HAL
open science

Rate Allocation in Predictive Video Coding Using a Convex Optimization Framework

Aniello Fiengo, Giovanni Chierchia, Marco Cagnazzo, Béatrice Pesquet-Popescu

To cite this version:

Aniello Fiengo, Giovanni Chierchia, Marco Cagnazzo, Béatrice Pesquet-Popescu. Rate Allocation in Predictive Video Coding Using a Convex Optimization Framework. IEEE Transactions on Image Processing, 2017, 26 (1), pp.479 - 489. 10.1109/TIP.2016.2621666 . hal-01718496

HAL Id: hal-01718496

https://hal.science/hal-01718496v1

Submitted on 27 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Rate allocation in predictive video coding using a convex optimization framework

Aniello Fiengo, *Student Member, IEEE*, Giovanni Chierchia, *Member, IEEE*,
Marco Cagnazzo, *Senior Member, IEEE*, Béatrice Pesquet-Popescu, *Fellow, IEEE*

***Abstract*— Optimal rate allocation is among the most challenging tasks to perform in the context of predictive video coding, because of the dependencies between frames induced by motion compensation. In this paper, using a recursive rate-distortion model that explicitly takes into account these dependencies, we approach the frame-level rate allocation as a convex optimization problem. This technique is integrated into the recent HEVC encoder, and tested on several standard sequences. Experiments indicate that the proposed rate allocation ensures a better performance (in the rate-distortion sense) than the standard HEVC rate control, and with a little loss w.r.t. an optimal exhaustive research which is largely compensated by a much shorter execution time.**

***Index Terms*—Video coding, rate distortion, convex optimization, bit allocation.**

I. Introduction

DIGITAL video technology has recently experienced a proliferation of new applications, such as video conferencing, ultra high-definition television broadcasting, and cinema/large-screen digital imagery. In order to respond to the increasing requirement for an efficient video coding, the new standard H.265/High Efficiency Video Coding (HEVC) [1] has been formally established to substitute its predecessor H.264/MPEG4 Advanced Video Coding (AVC) [2]. In these standards, a key role in improving the visual quality of encoded videos over capacity varying channels is played by the rate allocation, which is responsible to efficiently distribute the bit budget among frames and coding blocks. More specifically it represents an important part of the rate control module of a video encoder, which involves two steps: *bit allocation (BA)*, where the total bit budget is actually allocated to frames and coding units, and *quantization selection*, where the quantization parameter (QP) is adjusted in function of the allocated bits.

The goal of rate allocation is to achieve an optimal trade-off between rate and distortion, under a time-varying constraint imposed to one of such variables. To this end, a large panel of techniques have been proposed in the literature to select the coding parameters. Many conventional coding schemes tend to make these choices frame by frame [3]. However, it is widely recognized that, from a rate-distortion (R-D) standpoint, the optimal choice for a single frame may be potentially suboptimal for encoding the remaining frames. This is due to the motion compensation, which carries the quantization error of a coded frame into the prediction of successive ones, resulting in quantization error drifting when the latter are coded. Consequently, an allocation scheme that takes into account the dependency between frames yields a significant bit reduction [4], [5].

The focus of this paper is on optimal rate allocation in predictive video coding, for which it is of paramount importance to exploit the dependency between frames induced by motion compensation. To this end, we propose a convex optimization approach for *exactly* and *efficiently* solving the frame-level rate allocation problem (within the limit of the accuracy of our model). In addition to more detailed developments, this paper extends our preliminary work [6] by integrating our method into the HEVC encoder, and providing a large number comparisons with state-of-the-art methods.

A. Related work

The optimal rate allocation is a non-trivial problem, because of the frame-level dependency induced by motion compensation. In order to circumvent this difficulty, one can allocate the bit budget independently for each frame, yielding a simpler formulation that can be tackled with standard numerical methods, such as Lagrangian optimization [7] or dynamic programming [8]–[10]. This approach can be improved by using a more accurate R-D model involving polynomials [11], logarithms [12], [13], splines [14]–[16], or ρ-domains [17], [18].

Given the suboptimal nature of frame-by-frame approaches, a substantial research effort has been made in the last decade to enlarge the optimization scope from a single frame to a group of frames (GOP). One of the first attempts in this sense was made by Ramchandran *et al.* [4], who used an operational R-D framework to search the optimal combination of quantization parameters (QP) for the whole GOP: once evaluated all the couples $(R_n(Q_1, Q_2, \cdots, Q_i), D_n(Q_1, Q_2, \cdots, Q_i))$, where $\{Q_n, D_n(Q_n), R_n(Q_n)\}$ refer to QP, distortion, and bit rate for frame n, the method performs an exhaustive search of the R-D point with the minimum Lagrangian cost under a total bit rate constraint. Although it was conceived for the allocation at frame level, this method has been extended to different coding levels, such as blocks within the same frame [19], [20]. While this methods guarantees the R-D optimality, it requires to evaluate a huge set of points for each frame, making the computational burden very high.

Aniello Fiengo, Marco Cagnazzo (corresponding author), and Béatrice Pesquet-Popescu are with Institut Mines-Télécom, Télécom ParisTech, CNRS LTCI, Paris, France (e-mail: first.last@telecom-paristech.fr).

Giovanni Chierchia is with Université Paris-Est, LIGM UMR 8049, CNRS, ENPC, ESIEE Paris, UPEM, F-93162 Noisy-le-Grand, France (e-mail: giovanni.chierchia@esiee.fr).

The obstacle encountered by search-based techniques is that the R-D behavior of a residual frame is unknown before encoding the frames from which it was predicted. A possible approach to circumvent this difficulty amounts to estimating the distortion of such residuals on the basis of the encoded version of their reference frames [21]–[24]. Although the optimality is not guaranteed (due to the inaccuracy of the model), these methods can greatly decrease the computational complexity and yield promising results.

Within this context, a more theoretical approach dates back to Uz *et al.* [25], who proposed a recursive R-D model to describe the dependency between frames, and approached the frame-level rate allocation from a Lagrangian perspective. This recursive model was later extended at pixel level for heuristically selecting the optimal quantization parameters [26]. More recently, a similar approach was used to formulate the rate allocation as a non-convex optimization problem, thus requiring a sequence of approximations to be solved [27]. The principle behind the recursive approach was also used in distributed video coding (DVC) to analyze the R-D performance of new schemes for multi-view DVC [28].

Some other works focus on minimizing the maximum distortion of encoded frames (MINMAX), instead of the usual average of distortions (MINAVE), as the former criterion might lead to some benefits to the visual perception [29]. The MINMAX criterion can be optimized through dynamic programming [30], multi-pass approaches [31], [32], or ad-hoc iterative methods with lower encoding complexity [33]. However, it is not clear whether the MINMAX is always a better criterion than the more classical one.

B. Contributions

The methods proposed in the literature to deal with the frame-level rate allocation are either theoretical in nature [25], computationally demanding [4], approximated [27], or based on the MINMAX criterion [33]. In this paper, we propose an efficient solution to exactly solve the frame-level rate allocation problem formulated with the MINAVE criterion. Our approach is based on a recursive R-D model in which the error variance of a residual frame is decoupled in two terms: the distortion of the frame used to build the prediction, and the inaccuracy of motion estimation. Although being slightly similar to [25], the proposed technique presents three contributions. Firstly, we formulate a more general R-D model which allows us to better represent the intrinsic non-stationarity in a GOP. Secondly, we present an efficient algorithm based on modern convex optimization [34]–[39], in order to find the optimal solution in limited time even for hundreds of frames. Third, we integrate the proposed algorithm in the H.265/HEVC encoder, and finally we compare the R-D performance of our approach with the rate-control module of H.265/HEVC and the Ramchandran method [4].

