



**HAL**  
open science

# Dependencies consideration for Gobal Rate-Distortion Optimization : application to HEVC

Maxime Bichon

► **To cite this version:**

Maxime Bichon. Dependencies consideration for Gobal Rate-Distortion Optimization : application to HEVC. Signal and Image processing. INSA de Rennes, 2019. English. NNT : 2019ISAR0030 . tel-04426256

**HAL Id: tel-04426256**

**<https://theses.hal.science/tel-04426256>**

Submitted on 30 Jan 2024

**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.

# Thèse de doctorat de

l'Institut National des Sciences  
Appliquées Rennes  
Comue Université Bretagne Loire  
École Doctorale n° 601  
Mathématiques et Sciences et Technologies  
de l'Information et de la Communication  
Spécialité : Signal, Image, Vision

Par

« **Maxime BICHON** »

## « Dependencies consideration for Global Rate-Distortion Optimization: application to HEVC »

Thèse présentée et soutenue à « l'INSA de Rennes », le « 01 Mars 2019 »

Unité de recherche : IETR

Thèse N° : 19ISAR 03 / D19 - 03

### Rapporteurs avant soutenance :

Marco	CAGNAZZO	Professeur à Télécom-Paris Tech	(France)
Mathias	WIEN	Maître de conférences HDR à l'université d'Aachen	(Allemagne)

### Composition du Jury :

*Attention, en cas d'absence d'un des membres du Jury le jour de la soutenance, la composition du jury doit être revue pour s'assurer qu'elle est conforme et devra être répercutée sur la couverture de thèse*

Président :	Christine	GUILLEMOT	Directrice de recherche INRIA à Rennes	(France)
Examineurs :	Frédéric	DUFAUX	Directeur de recherche CNRS à Centrale Supélec Paris	(France)
	Ce	ZHU	Professeur à l'UESTC	(Chine)
	Marco	CAGNAZZO	Professeur à Télécom-Paris Tech	(France)
	Mathias	WIEN	Maître de conférences HDR à l'université d'Aachen	(Allemagne)
Encadrants :	Julien	LE TANOU	Ingénieur Senior à Ericsson Media Solutions	(France)
	Wassim	HAMIDOUCHE	Maître de conférences à l'INSA de Rennes	(France)
Dir. de thèse	Luce	MORIN	Professeur à l'INSA de Rennes	(France)

### Invité(s)

Michael	ROPERT	Ingénieur Principal à Ericsson Media Solutions	(France)
---------	--------	--	----------



Intitulé de la thèse :

Considération des dépendances pour l'optimisation débit-distorsion globale : application à HEVC

-

Dependencies consideration for Global Rate-Distortion Optimization : application to HEVC

**Maxime BICHON**

En partenariat avec :



*Document protégé par les droits d'auteur*



## Acknowledgements

First of all, I wish to express my sincere thanks to my supervisor Julien Le Tanou for giving me the opportunity to start my career at Ericsson/Mediakind. It is the rigor, precision and tenacity he taught me to apply in my work that has made this thesis possible. I am also deeply grateful to Michael Ropert for his help in the theoretical aspects of my work and his priceless sharing of knowledge. I would like to thank my colleagues of Ericsson/Mediakind, in particular Anthony Arch, Médéric Bestel and Jean Kypreos, for the uncountable times they helped me with inspiring discussions and their support.

I would also like to thank my academic supervisors Wassim Hamidouche and Luce Morin for the effort they put into this thesis and their continuous encouragements. I am very thankful to all my colleagues of the Vaader team at IETR laboratory who made working there so enjoyable.

Finally, I would like to thank my family and friends for supporting me during all stages of my education. Last but not least, I would like to thank all my teachers of the digital imaging specialization at *Ecole Supérieure d'Ingénieurs de Rennes* for giving me the desire to work in the field of image processing and encouraging me to complete my education with a PhD.



