

RBF-BASED VBR CONTROLLER FOR REAL-TIME H.264/SVC VIDEO CODING

Sergio Sanz-Rodríguez, Fernando Díaz-de-María

Department of Signal Theory and Communications
Carlos III University of Madrid, Spain
{sescalona, fdiaz}@tsc.uc3m.es

ABSTRACT

In this paper we propose a novel VBR controller for real-time H.264/SVC video coding. Since consecutive pictures within the same scene often exhibit similar degrees of complexity, the proposed VBR controller allows for just an incremental variation of QP with respect to that of the previous picture, so preventing unnecessary QP fluctuations. For this purpose, an RBF network has been carefully designed to estimate the QP increment at each dependency (spatial or CGS) layer.

A mobile live streaming application scenario was simulated to assess the performance of the proposed VBR controller, which was compared to a recently proposed CBR controller for H.264/SVC. The experimental results show a remarkably consistent quality, notably outperforming the reference CBR controller.

Index Terms— Rate Control, Variable Bit Rate, Scalable Video Coding, H.264/SVC, H.264/AVC, streaming.

1. INTRODUCTION

The H.264/SVC video coding standard provides bit rate adaptation for varying channel conditions in RTP/IP-based video communications, as well as for heterogeneous devices with different display resolutions and computational capabilities [1]. It enables the extraction of a sub-stream from the high-quality bit stream so that this sub-stream can be decoded by a given target decoder. H.264/SVC provides spatial, temporal and signal-to-noise ratio (SNR) scalability [2]. Furthermore, the SNR scalability can be either coarse grain scalability (CGS), which is a special case of spatial scalability with identical pictures sizes, or medium grain scalability (MGS).

In real-time applications, the rate control algorithm (RCA) plays an important role in the video coding process. In H.264/SVC, a RCA should operate at each dependency layer to select a proper quantization parameter (QP) value for each coding unit, so that the buffer fullness is maintained at secure levels, while maximizing the reconstructed video quality. According to the target application, two types of RCA can be distinguished: constant bit rate (CBR) and variable bit rate (VBR) algorithms. In CBR a short-term average bit rate adaptation is required to ensure low-delay. However, in VBR a long-term bit rate adaptation is feasible to improve the visual quality at the expense of longer buffer delay.

Several RC schemes have been proposed for H.264/SVC [3, 4, 5] that are able to achieve a high coding efficiency by means of frame-wise bit allocation models optimized for hierarchical group of pictures (GOP) structures. However, none of them has been designed to operate in VBR conditions, which are more appropriate for streaming or broadcasting scenarios.

In this paper we propose a novel VBR controller for real-time

H.264/SVC video coding. It aims to prevent unnecessary QP fluctuations by allowing just an incremental variation of QP with respect to that of the previous picture. For this purpose, an RBF network has been carefully designed to estimate the QP increment at each dependency layer.

The paper is organized as follows. In Section 2 a brief overview of the proposed RCA is given. In Section 3, the VBR controller at each dependency layer is described in more detail. Section 4 presents and discusses the experimental results. Finally, some conclusions are drawn in Section 5.

2. SYSTEM OVERVIEW

The proposed RCA is illustrated in Fig. 1 for a SVC encoder consisting of two dependency layers. Let us denote as D the number of dependency layers, identified as $d = \{0, 1, \dots, D - 1\}$, and let us denote as $T^{(d)}$ the number of temporal layers for the d^{th} layer, identified as $t = \{0, 1, \dots, T^{(d)} - 1\}$. Each dependency layer d involves a rate controller $RC^{(d)}$ and a virtual buffer. The virtual buffer at layer d receives the contributions of layers 0 to d and simulates the encoder buffer of the corresponding sub-stream. The generation of each sub-stream depends on two fundamental parameters: the target bit rate $R^{(d)}$ and output frame rate $f_{out}^{(d)}$. It should be noted that $R^{(d)}$ must be higher than those associated with lower dependency layers, i.e., $R^{(d-i)} \leq R^{(d)}$, with $i = 0, 1, \dots, d$, since those lower dependency layers form part of the d^{th} sub-stream.

