3.61)*	*-3.43*	*-5.43*	*-2.28*	*100.4*	*-8.6*	*-8.22*	*0.25*	*1.32*
	Sparks	(0.91, 1.61)	-0.28	-2.17	1.68	59.5	0.11	-0.45	-0.02	0.07
	SVT	(6.42, 0.90)	-0.76	-3.62	0.01	47	0.88	0.83	-0.09	0.03
	Average	(3.70, 2.08)	-1.89	-9.3	3.92	60.23	-3.53	-4.27	0.06	0.37

6. CONCLUSION

We have presented a new method of per-clip optimisation based on the rate control λ multiplier in AV1 for different frame types. The proposed method was tested on a 4K-HDR corpus of 50 videos. We reported improved average BD-rate gains of 1.6% for the proposed per-frame-type per-clip optimisation compared to the

0.4% for the global per-clip λ-optimisation. The proposed method showed improvements up to 3.4% on average for certain shots compared to the method of global λ tuning. The best improvements in BD-rate range from 2.1% to 9.3% with our proposed method. We have also showed that the computational complexity of the optimisation process can be mitigated by employing proxy settings and restricting the number of iterations used by the Powell optimiser. Thus, our proposed method of multivariable Powell-based optimiser gave the best improvements on average. Future work will focus on exploring the current implementation of deriving λ from the quantiser along with more in-depth analysis of the quality aspect of the results including a subjective study.

ACKNOWLEDGMENTS

REFERENCES

[1] Chen, Y., Murherjee, D., Han, J., Grange, A., Xu, Y., Liu, Z., Parker, S., Chen, C., Su, H., Joshi, U., Chiang, C.-H., Wang, Y., Wilkins, P., Bankoski, J., Trudeau, L., Egge, N., Valin, J.-M., Davies, T., Midtskogen, S., Norkin, A., and de Rivaz, P., "An overview of core coding tools in the av1 video codec," in [*Picture Coding Symposium (PCS)*], 41–45 (2018).

[2] Wu, P.-H., Katsavounidis, I., Lei, Z., Ronca, D., Tmar, H., Abdelkafi, O., Cheung, C., Amara, F. B., and Kossentini, F., "Towards much better svt-av1 quality-cycles tradeoffs for vod applications," in [*Applications of Digital Image Processing XLIV*], **11842**, 236–256, SPIE (2021).

[3] Bjontegaard, G., "Calculation of average PSNR differences between RD curves; VCEG-M33," in [*ITU-T SG16/Q6*], (2001).

[4] Aaron, A., Li, Z., Manohara, M., De Cock, J., and Ronca, D., "Netflix Technology Blog - Per-Title Encode Optimization," (2019). https://medium.com/netflix-techblog/per-title-encode-optimization-7e99442b62a2.

[5] Conklin, G. J., Greenbaum, G. S., Lillevold, K. O., Lippman, A. F., and Reznik, Y. A., "Video coding for streaming media delivery on the internet," *IEEE Trans. on Circuits and Systems for Video Technology* (2001).

[6] Katsavounidis, I. and Guo, L., "Video codec comparison using the dynamic optimizer framework," in [*Applications of Digital Image Processing XLI*], **10752**, SPIE (2018).

[7] Reznik, Y. A., Lillevold, K. O., Jagannath, A., Greer, J., and Corley, J., "Optimal design of encoding profiles for abr streaming," in [*Proceedings of the 23rd Packet Video Workshop*], 43–47 (2018).

[8] Bentaleb, A., Taani, B., Begen, A. C., Timmerer, C., and Zimmermann, R., "A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP," *IEEE Communications Surveys Tutorials* **21**(1) (2019).

[9] Katsenou, A. V., Sole, J., and Bull, D. R., "Efficient Bitrate Ladder Construction for Content-Optimized Adaptive Video Streaming," *IEEE Open Journal of Signal Processing* **2**, 496–511 (2021).

[10] Ringis, D. J., Pitié, F., and Kokaram, A., "Near optimal per-clip lagrangian multiplier prediction in hevc," in [*2021 Picture Coding Symposium (PCS)*], 1–5 (2021).

[11] Ringis, D. J., Pitié, F., and Kokaram, A., "Per-clip adaptive lagrangian multiplier optimisation with low-resolution proxies," in [*Applications of Digital Image Processing XLIII*], **11510**, 115100E, International Society for Optics and Photonics (2020).

[12] Vibhoothi, Pitié, F., and Kokaram, A., "Frame-type Sensitive RDO Control for Content-Adaptive-encoding," *ArxiV Preprint [Online]* **2206.11976** (2022).

[13] ITU-R, R., "BT2100-2: image parameter values for high dynamic range television for use in production and international programme exchange," (2018).

[14] Sullivan, G. J. and Wiegand, T., "Rate-distortion optimization for video compression," *IEEE signal processing magazine* **15**(6), 74–90 (1998).

[15] Zhang, F. and Bull, D. R., "Rate-distortion optimization using adaptive lagrange multipliers," *IEEE Trans. on Circuits and Systems for Video Technology* **29**(10), 3121–3131 (2019).