# Table of contents

<b>List of figures</b>	<b>ix</b>
<b>List of tables</b>	<b>xiii</b>
<b>Acronyms</b>	<b>xv</b>
<b>Mathematical notations</b>	<b>xix</b>
<b>Résumé en Français</b>	<b>1</b>
<b>Introduction</b>	<b>9</b>
<b>1 Source Coding</b>	<b>15</b>
1.1 Rate-Distortion Theory . . . . .	15
1.1.1 The Distortion . . . . .	16
1.1.2 The Rate . . . . .	17
1.1.3 The Rate-Distortion Function . . . . .	18
1.1.4 The Shannon Lower Bound . . . . .	20
1.1.5 Rate-Distortion Optimization . . . . .	21
1.1.6 Link with the central limit theorem . . . . .	23
1.2 Quantization . . . . .	24
1.2.1 Scalar Quantization Description . . . . .	24
1.2.2 Quantization Control for Rate-Distortion Optimization . . . . .	26
1.2.3 High-Rate R-D approximation . . . . .	27
1.3 Predictive Coding . . . . .	29
1.3.1 Principle of a Predictive Encoder . . . . .	29
1.3.2 Prediction in the RDO context . . . . .	31
1.3.3 Optimization of Predictive Coding . . . . .	32
1.4 Dependencies related to Source Coding . . . . .	33



1.5	Global Optimization in Source Coding . . . . .	36
1.5.1	Joint Optimization versus Independent Optimization . . . . .	36
1.5.2	Complexity formalization . . . . .	36
<b>2</b>	<b>The hybrid video coding standard: H.265/HEVC</b>	<b>39</b>
2.1	Frames Sequencing . . . . .	41
2.1.1	Slices Types . . . . .	41
2.1.2	Common Coding Structures . . . . .	42
2.2	QuadTree Partitioning . . . . .	44
2.2.1	Coding Tree Units and Coding Units . . . . .	44
2.2.2	Prediction Units . . . . .	45
2.3	Intra Coding . . . . .	45
2.4	Inter Coding . . . . .	47
2.4.1	Motion Estimation/Compensation . . . . .	47
2.4.2	Motion Vector Prediction . . . . .	47
2.4.3	Merge and Skip modes . . . . .	48
2.5	Transform and Quantization . . . . .	49
2.6	Entropy Coding . . . . .	50
2.7	Common Test Conditions . . . . .	51
2.8	Dependencies related to Hybrid Video Coding . . . . .	52
2.8.1	Dependency related to samples prediction . . . . .	52
2.8.2	Dependency related to side informations prediction . . . . .	53
2.8.3	Dependency related to cross-processing . . . . .	54
2.8.4	Dependency related to Entropic Coding . . . . .	55
<b>3</b>	<b>Coding dependencies in Intra coding</b>	<b>57</b>
3.1	Previous Methods . . . . .	58
3.1.1	Minimization of reference samples distortion . . . . .	59
3.1.2	Distortion propagation modeling . . . . .	63
3.2	Inter-Block Dependencies Consideration for Intra Coding Optimization . . . . .	65
3.2.1	Proposed JRDO models . . . . .	66
3.2.2	Experiments . . . . .	71
3.3	Low Complexity JRDO of Prediction Units Couples for HEVC Intra Coding . . . . .	74
3.3.1	Acceleration Methods . . . . .	75
3.3.2	Experiments and Results . . . . .	79
3.4	Dual-JRDO for quantizer parameter estimation . . . . .	83
3.4.1	Reference <a href="#">Adaptive Quantization (AQ)</a> method . . . . .	84

3.4.2	<i>Dual-JRDO AQ</i> results . . . . .	87
3.5	Conclusion . . . . .	87
<b>4</b>	<b>Coding dependencies in Inter coding</b>	<b>91</b>
4.1	Previous Methods . . . . .	92
4.1.1	Simplified propagation methods . . . . .	93
4.1.2	Dependent propagation methods . . . . .	99
4.1.3	Exhaustive joint search methods and empirical models . . . . .	101
4.1.4	State-of-the-art conclusion . . . . .	103
4.2	Temporal Distortion Propagation Model . . . . .	103
4.3	RDSTQ Algorithm . . . . .	106
4.3.1	Local Quantization Problem . . . . .	107
4.3.2	Analytical Solution . . . . .	107
4.4	Hypotheses Validation . . . . .	111
4.5	Performance Evaluation . . . . .	118
4.6	QuadTree Consideration for RDSTQ . . . . .	127
4.6.1	Probability of occurrence for non-uniform partitioning . . . . .	128
4.6.2	Temporal distortion propagation model with non-uniform partitioning	131
4.6.3	Comparative analysis with initial model . . . . .	132
4.7	Conclusion . . . . .	134
	<b>Conclusion</b>	<b>137</b>
	<b>Author Publications</b>	<b>141</b>
	<b>Appendix A Derivation of the Rate-Quantization function</b>	<b>143</b>
A.1	Derivation of (1.23) . . . . .	143
	<b>Appendix B Rate Distortion Spatio-Temporal Quantization (RDSTQ) equations</b>	<b>145</b>
B.1	Computing $D_{Tot}$ . . . . .	145
B.2	Accumulation factor in recursive form . . . . .	146
B.3	Computing the Lagrangian . . . . .	147
	<b>Appendix C QuadTree consideration in RDSTQ</b>	<b>149</b>
C.1	Derivation of optimal quantizer considering QuadTree partitioning . . . . .	149
	<b>References</b>	<b>151</b>