In order to encode the j^{th} picture with spatio-temporal identifier (d, t) , the $RC^{(d)}$ should provide an appropriate $QP_j^{(d)}$ value, on a frame basis, so that the QP fluctuation is minimized, while the buffer fullness $V^{(d)}$ is maintained at secure levels. To this end, the $RC^{(d)}$ module operation leans on three input parameters:

- 1) The amount of bits yield by the encoding of spatial layers from 0 to d for a given time instant. This amount of bits is usually referred to as *access unit* (AU) output bits $AU^{(d)}$ [2].
- 2) The fullness $V^{(d)}$ of the virtual buffer, which is updated from $AU^{(d)}$ and $R^{(d)}/f_{out}^{(d)}$.
- 3) The QP value used for encoding the previous picture of the same dependency layer $QP_{j-1}^{(d)}$.

A proper QP increment $\Delta QP^{(d)}$ is estimated from the two firsts, and $QP_{j-1}^{(d)}$ is used as a reference value to obtain the final quantization parameter as follows:

$$QP_j^{(d)} = QP_{j-1}^{(d)} + \Delta QP^{(d)}. \quad (1)$$

Furthermore, in the case of CGS scalability, $QP_j^{(d)}$ is lower bounded by the QP of the reference layer $QP_j^{(d-1)}$, so that a higher quality for the enhancement layer is ensured.

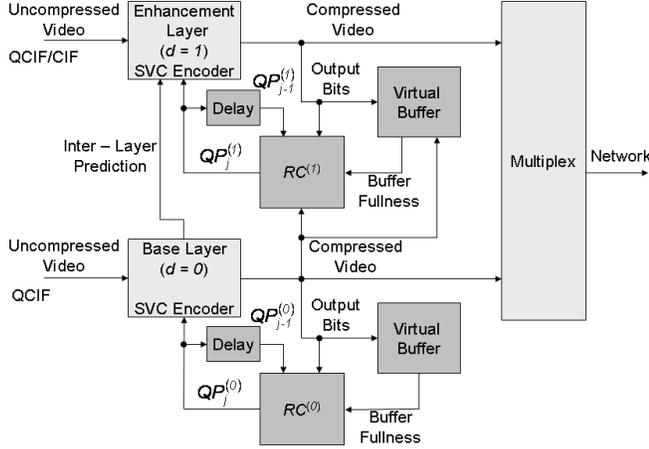


Fig. 1. Block diagram of the proposed H.264/SVC RCA for two dependency layers ($D=2$).

3. VBR CONTROLLER AT EACH DEPENDENCY LAYER

The RCA for a specific spatial/CGS layer is illustrated in Fig. 2. As it can be seen, the $RC^{(d)}$ module is organized in two stages named *parameter updating* and *RBF-based QP increment estimation*.

3.1. Parameter Updating

After encoding the $(j-1)^{th}$ picture with identifier (d, t') (t' is used instead of t because the previous picture can belong to a different temporal layer), the parameters required to estimate the QP increment for the j^{th} picture are updated. In particular, two parameters are required: 1) a normalized version of the buffer fullness, denoted as $A^{(d)}$; and 2) a normalized version of the amount of the bits generated by the AU, denoted as $B^{(d)}$. These parameters are defined as follows:

$$A^{(d)} = \frac{V^{(d)}}{BS^{(d)}}, \quad (2)$$

$$B^{(d)} = \frac{AU^{(d)}}{G^{(d)}}, \quad (3)$$

where $BS^{(d)}$ denotes the encoder buffer size, in bits, at the d^{th} dependency layer, and $G^{(d)}$ is the target bits for $AU^{(d)}$.

3.2. RBF-based QP Increment Estimation

Before encoding the j^{th} picture, the proper QP increment $\Delta QP^{(d)}$ with respect to $QP_{j-1}^{(d)}$, is estimated from $A^{(d)}$ and $B^{(d)}$. Furthermore, two additional constant parameters are considered in order to provide a solution suitable for a variety of scenarios. The first, denoted as μ , is the normalized target buffer fullness, and the second, denoted as τ , is the buffer size in seconds. Thus, the proposed $\Delta QP^{(d)}$ estimation method operates on the following input vector:

$$\mathbf{X}^{(d)} = \left(A^{(d)}, B^{(d)}, \mu, \tau \right)^T, \quad (4)$$

implicitly assuming that all the virtual buffers share the same μ and τ values. Since the input parameters μ and τ are set before starting the encoding process, the proposed estimation function can be seen as a surface whose shape depends on these constants.