C. Organization

The paper is organized as follows. In Section II, we illustrate the proposed R-D model, after briefly reviewing the hybrid video encoder and the HEVC rate control. In Section III, we formulate the frame-level rate allocation problem, and we propose an efficient algorithm to solve it. In Section IV, we compare the performance of the proposed approach to the standard H.265/HEVC rate control, and the Ramchandran method [4]. Finally, the conclusion is drawn in Section V.

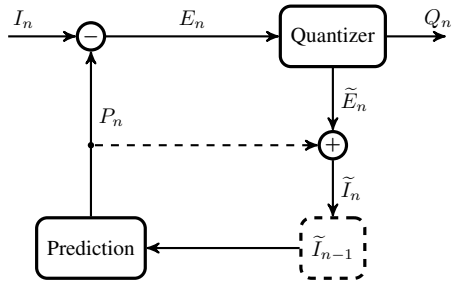

Figure 1. Predictive video encoder.

II. Rate-distortion model

The generic architecture of a hybrid video encoder is synthetically presented in Fig. 1. To take into account the temporal dependency between frames, the video sequence is organized in a group of pictures (GOP), which we denote by $I_0, \ldots, I_{N-1}$. The first frame I_0 is independently coded while the remaining frames $I_1, \ldots, I_{N-1}$ are jointly coded through a motion-compensated prediction scheme.

In the IPP…P setting, the frame I_n (with $n = 1, \ldots, N-1$) is subtracted from a prediction P_n, yielding a residual frame $E_n = I_n - P_n$ that is sent to the quantizer. The latter, besides the quantization indexes Q_n, produces the encoded residual $\widetilde{E}_n$ that serves to compute the encoded frame $\widetilde{I}_n = \widetilde{E}_n + P_n$, which will contribute to the prediction of the next frame.

The temporal prediction is based on motion estimation (ME) and compensation (MC). The ME consists of estimating the motion occurred between the input frame I_n and the previous encoded frame $\widetilde{I}_{n-1}$. The result is a motion vector field $\mathbf{v}_n$ that servers to generate the prediction, yielding $P_n = f(\widetilde{I}_{n-1}, \mathbf{v}_n)$ where f is MC function that, in its simplest form, translates the blocks of $\widetilde{I}_{n-1}$ according to the motion vectors in $\mathbf{v}_n$.

A. Rate allocation in HEVC

In order to effectively reduce the spatial and temporal redundancy of a video sequence, it is crucial to optimally distribute the bit budget within the sequence, because this directly affects the visual distortion of encoded frames. In this regard, the rate control of H.265/HEVC allocates the bit budget at three different levels: GOP level, frame level, and coding unit (CU) level, as depicted in Fig. 2.

For the GOP level, the bit allocation (BA) consists in a uniform assignment of the total bit budget for the sequence over the GOPs, with a possible fluctuation related to the number of the remaining frame to encode and the bits still available. The frame level BA works in a similar way, with a weighting mechanism to vary the bit assignment for each frame in the current GOP according to their different hierarchical position.

Our work aims to replace the frame level BA (which does not take into account the temporal redundancy for the bit assignment), without modifying the units responsible for the QP determination (which is performed with the R–λ model [40]), or the overlaying coding tree unit (CTU) level.

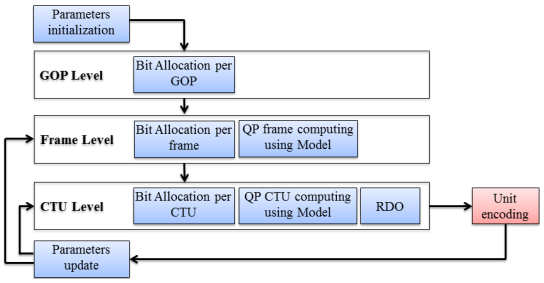

Figure 2. Rate control in HEVC.

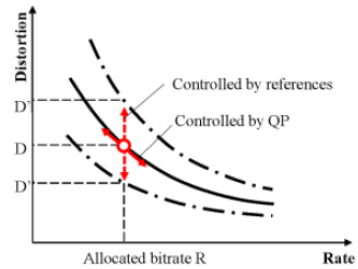

Figure 3. In dependent bit allocation the R-D point of frame *n* is not only controlled by its own QP but also affected by the QPs of the references [20].

B. Recursive R-D model

We propose to allocate the bit budget by resorting to a parametric R-D model. According to the R-D theory [41], the distortion of the frame $\widetilde{I}_n$ encoded at a high rate r_n reads

$$D_n = \mathbb{E}\left\{(I_n - \widetilde{I}_n)^2\right\} = \alpha_n\, \sigma_n^2\, 2^{-2\, r_n}, \tag{1}$$

where $\alpha_n > 0$ is a free parameter, and σ_n^2 is the variance of the frame I_0 when $n = 0$ or the residual E_n for $n \geq 1$. Unfortunately, the above R-D model is proven to be inaccurate in the context of predictive video coding [42], because the distortion D_n also depends on the frame $\widetilde{I}_{n-1}$ used to build the motion-compensated prediction, as shown in Fig. 3.

The behavior in Fig. 3 can be explained by the fact that the residual frame is factorisable in two terms, namely

$$\begin{aligned} E_n &= I_n - P_n \\ &= \underbrace{\left[I_n - f(I_{n-1}, \mathbf{v}_n)\right]}_{z_n} - \underbrace{\left[f(\widetilde{I}_{n-1}, \mathbf{v}_n) - f(I_{n-1}, \mathbf{v}_n)\right]}_{q_n}, \end{aligned} \tag{2}$$

where z_n is the prediction error of the *original* frame I_{n-1}, and q_n is the quantization error of the *encoded* frame $\widetilde{I}_{n-1}$. By assuming that z_n and q_n are uncorrelated, and $\mathbb{E}\{E_n\} = 0$, the variance of the residual can be decomposed as

$$\sigma_n^2 = \underbrace{\mathbb{E}\left\{z_n^2\right\}}_{M_n} + \mathbb{E}\left\{q_n^2\right\}, \tag{3}$$

where M_n is the mean square error of the motion estimation. The second term is exactly the distortion of $\widetilde{I}_{n-1}$, since

$$\begin{aligned} \mathbb{E}\left\{q_n^2\right\} &= \mathbb{E}\left\{\left[f(\widetilde{I}_{n-1}, \mathbf{v}_n) - f(I_{n-1}, \mathbf{v}_n)\right]^2\right\} \\ &= \mathbb{E}\left\{\left[\widetilde{I}_{n-1} - I_{n-1}\right]^2\right\} \\ &= D_{n-1}, \end{aligned} \tag{4}$$

under the hypothesis of wide-sense stationarity. Consequently, the R-D behavior of a frame I_n can be modeled as [24], [25]

$$D_n = \alpha_n\, (M_n + D_{n-1})\, 2^{-2\, r_n}, \tag{5}$$

where, for $n = 0$, M_0 denotes the variance of I_0 and $D_{-1} = 0$. Hence, for $n \geq 1$, the distortion D_n is controlled by two terms: the rate r_n used to encode $\widetilde{I}_n$, and the distortion D_{n-1} of its encoded reference $\widetilde{I}_{n-1}$, leading to a recursive R-D model.