# List of figures

1	Périmètres de la thèse et d'une norme de compression . . . . .	3
2	Historique des normes de codage vidéo (Figure modifiée issue de [LTB18])	4
3	Scope of the thesis and scope of a compression standard . . . . .	11
4	History of Video Coding Standards (Modified Figure from [LTB18]) . . . .	12
1.1	Operational <b>Rate-Distortion (R-D)</b> function and the convex hull of operational <b>R-D</b> points as the modeled <b>R-D</b> function. . . . .	18
1.2	Visual representation of the <b>Bjøntegaard-Delta Bit-Rate (BD-BR)</b> metric to compare two coding systems based on their respective <b>Rate-Quality (RQ)</b> curves. . . . .	19
1.3	Shannon bound for uniform, laplacian and gaussian distributions with $\sigma_i^2 = 1$ .	21
1.4	Input and Output of Scalar Quantization for Uniform Quantizer (a) and Uniform Quantizer with Dead-zone (b) . . . . .	25
1.5	High-Rate approximation and <i>Xu et al.</i> [XJGZ07] model of the Distortion-Quantization relationship, with distortion expressed in <b>Mean Square Error (MSE)</b> , against the maximum achievable distortion $\sigma_i^2$ . . . . .	28
1.6	Basic Encoder structure for predictive coding . . . . .	30
1.7	Basic Decoder structure for predictive coding . . . . .	30
1.8	Closed-loop encoder design for lossy predictive coding . . . . .	31
1.9	Illustration of the syntax bitrate and the residue energy relationship . . . . .	34
2.1	Block-based Hybrid Coding Scheme . . . . .	40
2.2	All-Intra Configuration . . . . .	43
2.3	Random Access Configuration . . . . .	43
2.4	Low Delay Configuration . . . . .	44
2.5	Illustration of the recursive QuadTree Partitioning in <b>High Efficiency Video Coding (HEVC)</b> . . . . .	45
2.6	Available Prediction Unit partitions in <b>HEVC</b> . . . . .	46
2.7	Intra Prediction in HEVC . . . . .	46

2.8	Motion Compensation Illustration . . . . .	48
2.9	Framework of <b>Context-Adaptive Binary Arithmetic Coding (CABAC)</b> . ❶: <b>Syntax Elements (SEs)</b> may be binary before CABAC. ❷: SEs are coded in Bypass mode or Context mode. ❸: Bin's probability is transferred for arithmetic coding. ❹: Context is updated for probability estimation. . . . .	51
3.1	Spatial distribution of the prediction error $\varepsilon = (x - \hat{x})^2$ with $x$ the source sample and $\hat{x}$ the predicted sample, using diagonal-left predictor (mode number 18) . . . . .	59
3.2	Illustration of Weighted Cross Prediction . . . . .	61
3.3	Joint Line and Pixel Prediction: Illustration of the two groups pixels . . . . .	62
3.4	QuadTree possible partitioning and related spatial dependencies illustration . . . . .	67
3.5	Example of <i>Dual-JRDO</i> and <i>Quad-JRDO</i> . . . . .	68
3.6	Proposed coding scheme for optimizing jointly a <b>Prediction Unit (PU)</b> with its right neighbor. . . . .	69
3.7	Proposed coding scheme for optimizing jointly a <b>PU</b> with its bottom neighbor. . . . .	70
3.8	Distortions according to possible reconstruction levels when varying the <b>Quantization Parameter (QP)</b> . . . . .	85
4.1	Illustration of the <b>Quantization Parameter Cascading (QPC)</b> for hierarchical layers. . . . .	94
4.2	An example of 3 reference frames search with 3 different $\lambda$ values for Inter-coded frames as presented by <i>Im and Chan</i> in [IC15]. . . . .	102
4.3	Illustration of the surface ratio resulting from the motion compensation. . . . .	104
4.4	R-D curves of non-Intra frames according to Intra <b>QP</b> offsets with (a) <i>Kimono</i> and (b) <i>Cactus</i> sequences . . . . .	113
4.5	Inter Probability $p_i$ , according to cost ratio $r_i$ estimated by (4.57) for the Initial function and (4.59) for the proposed function. . . . .	114
4.6	$U$ value evolution within a <b>Group of Pictures (GOP)</b> of 32 frames for different values of $p$ , with $\Psi_i = 1 \forall i, t$ . . . . .	117
4.7	Rate distribution of first <b>GOP</b> frames with sequence <i>RaceHorses</i> at <b>QP</b> = 32 for (a) <b>Rate Distortion Temporal Quantization (RDTQ)</b> with Initial Probability and <b>RDTQ</b> with Proposed Probability; for (b) Proposed Probability with and without Skip consideration. . . . .	122
4.8	<b>QP</b> distribution over the frame 128 of <i>BQTerrace</i> sequence (a) without psycho-visual function and (b) with psycho-visual enabled. (c) The source frame . . . . .	125