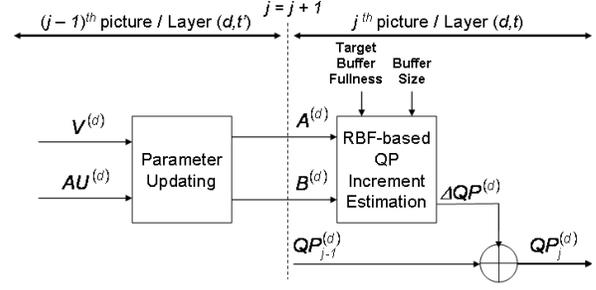


Fig. 2. Block diagram of the rate controller $RC^{(d)}$ for a specific dependency layer d .

A carefully designed RBF network is used to estimate $\Delta QP^{(d)}$ from the input vector $\mathbf{X}^{(d)}$. The RBF-based estimation obeys:

$$\Delta QP^{(d)} = \text{round} \left[w_0 + \sum_{i=1}^L w_i H_i \left(\mathbf{X}^{(d)} \right) \right], \quad (5)$$

where L is the number of basis functions $\{H_i(\mathbf{X}^{(d)})\}_{i=1, \dots, L}$, ω_i the output weights, and ω_0 the bias. It should be noted that the output of the RBF network is converted into an integer, given the discrete nature of the QP in H.264/SVC. The basis functions are Gaussian-type functions with centers \mathbf{C}_i and widths Σ , that is:

$$H_i \left(\mathbf{X}^{(d)} \right) = \exp \left(- \sum_{j=1}^4 \frac{\left(X_j^{(d)} - C_{ij} \right)^2}{\Sigma_j^2} \right). \quad (6)$$

The training of the RBF network relies on a data set containing pairs *input vector-desired output*, which have to be previously generated. Once these training data were generated, it was observed that their distributions for key (i.e., I and P pictures) and non-key pictures were different enough to justify the design of two RBF networks, one for key pictures and another for non-key pictures. Furthermore, some experiments were performed to properly dimension the RBF networks. These results led to 7 Gaussian functions in both cases. Finally, since some unnecessary fluctuations of the QP value at non-key pictures were observed in cases of stationary video complexity when the buffer level approached the target buffer fullness, a simple postprocessing stage for $\Delta QP^{(d)}$ has been proposed. This obeys:

$$\Delta QP^{(d)} = \begin{cases} -1 & \text{if } \Delta QP^{(d)} = -2 \\ 0 & \text{if } \Delta QP^{(d)} = -1 \\ 0 & \text{if } \Delta QP^{(d)} = 1 \\ 1 & \text{if } \Delta QP^{(d)} = 2 \end{cases} \quad (7)$$

In doing so, the number of small fluctuations of the QP value happening in stationary situations is minimized.

4. EXPERIMENTS AND RESULTS

The proposed VBR controller was implemented on the Joint Scalable Video Model (JSVM) H.264/SVC reference software version JSVM 9.16 [6]. In order to assess its performance, it was compared to two methods: 1) constant QP (CQP) encoding, which was used as a reference for nearly constant quality video; and 2) the CBR control algorithm described in [3]. The H.264/SVC encoder was configured for a mobile live streaming application [1]. Particularly, the spatial/CGS testing configuration proposed in [7] was used:

- Number of pictures: 900
- GOP size/Intra period: 8/32 pictures
- GOP structure: hierarchical B pictures
- Number of dependency layers: $D=5$
 - $d=0$: QCIF, $f_{out}^{(0)} = 6.25$ Hz ($T^{(0)} = 2$)
 - $d=1$: QCIF, $f_{out}^{(1)} = 12.5$ Hz ($T^{(1)} = 3$)
 - $d=2$: CIF, $f_{out}^{(2)} = 12.5$ Hz ($T^{(2)} = 3$)
 - $d=3$: CIF, $f_{out}^{(3)} = 12.5$ Hz ($T^{(3)} = 3$)
 - $d=4$: CIF, $f_{out}^{(4)} = 25$ Hz ($T^{(4)} = 4$)
- Symbol mode: CAVLC
- Rate control parameters: $\mu = 50\%$ and $\tau = 3$ s