C. Proposed approach

In this work, we extend the recursive R-D model in (5) by parameterizing the exponential decay, in order to better represent the fact that the R-D behavior within a GOP can change from a frame to another. This allows us to model the R-D function of a frame I_n as follows

$$D_n = \alpha_n\, (M_n + D_{n-1})\, e^{-\beta_n\, r_n}, \tag{6}$$

where (α_n, β_n) are parameters to be estimated within the GOP.

The key point of the above model is that the distortion D_n actually depends on all the frames involved in the chain of predictions leading to I_n. This can be demonstrated by the induction principle (See Appendix A), leading to

$$D_n = \sum_{\ell=0}^{n} \alpha^{(\ell,n)}\, M_\ell\, e^{-\sum_{j=\ell}^{n} \beta_j\, r_j}, \tag{7}$$

where $\alpha^{(\ell,n)} = \prod_{j=\ell}^{n} \alpha_j$. Since we have a sum of exponentials composed with a linear transformation, we can conclude that the distortion D_n is a convex function of rates $r_0, \ldots, r_n$. This property will be of crucial importance for designing the rate allocation algorithm proposed in Section III.

D. Estimation of model parameters

The parameter M_n can be obtained directly from the encoder by using a QP equal to 1 for each frame in the GOP. Conversely, the estimation of parameters (α_n, β_n) is performed by encoding the GOP with different QPs. For every QP, we record the values of D and R produced at the encoder output, obtaining a set of R-D points for each frame:

$$\mathcal{S}_n = \left\{\left(r_n^{(s)}, \log(D_n^{(s)})\right) \mid s = 1, \ldots, S\right\}. \tag{8}$$

Hereabove, we take the logarithm of $D_n^{(s)}$ so as to convert the R-D model in (6) into a linear relationship

$$\log(D_n) = \log\left(\alpha_n\, (M_n + D_{n-1})\right) - \beta_n\, r_n. \tag{9}$$

This allows us to estimate the parameter β_n and the quantity $\log\left(\alpha_n\, (M_n + D_{n-1})\right)$ through a simple linear regression. Then, we obtain the parameter α_n with simple algebra, as we have already estimated M_n and D_{n-1} (for the latter, we use the lowest distortion recorded in $\mathcal{S}_{n-1}$).

In order to assess the above estimation procedure, we compared the R-D model proposed in (6) to the classical R-D model (5), and to the model proposed in [27]. Table I collects the R^2 indexes [43] for the various models fitted to different sequences. As observed from the table, for all the test sequences, the proposed model gives higher R^2 values, which indicates higher accuracy.

Table I
DISTORTION ESTIMATION ACCURACY FOR DIFFERENT SEQUENCES

Sequence	R^2 *with* (6)	R^2 *with* (5)	R^2 *with* [27]
BasketballDrive	0.992	0.986	0.937
BQTerrace	0.998	0.921	0.727
Cactus	0.999	0.922	0.913
Kimono1	0.999	0.941	0.972
ParkScene	0.999	0.947	0.963
BasketballDrill	0.974	0.940	0.970
BQMall	0.999	0.916	0.975
RaceHorses (832x480)	0.998	0.952	0.934
BlowingBubbles	0.999	0.937	0.916
BQSquare	0.997	0.903	0.925
RaceHorses(416x240)	0.991	0.963	0.893

III. FRAME-LEVEL RATE ALLOCATION

Optimal rate allocation consists of finding the rate vector $r = (r_0, \ldots, r_{N-1})$ that minimizes the global distortion while keeping the total rate under a given budget $\eta > 0$, that is

$$\underset{r \in [0,+\infty[^N}{\text{minimize}} \quad \sum_{n=0}^{N-1} D_n(r_0, \ldots, r_n) \quad \text{s.t.} \quad \sum_{n=0}^{N-1} r_n \leq \eta. \quad (10)$$

By using the distortion function proposed in (7), the above formulation boils down to a convex optimization problem that can be *efficiently* and *exactly* solved by recent primal-dual proximal algorithms [38], as detailed in the following.

A. Optimization problem

To gain some insight into the solution of Problem (10), we can introduce a vector $u = (u_{n\ell})_{0 \leq n \leq N-1, 0 \leq \ell \leq n}$ defined as

$$u_{n\ell} = -\sum_{j=\ell}^{n} \beta_j r_j, \quad (11)$$

which allows us to express the global distortion as a separable sum of exponentials

$$\varphi(u) = \sum_{n=0}^{N-1} \sum_{\ell=0}^{n} \alpha^{(n,\ell)} M_\ell \, e^{u_{n\ell}}. \quad (12)$$

Therefore, Problem (10) can be reformulated as follows

$$\underset{r \in \mathbb{R}^N}{\text{minimize}} \;\; \varphi(\mathsf{L}r) + \iota_C, \quad (13)$$

where $\mathsf{L} \colon \mathbb{R}^N \mapsto \mathbb{R}^{\frac{N(N+1)}{2}}$ is the linear operator that maps the vector $r \in \mathbb{R}^N$ into the vector $u \in \mathbb{R}^{\frac{N(N+1)}{2}}$ defined in (11), $C \subset \mathbb{R}^N$ is the nonempty closed convex set defined as

$$C = \{r \in [0, +\infty[^N \;\; \Big| \;\; \sum_{n=0}^{N-1} r_n \leq \eta\}, \quad (14)$$

and ι_C denotes the indicator function of C, equal to 0 if $r \in C$ and $+\infty$ otherwise, so as to enforce the bit budget constraint.

Among the many approaches proposed in the literature to solve the above optimization problem, we resort here to *proximal algorithms* [34]–[39], because they allow us to enforce the bit budget as a hard constraint, instead of resorting to a Lagrangian formulation that would require us to determine the corresponding Lagrangian multiplier.