---

4.9	Illustration of the chosen spatial <b>Coding Unit (CU)</b> indexing for the first <b>Coding Tree Unit (CTU)</b> of a frame. . . . .	129
4.10	Example of occurrence probability . . . . .	130



# List of tables

2.1	I-Frame period based on Framerate as recommended by the <a href="#">Common Test Conditions (CTC)</a> proposed by Bossen in <a href="#">[Bos13]</a> . . . . .	52
3.1	Virtual YUV-BD-BR (Peak Signal to Noise Ratio (PSNR)) of using source as prediction reference against reconstructed in <a href="#">HEVC Test Model (HM)16.12</a> . . . . .	58
3.2	Average Y-BD-BR (PSNR) depending of which neighboring PU is jointly optimized in <a href="#">HM16.12 Dual-JRDO</a> . . . . .	70
3.3	Y BD-Rate of <a href="#">Dual-JRDO</a> in <a href="#">Joint Model (AVC Reference Software) (JM)19.0</a> and <a href="#">HM16.6</a> . . . . .	73
3.4	Y BD-Rate of <a href="#">Quad-JRDO</a> in <a href="#">JM19.0</a> and <a href="#">HM16.6</a> . . . . .	74
3.5	The <a href="#">Dual-JRDO</a> gain against <a href="#">Independent Rate-Distortion Optimization (Independent-RDO)</a> for each depth . . . . .	75
3.6	Configurations . . . . .	79
3.7	<a href="#">Dual-JRDO</a> coding efficiency over <a href="#">Independent-RDO</a> in <a href="#">HM16.12</a> . . . . .	81
3.8	<a href="#">Dual-JRDO</a> complexity increase over <a href="#">Independent-RDO</a> in <a href="#">HM16.12</a> . . . . .	82
3.9	<a href="#">AQ</a> method coding efficiency with or without <a href="#">QP</a> offset overhead over no <a href="#">AQ</a> in <a href="#">HM16.12</a> . . . . .	86
3.10	<a href="#">Dual-JRDO AQ</a> method coding efficiency in <a href="#">HM16.12</a> . . . . .	88
4.1	Results of methods considering the Inter-layer frames distortion propagation . . . . .	97
4.2	Theoretical number of frames required for $U$ convergence based on $p$ values and related $U_{max}$ . . . . .	117
4.3	Coding efficiency over no local quantization in <a href="#">HM</a> for <a href="#">RDTQ</a> . . . . .	119
4.4	Coding efficiency over no local quantization in <a href="#">HM</a> for <a href="#">RDSTQ</a> . . . . .	120
4.5	Coding efficiency over no local quantization in $x265$ for <a href="#">RDTQ</a> . . . . .	121
4.6	Coding efficiency over no local quantization in $x265$ for <a href="#">RDSTQ</a> . . . . .	123
4.7	Coding efficiency over no local quantization in <a href="#">HM</a> . . . . .	126
4.8	Comparison of encoding runtime (in seconds) with no- <a href="#">AQ</a> and <a href="#">RDSTQ</a> in <a href="#">HM</a> . . . . .	127





# Acronyms

<i>HM</i>	HEVC Test Model
<i>Independent-RDO</i>	Independent Rate-Distortion Optimization
<i>JM</i>	Joint Model (AVC Reference Software)
<i>iid</i>	<i>independent and identically distributed</i>
AI	All-Intra
AMVP	Advanced Motion Vector Prediction
AOM	Alliance for Open Media
AQ	Adaptive Quantization
AVC	Advanced Video Coding
AVS	Audio-Video coding Standard
BD-BR	Bjøntegaard-Delta Bit-Rate
CABAC	Context-Adaptive Binary Arithmetic Coding
CAVLC	Context-Adaptive Variable-Length Coding
CBR	Constant Bitrate
CDA	Competition Decision Algorithm
CTC	Common Test Conditions
CTU	Coding Tree Unit
CU	Coding Unit
DCT	Discrete Cosine Transform
DFD	Displaced Frame Difference
DPB	Decoded Picture Buffer
DST	Discrete Sine Transform
DVB	Digital Video Broadcasting