Two sets of video sequences at 25 Hz exhibiting a variety of complexities were used in our experiments. The first set consisted of four well-known test sequences recommended in [7] for streaming applications: "Bus", "Football", "Foreman" and "Mobile". These sequences were concatenated to themselves several times to reach 900 pictures. The second set consisted of three sequences displaying scene changes: "Soccer-Mobile-Foreman", "Spiderman", and "The Lord of the Rings" (LOTR). "Soccer-Mobile-Foreman" was formed by concatenating 300 frames of each sequence. The other two were extracted from high-quality DVDs and downsampled to either QCIF or CIF format, and have been made available on-line [8].

All the sequences were encoded using the set of QP values that best approached some pre-established target bit rates. For the first group of sequences, the target bit rates were those suggested in [7]. For the second group, the following medium-quality target bit rates were selected: 64 ($d=0$), 96 ($d=1$), 192 ($d=2$), 384 ($d=3$) and 512 kbps ($d=4$). Then, the exact output bit rates obtained by CQP encoding were used as target bit rates $R^{(d)}$ for both the RCA in [3] and the VBR controller.

In order to assess the performance of the proposed VBR controller from a quality point of view, the average luminance PSNR, μ_{PSNR} , was used. The PSNR differences with respect to the CQP case, $\Delta\mu_{PSNR}$, were computed following [9]. The average results over all the test sequences in terms of $\Delta\mu_{PSNR}$ are summarized in Table 1. As can be observed, the average PSNR performance achieved by the proposed algorithm was similar to that of CQP encoding, and notably superior to that of [3].

Tables 2 and 3 show a detailed comparison of the three assessed algorithms for "Bus" and "LOTR" sequences, as representative examples of stationary and non-stationary video complexity, respectively. These results allowed us to draw two conclusions: 1) for stationary complexity sequences, the performance of the proposed method was quite close to that of CQP encoding; and 2) for non-stationary complexity sequences, the performance of the proposed method was remarkably good, exceeding even that of CQP encoding.

Representative behaviors of the buffer occupancy, PSNR, and QP evolutions at layer $d=3$ are depicted in Fig. 3 for "LOTR". When compared to [3], our proposal clearly made better use of the buffer to provide PSNR and QP evolutions closer to those of the nearly constant quality system. Furthermore, the strong correlation observed among buffer fullness, PSNR and QP evolutions reveals that the proposed VBR controller made a proper use of the buffer to successfully allocate larger amount of bits for complex scenes, and vice versa. High-quality plots at every dependency layer for "Bus" and "LOTR" have been made available on-line [8].

In order to assess the proposed method from the quality consistency point a view, a time-local version of the PSNR standard deviation

Table 1. Average results achieved by both the RCA in [3] and the proposed VBR controller. Incremental results are given with respect to constant QP encoding.

d	Algorithm	$\Delta\mu_{PSNR}$ (dB)	$\Delta\bar{\sigma}_{PSNR,j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
0	[3]	-0.19	0.41	1.87	8/0	57.42
	Proposed	-0.07	0.12	0.94	0/0	49.42
1	[3]	-0.43	0.75	1.35	9/0	57.29
	Proposed	-0.08	0.13	1.30	0/0	57.20
2	[3]	-0.33	0.35	0.68	6/0	54.91
	Proposed	-0.03	0.06	0.55	0/0	50.28
3	[3]	-0.20	0.36	0.44	0/0	52.81
	Proposed	-0.03	0.05	0.64	0/0	49.84
4	[3]	-0.46	0.51	0.30	0/0	53.45
	Proposed	-0.03	0.05	0.55	0/0	54.77

was computed. This aims to measure the quality consistency within a scene, so reducing the impact of the scene changes on the PSNR standard deviation. Specifically, the local PSNR standard deviation was computed over a time-window as follows:

$$\sigma_{PSNR,j} = \sqrt{\frac{1}{W} \sum_{i=j-W/2}^{j+W/2-1} (PSNR_i - \mu_{PSNR,W})^2}, \quad (8)$$

where W denotes the time-window size (in number of pictures), and $\mu_{PSNR,W}$ the average PSNR for a given window size. W was set to $2^{T^{(d)}}$ pictures in our experiments, which is a time interval short enough to minimize the influence of PSNR leaps at the scene changes. Finally, in order to summarize the results in an unique measurement, the mean value of this local PSNR standard deviation, denoted as $\bar{\sigma}_{PSNR,j}$, was computed.