Algorithm 1 FBPD [35] for solving Problem (13)

Inputs
- model parameters: $(\alpha_n, \beta_n, M_n)_{0 \leq n \leq N-1}$
- budget: η

Output
- rate vector: r

Initialization
- select $r^{[0]} \in \mathbb{R}^N$ and $u^{[0]} \in \mathbb{R}^{\frac{N(N+1)}{2}}$
- set $\tau > 0$ and $\gamma > 0$ such that $\tau\gamma\|\mathsf{L}\|^2 \leq 1$

For $i = 0, 1, \ldots$

$$\begin{cases} r^{[i+1]} = P_C\big(r^{[i]} - \tau\, \mathsf{L}^\top u^{[i]}\big) \\ u^{[i+1]} = \operatorname{prox}_{\gamma\varphi^*}\Big(u^{[i]} + \gamma\, \mathsf{L}\big(2r^{[i+1]} - r^{[i]}\big)\Big) \end{cases}$$

B. Primal-dual proximal algorithm

The cornerstone of proximal algorithms is the *proximity operator* of a convex function f from an Hilbert space $\mathcal{H}$ to $]-\infty, +\infty]$, defined as

$$(\forall x \in \mathcal{H}) \quad \operatorname{prox}_f(x) = \underset{y \in \mathcal{H}}{\operatorname{argmin}} \; \frac{1}{2}\|y - x\|^2 + f(y). \quad (15)$$

The proximity operator can be interpreted as a sort of gradient descent step for the function f, since

$$p = \operatorname{prox}_f(x) \quad \Leftrightarrow \quad p \in x - \partial f(p), \quad (16)$$

where $\partial f(p) \subset \mathcal{H}$ denotes the subdifferential of f at p.[1]

For the resolution of Problem (13), we employ the Forward-Backward Primal-Dual algorithm [35], which consists of iterating the following steps for a given initialization $(r^{[0]}, u^{[0]})$:

$$(\forall i \in \mathbb{N}) \quad \begin{cases} r^{[i+1]} = \operatorname{prox}_{\tau\iota_C}\Big(r^{[i]} - \tau \mathsf{L}^\top u^{[i]}\Big) \\ u^{[i+1]} = \operatorname{prox}_{\gamma\varphi^*}\Big(u^{[i]} + \gamma\, \mathsf{L}\big(2r^{[i+1]} - r^{[i]}\big)\Big) \end{cases} \quad (17)$$

where $\tau > 0$ and $\gamma > 0$ are two algorithmic parameters, whereas φ^* denotes the convex conjugate of the function φ.

The main advantage of proximity operators is that the functions to be optimized do not need to be differentiable. This allows us to enforce the constraint C defined in (14) through its indicator function, as the proximity operator of ι_C coincides to the *orthogonal projection* onto C, yielding

$$\operatorname{prox}_{\iota_C}(x) = P_C(x) = \underset{y \in C}{\operatorname{argmin}} \, \|y - x\|^2. \quad (18)$$

Consequently, the iterations in (17) lead to Algorithm 1, which is guaranteed to converge to a solution to Problem (13) for a suitable choice of τ and γ [35], and can be efficiently implemented on both multicore and GPGPU architectures [44].

[1] $\partial f(p) = \{t \in \mathcal{H} \;\; \big| \;\; (\forall y \in \mathcal{H}) \;\; \langle y - p \mid t\rangle + f(p) \leq f(y)\}$, which reduces to the singleton $\partial f(p) = \{\nabla f(p)\}$ when f is differentiable.

C. Implementation details

Algorithm 1 has the ability to decompose the optimization process into elementary steps, such as P_C, $\mathrm{prox}_{\varphi^*}$, L and $\mathsf{L}^\top$. For what concerns the linear operator L defined in (11), its adjoint $\mathsf{L}^\top u$ yields a vector $r \in \mathbb{R}^N$ such that

$$r_n = -\beta_n \sum_{j=n}^{N-1} \sum_{\ell=0}^{n} u_{j\ell}. \tag{19}$$

In order to set the parameters τ and γ in Algorithm 1, one needs to compute the spectral norm of L, which is equal to the largest eigenvalue of the positive-semidefinite matrix $\mathsf{L}^\top\mathsf{L}$. This quantity can be estimated with Gelfand's Formula [45].

Moreover, the projection onto the set C can be efficiently computed with the linear-time algorithm proposed in [46], while the proximity operator of φ^* reads

$$\mathrm{prox}_{\gamma\varphi^*}(u) = \left[\gamma W\left(\frac{\alpha^{(n,\ell)}\, M_\ell\, e^{u_{n\ell}/\gamma}}{\gamma}\right)\right]_{0\le n\le N-1, 0\le \ell\le n} \tag{20}$$

where W denotes the Lambert W function.

IV. Experimental Results

In this section, experimental results are presented to validate our convex optimization algorithm for frame level bit allocation on the HEVC platform. We embedded our method into a standard version of the HEVC encoder (*HM.13 RExt-6.0*),[2] which is able to encode only the Luma component of a video sequence, in order to better evaluate the validity of our R-D model and easily evaluate the overall performances.

In our experiments, we used several sequences of class A, B, C, D, and E from HEVC common test conditions [47], each with 150 or 200 frames, presenting significantly different video characteristics in terms of smooth and complex scenes, and slow and fast motion. The configuration of the encoder is set as follows: GOP structure of I-P-P-P with a single reference frame (each P frame in a GOP only depends on the previous one), in order to have the best correspondence with the proposed R-D model; the maximum search range for motion estimation is set to 64. Four different target bit rates are selected for each sequence, in order to cover a PSNR range from 35 dB to 42 dB, as reported in Table II.

We compared our method to the standard R-λ rate control scheme of HEVC, and the bit allocation method in [4]. We used the latter as an optimal benchmark, as it performs an exhaustive research among all the possible R-D operating points, in order to find the one with the minimum Lagrangian cost for a constant λ under a total bit rate constraint.

A. Implementation

The experimental framework is composed of three steps, as showed in Fig. 4 and explained in the following.

- **Parameter estimation (S)**. A standard version of HEVC reference software HM.13 RExt-6.0 is used to encode the sequences at different QPs: 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40. The corresponding R-D points recorded at the encoder output are then used to estimate the parameters (α_n, β_n) for each frame, as explained in Section II-D. Furthermore, the parameters M_n are obtained by encoding the sequences with a modified version of HM.13 RExt-6.0 that extrapolates M_n for each frame, using a QP equal to 1.

- **Bit allocation (BA)**. The estimated parameters are used in Algorithm 1 to optimally allocate the rates under a bit budget constraint. The different target bit rates used for each sequence are reported in Table II. For a fixed target bit rate, the algorithm allocates the bits to each frame of a GOP by minimizing the global distortion.

- **Encoding (E)**. The encoder is a modified version of *HM.13 RExt-6.0*, where the rate control module accepts the rates computed in the previous step.

[2]Source code available at *https://hevc.hhi.fraunhofer.de/svn/* under the branch *svn_HEVCSoftware/tags/HM-13.0+RExt-6.0/*

Table II
Sequences used in our tests. Values of target bit rare are set according to the resolution, frame rate and movement characteristics of each sequence in order to cover a PSNR range from 35 dB to 42 dB

Sequence	*Resolution*	*Frame Rate* [fps]	*Target Bit Rates* [Mbps]
SteamLocomotive (10 bit)	2560 x 1600	60	10 25 70 140
NebutaFestival (10 bit)		60	100 200 250 300
PeopleOnStreet		30	15 25 45 80
Traffic		30	15 30 40 50
BasketballDrive	1920 x 1080	50	30 35 40 45
BQTerrace		60	30 35 40 45
Cactus		50	10 15 20 30
Kimono1		24	10 15 20 25
ParkScene		24	10 15 20 25
BasketballDrill	832 x 480	50	2 3 4 5
BQMall		60	2 3 4 5
PartyScene		50	2 3 4 5
RaceHorses		30	1 2 3 4
BasketballPass	416 x 240	50	1 1.5 2 2.5
BlowingBubbles		50	1 1.5 2 2.5
BQSquare		60	1 1.5 2 2.5
RaceHorses		30	0.5 1 1.5 2
FourPeople	1280 x 720	60	2 3 4 5
KristenAndSara		60	2 3 4 5
vidyo1		60	2 3 4 5
vidyo3		60	2 3 4 5
vidyo4		60	2 3 4 5

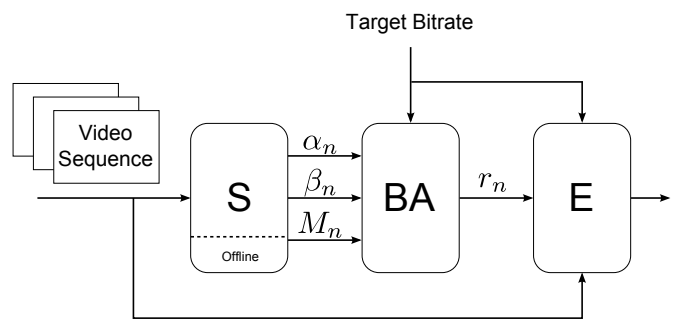

Figure 4. Flowchart of the proposed method: parameter estimation (block S), bit allocation (block BA), and video encoding (block E).