DWP	Distance-based Weighting Prediction
GOP	Group of Pictures
HD	High Definition
HDR	High Dynamic Range
HEVC	High Efficiency Video Coding
HVS	Human Visual System
IBDM	Inter-Block Dependency Model
IDR	Instantaneous Decoder Refresh
IEC	International Electrotechnical Commission
IPTV	Internet Protocol Television
ISO	International Organization for Standardization
ITU	International Telecommunication Union
JCT-VC	Joint Collaborative Team on Video Coding
JLPP	Joint Line and Pixel Prediction
JRDO	Joint Rate-Distortion Optimization
JVET	Joint Video Exploration Team
JVT	Joint Video Team
LD	Low-Delay
LDB	Low-Delay B
LDP	Low-Delay P
MB	MacroBlock
MCP	Motion Compensated Prediction
MDCS	Mode Dependent Coding Scan
ME	Motion Estimation
MLM	Multiple Lagrangian Multiplier
MPEG	Moving Picture Experts Group
MPM	Most Probable Mode
MQP	Multi Quantization Parameter
MSE	Mean Square Error
MV	Motion Vector

---

MVD	Motion Vector Difference
OTT	Over-The-Top
PDF	Probability Density Function
POC	Picture Order Count
PSNR	Peak Signal to Noise Ratio
PU	Prediction Unit
QoE	Quality of Experience
QP	Quantization Parameter
QPC	Quantization Parameter Cascading
R-D	Rate-Distortion
RA	Random-Access
RC	Rate Control
RDO	Rate Distortion Optimization
RDSTQ	Rate Distortion Spatio-Temporal Quantization
RDTQ	Rate Distortion Temporal Quantization
RMD	Rough Mode Decision
RQ	Rate-Quality
RQT	Residual QuadTree
SAO	Sample Adaptive Offset
SATD	Hadamard Transformed Sum of Absolute Difference
SD	Standard Definition
SDTP	Source Distortion Temporal Propagation
SE	Syntax Element
SSIM	Structural Similarity Index
SVM	Support Vector Machine
TBD	Target Bitrate Deviation
TU	Transform Unit
UHD	Ultra High Definition

VBR	Variable Bitrate
VCEG	Video Coding Experts Group
VLC	Variable-Length Coding
VQM	Visual Quality Metric
VVC	Versatile Video Coding
WCG	Wide Color Gamut
WCP	Weighted Cross Prediction

# Mathematical notations

Symbol	Description	Type
$X_i$	$X$ sample or group of samples	Random variable
$X$	Source signal (before coding)	Vector of random variables
$Y_i$	$Y$ sample or group of samples	Random variable
$Y$	Reconstructed signal (after coding)	Vector of random variables
$\mathbb{E}$	Expected value of a random variable	Function
$\mathbb{P}$	Probability of a discrete variable	Discrete probability
$\vec{p}$	Coding parameters	Vector of scalars
$f_{X_i}(x)$	Probability Density Function of $X_i$	Probability Density Function



## AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

**Titre de la thèse:**

Considération des dépendances pour l'optimisation débit-distortion globale : application à HEVC

**Nom Prénom de l'auteur : BICHON MAXIME**

**Membres du jury :**

- Monsieur ZHU Ce
- Monsieur LE TANOU Julien
- Madame GUILLEMOT Christine
- Monsieur WIEN Mathias
- Madame MORIN Luce
- Monsieur HAMIDOUCHE Wassim
- Monsieur CAGNAZZO Marco
- Monsieur DUFAUX Frédéric

Président du jury : *Christine GUILLEMOT*

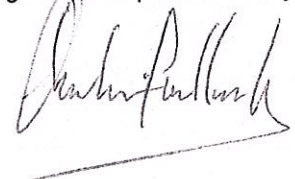
Date de la soutenance : 01 Mars 2019

Reproduction de la these soutenue

- Thèse pouvant être reproduite en l'état  
 Thèse pouvant être reproduite après corrections suggérées