The results in terms of $\bar{\sigma}_{PSNR,j}$ increment with respect to CQP encoding, $\Delta\bar{\sigma}_{PSNR,j}$, are shown in Table 1. As can be observed, our proposal achieved better quality consistency than that of the RCA in [3]. It should be noted that the comparison to CQP in these terms is not fair, since in this case large instantaneous bit rate variations are allowed to reduce the PSNR leaps, especially at the scene changes. Finally, the proposed VBR controller was also assessed in terms of target bit rate adjustment and mean buffer level by computing the output bit rate error, the number of pictures in which either an overflow (#O) or an underflow (#U) occurred, and the mean buffer level, μ_V . As it can be seen in Tables 1–3, the RCA in [3] and the proposed algorithm provided in most cases output bit rate differences below 2%, that is the maximum bit rate error recommended in [7] for spatial/CGS testing scenario. The results in terms of μ_V achieved by our proposal were closer to the target buffer fullness, thus proving a good long-term adaptation to the target bit rate at each dependency layer. Furthermore, the results in terms of #O and #U revealed that the proposed VBR controller was able to significantly reduce both the overflow and underflow risks.

5. CONCLUSIONS AND FURTHER WORK

In this paper a novel VBR controller for real-time H.264/SVC video coding applications has been proposed. The VBR controller aims to improve the quality consistency by preventing unnecessary QP fluctuations. For this purpose, a novel method for estimating a proper QP increment with respect to that of the previous picture has

Table 2. Performance comparison between the RCA in [3] and the proposed VBR controller for "Bus". The results achieved by constant QP encoding have also been included for reference. The experiments have been conducted using the following target bit rates: 66.21 (d = 0), 93.61 (d = 1), 199.56 (d = 2), 364.22 (d = 3) and 480.60 kbps (d = 4).

d	Algorithm	μ_{PSNR} (dB)	$\bar{\sigma}_{\text{PSNR},j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
0	CQP	30.53	0.28	-	0/0	52.73
	[3]	30.57	0.55	0.47	0/0	51.96
	Proposed	30.50	0.29	-0.41	0/0	47.94
1	CQP	30.43	0.25	-	0/0	53.52
	[3]	30.32	0.77	0.38	0/0	53.33
	Proposed	30.41	0.27	-0.39	0/0	49.01
2	CQP	26.89	0.16	-	0/0	52.16
	[3]	26.82	0.46	0.19	0/0	52.08
	Proposed	26.85	0.18	-0.01	0/0	49.14
3	CQP	29.31	0.18	-	0/0	52.35
	[3]	29.20	0.46	0.23	0/0	52.06
	Proposed	29.30	0.19	-0.38	0/0	47.80
4	CQP	29.38	0.17	-	0/0	52.93
	[3]	29.17	0.50	0.18	0/0	52.67
	Proposed	29.36	0.18	0.15	0/0	54.05

Table 3. Performance comparison between the RCA in [3] and the proposed VBR controller for "LOTR". The results achieved by constant QP encoding have also been included for reference. The experiments have been conducted using the following target bit rates: 66.47 (d = 0), 97.32 (d = 1), 189.47 (d = 2), 388.07 (d = 3) and 500.56 kbps (d = 4).