B. Comparison with R-λ rate control

To assess the validity of our bit rate allocation method, we compared it with the standard rate control algorithm of HEVC. Bjontegaard bitrate (BDBR) and Bjontegaard PSNR (BDPSNR) [48] are deployed to measure the average performance over the four target bitrates for each sequence. For BDPSNR, a positive number indicates a PSNR gain achieved by our method at the same bit rate; for BDBR, a negative number in the table indicates a rate reduction achieved at the same PSNR quality. Table III shows the the rate-distortion performance of HM R-λ and our scheme for each sequence, and the average results for each class. As shown in the table, our method outperforms the conventional encoder in terms of average PSNR for all video sequences at different bitrate; in particular, we have a consistent gain for all the Class B sequences with a maximum of 1.2 dB in terms of Δ PSNR and a bit rate reduction of 34.3 %.

Table III also reports the execution times of the compared methods. For our method, we show the times relative to the parameter estimation (block **S** in Fig. 4), the convex optimization algorithm (block **BA** in Fig. 4), and the video encoding (block **E** in Fig. 4). The parameter estimation takes the 90% of the total time (but is executed only once for each sequence), whereas the convex optimization algorithm takes a negligible fraction of the total time.

Another advantage of the proposed method over HM R-λ is a better stability in terms of bit rate. As shown in Fig. 5, HM produces strong fluctuations around the target bit rate over the GOPs, while our method is much more stable, reaching the target bit rate for every GOP. Those fluctuations have a strong impact on the *subjective quality* of the video sequences, which dramatically decrease for the last GOPs, as shown in the right plot of Fig. 5. Moreover, Fig. 6 and Fig. 7 illustrate some subjective quality results of the proposed method and the rate control of HEVC. For both figures the left pictures correspond to our method, whereas the right ones correspond to HM R-λ. The above two pictures of Fig. 6 are the frame No. 90 in *BasketballDrive* sequence at 30 Mbps and the below two pictures are the frame No. 198 in the same sequence at the same bit rate; the above two pictures of Fig. 7 are the frame No. 90 in *Kimono* sequence at 10 Mbps and the below two pictures are the frame No. 198 in the same sequence at the same bit rate. The two figure give a visual example of the quality fluctuations pointed out in Fig. 5, showing a constant visual quality for our method over the whole sequence, while the quality loss of HM results in severe blocking artifacts.

C. Comparison with an exhaustive search method

In order to evaluate the performance of our method, we decided to compare it with the bit allocation technique proposed by Ramchandran *et al.* [4]; the optimality of this method is guaranteed by the fact that it chooses the quantizers combination with the minimum Lagrangian cost among all the possible combinations. In [4], two *pruning condition* implied by the monotonicity property of the R-D curves, which are proven to not affect the optimality of the solution, are introduced in order to ease the computational burden, and *suboptimal*

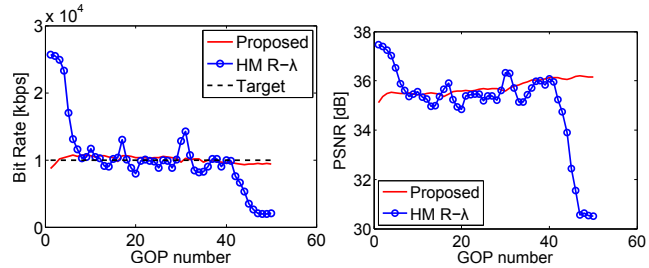

Figure 5. Bit rate vs. GOP number (left) and PSNR vs. GOP number (right) for the sequence *Cactus* at 10 Mbps. The proposed method, guarantees a uniform visual quality over the whole sequence.

heuristics are proposed to obtain an additional reduction of the complexity in exchange to performance lost. We compared our method to this suboptimal, but feasible, solution in terms of R-D performance and execution time.

The first step to implement [4] was the *data generation* phase: in order to generate the R-D, we encoded each sequence with all the possible QP combinations along the frames. The complexity of this phase is clearly exponential with the number of frames, and it is possible to calculate the number of encoding operations per sequence as $\sum_{n=1}^{N} q^n$, where N is the number of encoded frames, and q is the number of used QP. In order to have a reasonable encoding time, we choose $N = 4$ and $q = 9$, using the QPs 20, 22, 25, 27, 30, 32, 35, 37, 40; again, we used as encoder HM.13 RExt-6.0 with the rate control disabled, and an I-P-P-P configuration. We implemented in Matlab:

- a tree where the nodes of level n contain the R-D data of the n-th encoded frame for each quantization choice,
- the Lagrangian cost functions for a fixed value λ,
- the tree pruning conditions implied by the monotonicity of R-D curves [4] to eliminate suboptimal points.

For each sequence, we defined a set of λ values, and for each of them we selected the optimal path, that is the combination of QPs, with the minimum Lagrangian cost; among all the optimal R-D points, we selected four with a PSNR inside the range of 35 dB $\sim$ 42 dB, and used the relative bit rate values as *target bit rates* for our method. Fig. 8 shows an example of full constellation of R-D points evaluated for the sequence *Cactus* without applying the pruning conditions, and presents the four optimal points associated to the selected λ values, which are situated on the convex hull of the constellation; by applying the pruning conditions, the number of points in the constellation are cut in a percentage dependent to λ, with higher quality target (smaller λ) achieving a better reduction. Table IV presents the R-D performance of the suboptimal heuristics of [4] and our method in terms of BDBR and BDPSNR, showing that our method performs slightly worse than [4], with a difference in terms of bit rate smaller than 7% for several sequences, and an average difference of 9%.

Moreover, we compared the execution times of these methods. Table V shows the setup time of [4], that is the total encoding time to fully populate the dependency tree, obtained by summing the encoding time of the 7380 encoding operations. The cost of encoding just 4 frames using 9 QPs is measured in hundreds of hours for most of the sequences, and makes [4]

Table III
COMPARISON WITH THE HM R-λ. FOR OUR METHOD, THE TIME VALUES CONCERN THE PARAMETER ESTIMATION (S), THE CONVEX OPTIMIZATION ALGORITHM (BA), AND THE VIDEO ENCODING (E); EACH VALUE IS AN AVERAGE OF THE TIMES OBTAINED WITH FOUR FIXED TARGET BIT RATES.