Fait à Rennes, le 01 Mars 2019

Signature du président de jury



Le Directeur,

M'hamed DRISSI







# Résumé en Français

## Préambule

Dans le domaine de la vidéo, le besoin d'évolutions technologiques est une constante intemporelle. Auparavant, les contenus vidéo étaient principalement diffusés dans les cinémas ou bien sur les téléviseurs à domicile. De nos jours, les vidéos sont disponibles et visionnées sur une grande variété d'appareils, via IP ou réseaux mobiles. Selon une étude récente de la société Cisco [Cis18], les données vidéos occuperont 82% du trafic IP mondial d'ici 2022, contre 75% en 2017. Dans un précédent rapport [Cis14], l'entreprise américaine prévoyait une occupation de la vidéo à 79% pour 2018, ce qui montre la fiabilité de ces estimations. Dans son dernier rapport [Eri18], la société Ericsson annonce que la quantité de contenu vidéo circulant via les réseaux mobiles devrait augmenter de 35% par an au cours des 6 prochaines années.

Ces chiffres peuvent s'expliquer par une évolution des usages et habitudes utilisateurs. L'importante consommation de contenu vidéo est stimulée par le développement du modèle économique *TV Everywhere*. L'accès au contenu vidéo, via Internet et les applications mobiles, devient uniquement conditionné par un processus d'authentification et ne dépend aucunement de l'appareil physique utilisé pour visionner la vidéo. Le concept est adopté par de célèbres fournisseurs de contenu tels que Netflix, Youtube ou Amazon. La popularité croissante du sport électronique (*e-sport*) contribue également à la création massive de contenus vidéo par le biais de joueurs offrant la possibilité de visionner en live leurs parties. La plateforme Twitch, qui diffuse principalement des retransmissions en direct de jeux vidéo, signale qu'en juillet 2018 [Twi], 1 million de téléspectateurs en moyenne étaient connectés sur leur plateforme en permanence. Les jeunes générations (12-16 ans) sont des fournisseurs et des consommateurs très actifs de vidéos par le biais de réseaux sociaux comme Facebook, Twitter, Snapchat et plus récemment TikTok. Cet écosystème très dynamique explique les changements drastiques dans la façon de consommer les vidéos ainsi que la croissance du trafic vidéo mondial.

Parallèlement à l'importante diversité des contenus et au nombre de solutions d'accessibilité, l'augmentation du volume de données vidéos s'explique aussi par les exigences croissantes de qualité et d'immersion des utilisateurs. Les écrans sont passés d'une définition standard (SD: Standard Definition) à une haute définition (HD: High Definition) et une vaste majorité des téléviseurs actuels supportent une ultra-haute définition (UHD: Ultra-High Definition). De plus, nous observons un déploiement massif de contenus UHD, notamment via des fournisseurs de contenu tel que Netflix. Le développement des fabricants de téléviseurs a également autorisé une plus grande fluidité dans l'affichage des contenus (HFR: High Frame Rate). D'autres évolutions technologiques telles que la 3D, l'imagerie à haute dynamique (HDR: High Dynamic Range) et à gamme de couleurs étendues (WCG: Wide Color Gamut) sont également déployées afin d'améliorer la qualité d'expérience des utilisateurs finaux. Malgré les avantages évidents de ces progrès techniques, chacun implique une croissance physique de la quantité de données à afficher, stocker et transporter.

En résumé, les données vidéo sont de plus en plus présentes dans la vie quotidienne des utilisateurs et la taille numérique de ces données augmente au fil des nouvelles technologies. La compression vidéo optimale, qui consiste à réduire la quantité de données vidéos en minimisant l'impact sur la qualité visuelle, est l'objectif principale de cette thèse. Bien que la capacité des réseaux et des systèmes de stockage soit nettement plus élevée que par le passé, la conclusion est simple. La compression vidéo n'est pas seulement pertinente : La compression vidéo est essentielle.

## Contexte

La compression vidéo est un sujet soumis à un environnement très dynamique auquel on peut difficilement se soustraire. Les normes de compression ouvertes imposent certaines contraintes et touchent un très large nombre d'entreprises, allant des fabricants d'encodeurs ou décodeurs aux fournisseurs de contenu. En conséquence, il est important de connaître à tout moment l'actualité de ce contexte. Dans le reste de ce document, les termes de codage et de compression sont utilisés comme synonymes.

## Norme de Compression Vidéo

Une norme de compression spécifie la syntaxe du flux compressé et le processus de décodage afin d'obtenir une vidéo à afficher sur un écran. L'encodage consiste à créer un flux binaire compatible avec une norme et cherche à optimiser l'efficacité de compression. Le processus de codage/décodage est représenté sur la Fig. 1, précisant le périmètre défini par la norme.