d	Algorithm	μ_{PSNR} (dB)	$\bar{\sigma}_{\text{PSNR},j}$ (dB)	Bit Rate Error (%)	#O/#U	μ_V (%)
0	CQP	34.45	0.66	-	42/48	49.76
	[3]	33.14	1.10	3.82	55/0	78.04
	Proposed	34.65	0.85	1.38	0/0	50.86
1	CQP	34.39	0.67	-	100/107	46.90
	[3]	33.19	2.05	1.72	66/0	69.65
	Proposed	34.63	0.91	1.74	0/0	53.87
2	CQP	32.88	0.91	-	96/111	47.15
	[3]	32.26	1.51	0.30	40/0	63.69
	Proposed	33.04	1.04	0.94	0/0	49.83
3	CQP	35.24	0.82	-	92/114	45.22
	[3]	35.43	1.31	1.26	0/0	52.99
	Proposed	35.45	0.95	0.83	0/0	50.52
4	CQP	35.14	0.82	-	205/237	45.58
	[3]	34.86	1.57	1.00	0/0	53.82
	Proposed	35.36	0.97	1.53	0/0	58.42

been developed. In particular the QP increment estimation at each dependency layer is computed by means of an RBF network that is especially designed for this purpose. Furthermore, the input vector to the RBF network is enlarged with two additional parameters to provide an effective solution for a wide range of both target buffer fullness and buffer size.

A mobile live streaming scenario was simulated to assess the performance of the VBR controller, which was compared to both CQP encoding, as a reference for nearly constant quality, and a recently

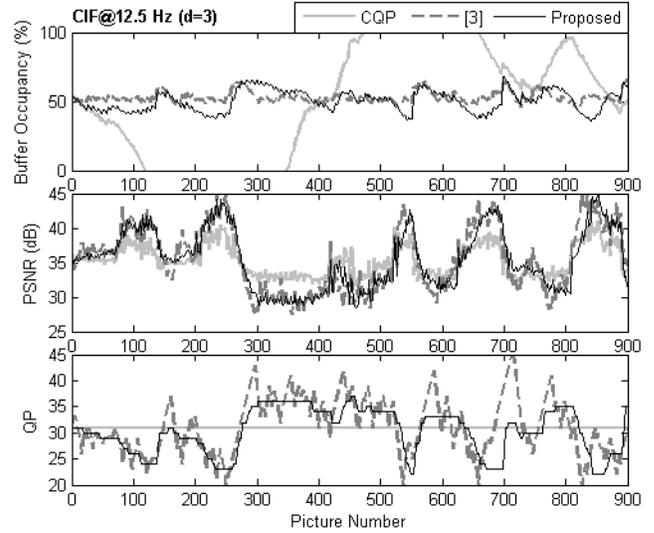


Fig. 3. Encoder buffer occupancy, PSNR and QP evolutions at layer $d=3$ for "LOTR".

proposed CBR controller for H.264/SVC [3]. The proposed RBF-based VBR controller achieved an excellent performance in terms of average quality, quality consistency, long-term adjustment to the target bit rate, and buffer overflow and underflow prevention, at each spatial/CGS layer.

As future work, an approximate and fast implementation of the input-output relation defined by the RBF is being developed. Furthermore, we plan to extend the VBR controller to MGS.

6. REFERENCES

- [1] R. Schaefer, H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand, "MCTF and scalability extension of H.264/AVC and its application to video transmission, storage, and surveillance," in *Proceedings of VCIP 2005, Peking, China*, July 2005.
- [2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 9, pp. 1103–1120, Sept. 2007.
- [3] Y. Liu, Z. G. Li, and Y. C. Soh, "Rate control of H.264/AVC scalable extension," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 1, pp. 116–121, Jan. 2008.
- [4] A. Leontaris and A. M. Tourapis, "Rate control for the Joint Scalable Video Model (JSVM)," *Video Team of ISO/IEC MPEG and ITU-T VCEG, JVT-W043, San Jose, California*, April 2007.
- [5] L. Xu, W. Gao, X. Ji, D. Zhao, and S. Ma, "Rate control for spatial scalable coding in SVC," in *Picture Coding Symposium, 2007. PCS 2007*, Nov. 2007.
- [6] J. Vieron, M. Wien, and H. Schwarz, "JSVM 11 software," *24th Meeting: Geneva, Doc. JVT-X203*, July 2007.
- [7] M. Wien and H. Schwarz, "Testing conditions for SVC coding efficiency and JSVM performance evaluation," *JVT-Q205, 16th JVT Meeting, Poznan, Poland*, July 2005.
- [8] [Online], "<http://www.tsc.uc3m.es/~sescalona/PCS2010/>," .
- [9] G. Bjøntegaard, "Calculation of average PSNR differences between RD curves," *VCEG contribution, VCEG-M33, Austin*, April 2001.