Sequence	*R-D Performance*				*Time Comparison (s)*			
					Proposed			
	BDPSNR (dB)	BDBR (%)	av. BDPSNR (dB)	av. BDBR (%)	S	BA	E	HM R-λ
SteamLocomotiveTrain	0.62	-18.62	0.41	-8.68	288470	0.03	15652	21494
NebutaFestival	0.50	-5.97			480322	0.03	35806	55445
PeopleOnStreet	0.45	-8.74			333428	0.03	23474	19575
Traffic	0.10	-1.41			249457	0.05	17155	14536
BasketballDrive	1.02	-28.00	0.93	-24.85	312915	0.05	18192	13197
BQTerrace	1.01	-26.47			344397	0.03	18323	10135
Cactus	0.89	-30.73			332231	0.03	20409	12355
Kimono1	0.82	-21.21			261911	0.04	17601	13892
ParkScene	0.93	-17.84			309619	0.03	19889	12300
BasketballDrill	0.41	-9.06	0.49	-10.77	55801	0.04	2187	1548
BQMall	0.59	-12.06			60668	0.04	1864	1357
PartyScene	0.43	-9.54			78543	0.04	1963	1435
RaceHorses (832x480)	0.55	-12.41			26410	0.03	2759	1908
BasketballPass	0.12	-1.89	0.74	-11.32	11466	0.03	649	468
BlowingBubbles	0.99	-15.25			17305	0.03	756	455
BQSquare	1.32	-19.80			17673	0.03	701	385
RaceHorses (416x240)	0.52	-8.35			16571	0.03	932	646
FourPeople	0.43	-6.87	0.23	-5.86	93957	0.03	3224	2279
KristenAndSara	0.22	-6.79			89341	0.03	3409	2286
vidyo1	0.17	-6.34			87383	0.04	3411	2344
vidyo3	0.23	-4.83			73906	0.03	3320	2242
vidyo4	0.12	-4.48			82536	0.04	3383	2334

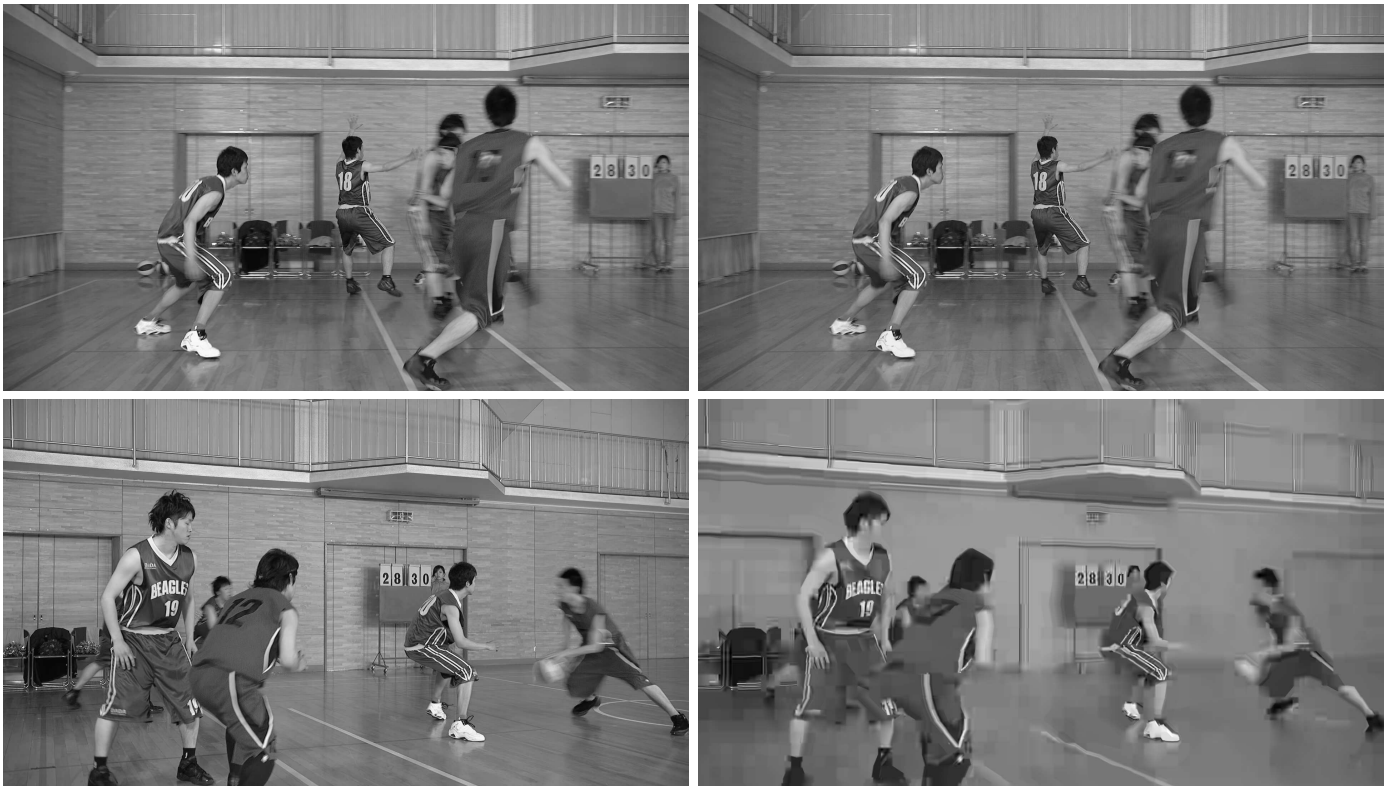

Figure 6. Subjective quality of sequence *BasketballDrive* at 30 Mbps for two different frames (Left: Proposed, Right: HM R-λ)

Figure 7. Subjective quality of sequence *Kimono* at 10 Mbps for two different frames (Left: Proposed, Right: HM R-λ)

Table IV
COMPARISON IN PERFORMANCE OF [4] AND OUR METHOD USING THE BJONTEGAARD METRICS.

Sequence	BDPSNR (dB)	BDBR (%)	av. BDPSNR (dB)	av. BDBR (%)
SteamLocomotiveTrain	-0.38	7.49	-0.19	6.10
NebutaFestival	0.51	-5.41		
PeopleOnStreet	-0.17	3.35		
Traffic	-0.72	18.96		
BasketballDrive	-0.01	0.16	-0.18	5.01
BQTerrace	-0.30	6.44		
Cactus	-0.16	6.01		
Kimono1	-0.03	0.77		
ParkScene	-0.40	12.00		
BasketballDrill	-0.79	18.96	-0.55	12.86
BQMall	-0.74	17.75		
RaceHorses (832x480)	-0.10	1.88		
BasketballPass	-0.68	12.36	-0.66	11.13
BlowingBubbles	-0.61	11.54		
BQSquare	-0.97	14.81		
RaceHorses (416x240)	-0.40	5.79		

Table V
TOTAL ENCODING TIME FOR A FULL POPULATION OF THE DEPENDENCY TREE.