[16] Ringis, D. J., Pitie, F., and Kokaram, A., "Per clip lagrangian multiplier optimisation for (HEVC)," *Electronic Imaging* **2020**(12) (2020).

[17] Flannery, B. P., Press, W. H., Teukolsky, S. A., and Vetterling, W., "Numerical recipes in c," *Press Syndicate of the University of Cambridge, New York* **24**, 78 (1992).

[18] Pourazad, M. T., Sung, T., Hu, H., Wang, S., Tohidypour, H. R., Wang, Y., Nasiopoulos, P., and Leung, V. C., "Comparison of Emerging Video Compression Schemes for Efficient Transmission of 4K and 8K HDR Video," in [*IEEE International Mediterranean Conference on Communications and Networking*], (2021).

[19] Topiwala, P. and Dai, W., "HDR video coding for aerial videos with VVC and AV1," in [*Applications of Digital Image Processing XLIV*], **11842**, 118420J, International Society for Optics and Photonics (2021).

[20] Barman, N. and Martini, M. G., "User generated hdr gaming video streaming: dataset, codec comparison and challenges," *IEEE Trans. on Circuits and Systems for Video Technology* **abs/2103.02189** (2021).

[21] Zhou, M., Wei, X., Wang, S., Kwong, S., Fong, C.-K., Wong, P. H. W., and Yuen, W. Y. F., "Global Rate-Distortion Optimization-Based Rate Control for HEVC HDR Coding," *IEEE Trans. on Circuits and Systems for Video Technology* **30**(12), 4648–4662 (2020).

[22] Mantiuk, R., Kim, K. J., Rempel, A. G., and Heidrich, W., "HDR-VDP-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions," *ACM Trans. on Graphics* , ACM New York, NY, USA (2011).

[23] Wang, Z., Simoncelli, E., and Bovik, A., "Multiscale structural similarity for image quality assessment," in [*37th Asilomar Conference on Signals, Systems Computers, 2003*], **2**, 1398–1402 Vol.2 (2003).

[24] Wu, P.-H., Kondratenko, V., and Katsavounidis, I., "Fast encoding parameter selection for convex hull video encoding," in [*Applications of Digital Image Processing XLIII*], **11510**, 181–194, SPIE (2020).

[25] Wu, P.-H., Kondratenko, V., Chaudhari, G., and Katsavounidis, I., "Encoding parameters prediction for convex hull video encoding," in [*2021 Picture Coding Symposium (PCS)*], 1–5, IEEE (2021).

[26] Gao, F. and Han, L., "Implementing the nelder-mead simplex algorithm with adaptive parameters," *Computational Optimization and Applications* **51**(1), 259–277 (2012).

[27] Nocedal, J. and Wright, S. J., "Conjugate gradient methods," *Numerical optimization* , 101–134 (2006).

[28] Powell, M. J., "An efficient method for finding the minimum of a function of several variables without calculating derivatives," *The computer journal* **7**(2), 155–162 (1964).

[29] Xin, Z., Zhijun(Ryan), L., Andrey, N., Thomas, D., and Alexis, T., "AOM Common Test Conditions v2.0," *Alliance for Open Media, Codec Working Group Output Document* **CWG/B075o** (2021). https://aomedia.org/docs/CWG-B075o_AV2_CTC_v2.pdf.

[30] ITU-T and ISO/IEC, "HDRTools pacakge [Online]," (2015). Available: https:gitlab.com/standards/HDRTools.

[31] Recommendation, I., "Subjective video quality assessment methods for multimedia applications," (2021). ITU-T P.910.

[32] Netflix, "Netflix open content." Available: https://opencontent.netflix.com/.

[33] Josef, A., Olof, L., Marcus, L., and Fredrik, L., "SVT OpenContent Video Test Suite 2022– Natural Complexity," (2022). Sveriges Television AB, https://www.svt.se/open/en/content/.

[34] Cable Television Laboratories, I., "4K Video Set," (2014). https://www.cablelabs.com/4k.

[35] Liu, Z., Mukherjee, D., Lin, W.-T., Wilkins, P., Han, J., and Xu, Y., "Adaptive multi-reference prediction using a symmetric framework," *Electronic Imaging* **2017**(2), 65–72 (2017).

[36] Chen, C., Han, J., and Xu, Y., "A hybrid weighted compound motion compensated prediction for video compression," in [*Picture Coding Symposium (PCS)*], 223–227 (2018).

[37] Netflix, "VMAF - Video Multi-Method Assessment Fusion," (2016). https://github.com/Netflix/vmaf.

[38] Xiph Org, F., "Are We Compressed Yet, AWCY Source[Online]," (2015). https://github.com/xiph/awcy.

[39] Lin, J. Y., Liu, T.-J., Wu, E. C.-H., and Kuo, C.-C. J., "A fusion-based video quality assessment (FVQA) index," in [*Signal and Information Processing Association Annual Summit and Conference (APSIPA)*], (2014).