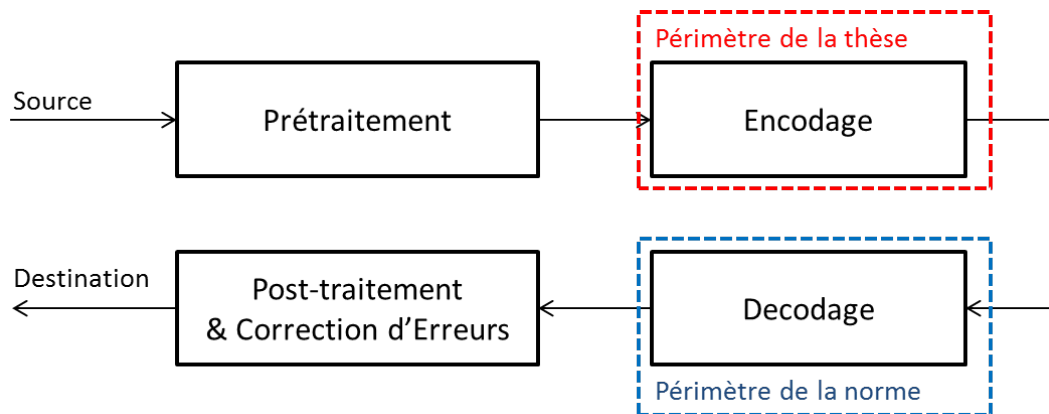


Fig. 1 Périmètres de la thèse et d'une norme de compression

Actuellement, deux axes de développement pour les futures technologies de compression vidéo sont en concurrence. Les organismes de normalisation classiques composés de l'ISO/IEC (International Organization for Standardization/ International Electrotechnical Commission) et de l'ITU (International Telecommunication Union) élaborent des normes de codage vidéo depuis plus de trois décennies. Récemment, un consortium d'entreprises appelé AOM (Alliance for Open Media) s'est révélé être un concurrent à ces organismes.

### Organismes de Normalisation

VCEG (Video Coding Experts Group) est un groupe de travail de l'ITU responsable du développement des normes de codage vidéo H.120, H.261 et H.263+, entre autres. MPEG (Moving Picture Experts Group) est un autre groupe de travail formé par l'ISO et l'IEC pour la normalisation du codage et de la transmission audio et vidéo. MPEG est responsable de la norme MPEG-1 publiée en 1993 et de la norme MPEG-4 publiée en 1998. VCEG et MPEG ont uni leurs efforts dans le codage vidéo pour la première fois lors du développement de la norme H.262/MPEG-2 partie 2 en 1996.

Les différents partenariats qui ont suivi entre ces deux groupes ont été présentés sous des noms de projets tels que JVT (Joint Video Team) pour AVC [ITa, ISOa] (Advanced Video Coding) en 2003 et JCT-VC (Joint Collaborative Team on Video Coding) pour HEVC [ITb, ISOb] (High Efficiency Video Coding) en 2013. Alors que AVC est actuellement la norme de codage vidéo la plus utilisée dans le monde, HEVC permet d'importantes économies de débit pour une qualité équivalente et suscite un intérêt croissant. JVET (Joint Video Exploration Team), le nom de l'actuelle collaboration entre VCEG et MPEG, a annoncé en avril 2018 le développement du future standard VVC (Versatile Video Coding) qui se terminera en 2020. Les nouvelles solutions de codage étudiées [CAS<sup>+</sup>17] permettront de dépasser les capacités d'HEVC.