Sequence	*Setup Time (h)*
SteamLocomotiveTrain	955
NebutaFestival	1577
PeopleOnStreet	798
Traffic	615
BasketballDrive	412
BQTerrace	474
Cactus	402
Kimono1	293
ParkScene	353
BasketballDrill	71
BQMall	78
RaceHorses (832x480)	100
BasketballPass	17
BlowingBubbles	31
BQSquare	26
RaceHorses (416x240)	28

impossible to use in practical applications. More reasonable times are obtained when [4] uses a tree pruning strategy, as it leads to a smaller number of encoding operations.

Table VI presents a comparison of execution times between our method and the one in [4]. The global time of our method is given by the sum of three contributions: the parameters estimation (90% of the global time), the convex optimization algorithm (a small fraction of the global time), and the video encoding (about the 10% of the global time). The global time of [4] is given by the sum of two contributions: the dataset generation (99% of the total time), and the search for the best R-D point. These times depend on the chosen λ value: a smaller λ implies the pruning of a higher number of R-D points (less encoding operations to perform), resulting in a smaller time. Table VI shows that the method in [4] requires at least four times the time required by our method, with a difference of two magnitude orders in the worst case.

In summary, our method presents a strong gain over the standard HM R-λ rate control scheme with a better stability in terms of bit rate, whereas the small loss w.r.t. the method in [4] is largely compensated by a much faster execution time.

Table VI
COMPARISON OF GLOBAL TIMES. FOR OUR METHOD, THE TIME VALUES CONCERN THE PARAMETER ESTIMATION (S), THE CONVEX OPTIMIZATION ALGORITHM (BA), AND THE VIDEO ENCODING (E). FOR THE METHOD IN [4], THE TIMES CONCERN A LARGE λ (MAX) AND A SMALL λ (MIN).

	Setup Time (s)			*Execution Time (s)*				*Global Time (s)*		
Sequence	*Proposed*	Method in [4]		*Proposed*		Method in [4]		*Proposed*	Method in [4]	
	S	*Max*	*Min*	BA	+ E	*Max*	*Min*	S+BA+E	*Max*	*Min*
SteamLocomotiveTrain	9579	1470060	37355	0.03	193.518	61.631	1.511	9773	1470122	37356
NebutaFestival	14359	2051171	52867	0.04	1198.331	63.194	1.628	15557	2051234	52869
PeopleOnStreet	8643	1462725	27515	0.03	387.052	106.361	1.566	9030	1462831	27516
Traffic	6897	1225989	22674	0.04	289.185	110.398	1.939	7186	1226099	22676
BasketballDrive	4424	968370	12904	0.12	286.642	150.672	3.921	4711	968521	12908
BQTerrace	4830	941150	18827	0.05	350.262	125.125	1.408	5180	941275	18828
Cactus	4759	932963	17079	0.05	358.504	158.755	1.359	5118	933122	17080
Kimono1	3652	600207	10151	0.03	359.219	132.866	1.183	4011	600340	10152
ParkScene	4318	822478	14018	0.05	462.142	170.827	1.430	4780	822649	14019
BasketballDrill	785	103651	2276	0.04	48.079	86.823	1.516	833	103738	2278
BQMall	886	178867	2893	0.05	54.164	220.521	1.525	940	179088	2895
RaceHorses (832x480)	1008	149803	1858	0.04	91.317	105.121	1.530	1099	149908	1860
BasketballPass	149	38520	578	0.04	15.412	191.807	1.597	164	38712	580
BlowingBubbles	248	69695	994	0.04	24.602	180.332	1.676	273	69875	996
BQSquare	251	55029	962	0.05	22.731	169.733	1.718	274	55199	964
RaceHorses (416x240)	241	58712	972	0.05	28.290	161.830	1.780	269	58874	974

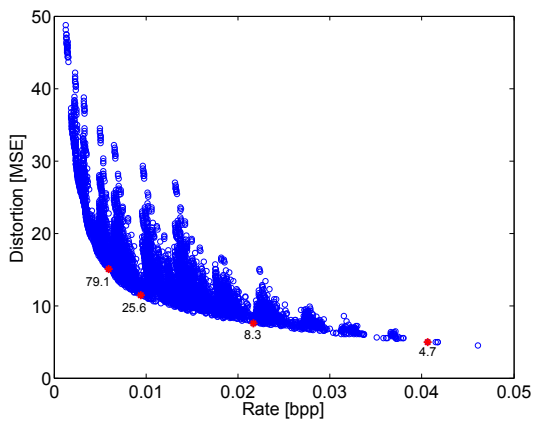

Figure 8. R-D operating points for sequence *Cactus*. Each point represents a QP combination. The points used in the comparisons are marked in red.

V. CONCLUSIONS

We have proposed a new frame level rate allocation scheme based on modern convex optimization. Firstly, an analytical rate-distortion model that explicitly takes into account the dependencies between frames was derived, for the purpose of formulating frame-level optimal rate allocation as a convex optimization problem. Then, an efficient solution based on proximal tools was provided, in order to find the *exact* solution of the R-D problem.

Our experiments indicate that the optimal rate allocation, when supported by an accurate R-D model, attains better results (in terms of Δ PSNR and % bit rate) than the standard rate control in HEVC. Moreover, the comparison with the exhaustive search method in [4] showed an average loss of 9% in terms of bit rate, which is widely compensated by an execution time that is two hundred times faster.

A more general model for for multiple reference frames and B-predictive frames, based on the introduction of a weight $\omega_{n,n-1}$ which takes into consideration the percentage of blocks of frame n predicted from frame $n-1$, is currently under development and is considered for our future work.

APPENDIX A
PROOF OF EQUATION (7)

For $n = 0, \ldots, N-1$, the distortion of the frame I_n reads

$$D_n = \alpha_n \left(M_n + D_{n-1}\right) e^{-\beta_n r_n}, \tag{21}$$

where $D_{-1} = 0$. We aim at demonstrating that

$$D_n = \sum_{\ell=0}^{n} \alpha^{(\ell,n)} M_\ell \, e^{\sum_{j=\ell}^{n} -\beta_j r_j}, \tag{22}$$

where $\alpha^{(\ell,n)} = \prod_{j=\ell}^{n} \alpha_j$.

Firstly, let us observe that, when $n = 0$, (22) reduces to

$$D_0 = \alpha_0 M_0 e^{-\beta_0 r_0}. \tag{23}$$

Now, let us assume that (22) holds for D_{n-1}, with $n \geq 1$. By replacing this expression of D_{n-1} into (21), we get

$$\begin{aligned} D_n &= \alpha_n (M_n + D_{n-1}) e^{-\beta_n r_n} \\ &= \alpha_n \Big(M_n + \sum_{\ell=0}^{n-1} \alpha^{(\ell,n-1)} M_\ell \, e^{-\sum_{j=\ell}^{n-1} \beta_j r_j} \Big) e^{-\beta_n r_n} \\ &= \alpha_n M_n e^{-\beta_n r_n} + \sum_{\ell=0}^{n-1} \alpha^{(\ell,n)} M_\ell \, e^{-\sum_{j=\ell}^{n} \beta_j r_j} \\ &= \sum_{\ell=0}^{n} \alpha^{(\ell,n)} M_\ell \, e^{-\sum_{j=\ell}^{n} \beta_j r_j}. \end{aligned} \tag{24}$$

This proves that (22) holds for every $n \in \mathbb{N}$.

REFERENCES

[1] G. Sullivan, J.-R. 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, pp. 1649–1668, 2012.

[2] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, 2003.

[3] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–703, July 2003.

[4] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and mpeg video coders," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 533–545, 1994.

[5] V. Chellappa, P. Cosman, and G. Voelker, "Dual frame motion compensation with uneven quality assignment," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 2, pp. 23–50, 2008.

[6] A. Fiengo, G. Chierchia, M. Cagnazzo, and B. Pesquet-Popescu, "A convex-optimization framework for frame-level optimal rate allocation in predictive video coding," in *Proc. of ICASSP*, Florence, Italy, 2014.

[7] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, 1998.

[8] A. Ortega and K. Ramchandran, "Rate-distortion techniques in image and video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 23–50, 1998.

[9] 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, no. 12, pp. 1533–1544, 2005.

[10] Z. Ma, M. Xu, Y. F. Ou, and Y. Wang, "Modeling of rate and perceptual quality of compressed video as functions of frame rate and quantization stepsize and its applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 5, pp. 671–682, 2012.

[11] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 246–250, Jan. 1997.

[12] B. Tao, H. A. Peterson, and B. W. Dickinson, "A rate–quantization model for mpeg encoders," in *Proc. of ICIP*, 1997, pp. 338–341.

[13] Jordi Ribas-corbera and Shawmin Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 172–185, 1999.

[14] T. André, M. Cagnazzo, M. Antonini, and M. Barlaud, "A jpeg2000-compatible full scalable video coder," *EURASIP Journal of Image and Video Processing*, vol. 2007, pp. 11, 2007.

[15] M. Cagnazzo, T. André, M. Antonini, and M. Barlaud, "A model-based motion compensated video coder with jpeg2000 compatibility," in *Proc. of ICIP*, Singapore, 2004, vol. 1, pp. 2255–2258.

[16] W. Ding and B. Liu, "Rate control of mpeg video coding and recording by rate-quantization modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 1, pp. 12–20, Jan. 1996.

[17] Z. He and S. 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, pp. 840–849, 2002.

[18] S. Wang, S. Ma, S. Wang, D. Zhao, and W. Gao, "Rate-GOP based rate control for high efficiency video coding," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, Dec. 2013.

[19] Y. Yang and S. S. Hemami, "Generalized rate-distortion optimization for motion-compensated video coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 6, pp. 942–955, 2000.

[20] C. Wang, X. Mou, and L. Zhang, "Optimization of the block-level bit allocation in perceptual video coding based on MINMAX," http://arxiv.org/abs/1511.04691, 2015.

[21] L. J. Lin and A. Ortega, "Bit-rate control using piecewise approximated rate-distortion characteristics," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 4, pp. 446–459, Aug. 1998.

[22] J. Liu, Y. Cho, Z. Guo, and C. C. Jay Kuo, "Bit allocation for spatial scalability coding of H.264/SVC with dependent rate-distortion analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 7, pp. 967–981, July 2010.

[23] S. Hu, S. Kwong, T. Zhao, and C. C. Jay Kuo, "Rate control optimization for temporal-layer scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 8, pp. 1152–1162, Aug. 2011.

[24] H. Yuan, S. Kwong, X. Wang, W. Gao, and Y. Zhang, "Rate distortion optimized inter-view frame level bit allocation method for MV-HEVC," *IEEE Trans. Multimedia*, vol. 17, pp. 2134–2146, Dec. 2015.

[25] K. Uz, J. Shapiro, and M. Czigler, "Optimal bit allocation in the presence of quantizer feedback," in *Proc. of ICASSP*, 1993.

[26] G. Valenzise and A. Ortega, "Improved video coding efficiency exploiting tree-based pixelwise coding dependencies," in *Proc. of SPIE*, San Jose, USA, Jan. 2010.

[27] C. Pang, O. Au, F. Zou, J. Dai, X. Zhang, and W. Dai, "An analytic framework for frame-level dependent bit allocation in hybrid video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 6, pp. 990–1002, June 2013.

[28] T. Maugey and B. Pesquet-Popescu, "Side information estimation and new schemes for multiview distributed video coding," *J. Vis. Commun. Image Represent.*, vol. 19, no. 8, pp. 589–599, Dec. 2008.

[29] J. M. Henderson, P. A. Weeks Jr., and A. Hollingworth, "The effects of semantic consistency on eye movements during complex scene viewing," *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 25, no. 1, pp. 210–228, Feb. 1999.

[30] G. M. Schuster, G. Melnikov, and A. K. Katsaggelos, "A review of the minimum maximum criterion for optimal bit allocation among dependent quantizers," *IEEE Trans. Multimedia*, vol. 1, no. 1, pp. 3–17, 1999.

[31] B. Xie and W. Zeng, "A sequence-based rate control framework for consistent quality real-time video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, pp. 56–71, Jan. 2006.

[32] K. Wang and J. W. Woods, "Mpeg motion picture coding with long-term constraint on distortion variation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 3, pp. 294–304, 2008.

[33] N. Cherniavsky, G. Shavit, M. F. Ringenburg, R. E. Ladner, and E. A. Riskin, "Multistage: A MINMAX bit allocation algorithm for video coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 1, pp. 59–67, 2007.

[34] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, Eds., pp. 185–212. Springer-Verlag, New York, 2011.

[35] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *J. Optim. Theory Appl.*, vol. 158, no. 2, pp. 460–479, 2013.

[36] P. L. Combettes and J.-C. Pesquet, "Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators," *Set-Valued Var. Anal.*, vol. 20, no. 2, pp. 307–330, June 2012.

[37] B. C. Vũ, "A splitting algorithm for dual monotone inclusions involving cocoercive operators," *Adv. Comput. Math.*, vol. 38, no. 3, pp. 667–681, Apr. 2013.

[38] N. Komodakis and J.-C. Pesquet, "Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 31–54, Nov. 2015.

[39] G. Chierchia, N. Pustelnik, J.-C. Pesquet, and B. Pesquet-Popescu, "Epigraphical projection and proximal tools for solving constrained convex optimization problems," *Signal Image Video P.*, vol. 9, no. 9, pp. 1737–1749, Nov. 2015.

[40] B. Li, H. Li, L. Li, and J. Zhang, "Rate control by R-lambda model for HEVC," in *Joint Collaborative Team on Video Coding of ITU-T SG*, Shanghai, China, Oct. 2012.

[41] T. M. Cover and J. M. Thomas, *Elements of Information Theory*, Wiley, New York, 1991.

[42] Z. He and S. K. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 970–982, 2002.

[43] J. L. Devore and N. R. Farnum, *Applied Statistic for Engineers and Scientists*, Duxbury, 1999.

[44] R. Gaetano, G. Chierchia, and B. Pesquet-Popescu, "Parallel implementations of a disparity estimation algorithm based on a proximal splitting method," in *Proc. of VCIP*, San Diego, USA, 2012.

[45] P. B. Lax, *Functional Analysis*, Wiley-Interscience.

[46] L. Condat, "Fast projection onto the simplex and the l1 ball," *Mathematical Programming Series A*, vol. 158, no. 1, pp. 575–585, July 2016.

[47] F. Bossen, "Common test conditions and software reference configurations," in *Joint Collaborative Team on Video Coding of ITU-T SG*, Geneva, Switzerland, 2013.

[48] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *VCEG Meeting*, 2001.