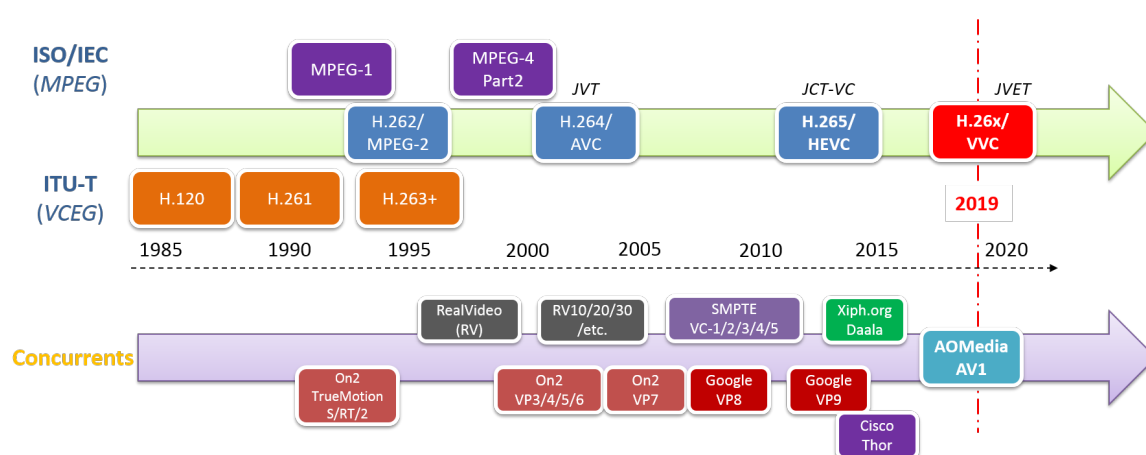


Fig. 2 Historique des normes de codage vidéo (Figure modifiée issue de [LTB18])

## Alliance for Open Media

AOM est un consortium industriel visant à fournir une alternative libre de droits aux normes de compression vidéo présentées ci-dessus. Plusieurs codec propriétaires destinées à s'imposer sur le marché ont été initiés au cours des années 90 avec des sociétés telles que On2 (codecs VP3 à VP7) et RealNetworks (codec RealVideo). Une autre revendication à l'initiative d'AOM est le désir de produire de nouvelles normes plus rapidement et d'accélérer le processus d'innovation, jugé trop lent concernant les organismes de normalisation.

AOM a annoncé en mars 2018 la sortie d'AV1 version 1 [fOM], un codec vidéo libre de droits avec des performances de compression vidéo améliorées par rapport à l'état de l'art. Cette norme émergente a pour ambition de concurrencer la norme HEVC en proposant un système de compression vidéo efficace, sans aucune limitation quant aux coûts élevés ou aux conditions complexes d'utilisation légales. Une chronologie résumée des normes de compression vidéo est proposée dans la Figure 2.

## Conception d'un encodeur

Une norme est conçue pour optimiser l'efficacité maximale de compression en mettant à disposition un ensemble optimisé d'outils de codage. Cependant, il n'y a aucune garantie qu'un encodeur exploite correctement ces outils de codage et soit capable de fournir la meilleure qualité possible. Dans l'industrie, les fournisseurs d'encodeurs sont généralement jugés ou mis en concurrence sur la base de leur capacité à optimiser ces processus d'encodage grâce à leurs algorithmes propriétaires, tout en se conformant au même processus de décodage.

En conséquence, le monde du codage vidéo observe une compétition de standards mais aussi une compétition d'algorithmes d'encodage. Ces algorithmes sont soumis à plusieurs contraintes liées au cas d'application, comme le cas de la diffusion vidéo numérique qui présente une contrainte de débit maximum fixée pour une diffusion en continu. Les contraintes en termes de complexité de calcul et de délai sont également des composantes nécessaires pour la conception d'encodeurs.

## Motivations

Deux approches existent afin d'améliorer l'efficacité de compression vidéo, et ne sont pas mutuellement exclusives. La première consiste à développer des outils de codage plus efficaces dans le cadre du processus de normalisation pour construire la prochaine norme vidéo. Les nouveaux outils proposés par JVET pour VVC ambitionnent d'augmenter l'efficacité du codage de 40% à 50% d'ici 2020 par rapport à HEVC [LTB18]. Cependant, ce gain s'accompagne d'une augmentation de la complexité, qui est environ de 3 à 4 fois la complexité d'HEVC côté encodeur et 2 fois côté décodeur [SRS<sup>+</sup>16, AACP16, LTB18]. Il est important de spécifier que ces chiffres varient régulièrement, puisque l'encodeur de référence est toujours en phase de développement. Cette approche est une solution à long terme et on observe généralement une décennie entre l'adoption de deux normes vidéo. De plus, le coût et le temps nécessaires au déploiement d'une nouvelle norme sont importants puisqu'il existe de nombreux dispositifs de décodage dans le monde, comparativement aux dispositifs d'encodage.

La deuxième approche vise à améliorer l'efficacité de codage d'encodeurs existants sans modifier la syntaxe du décodeur. Malgré la variété de normes de compression disponibles et leur différence en termes d'efficacité de codage, toutes utilisent le même schéma fondamental. Cette observation conduit à la possibilité pour toute optimisation d'encodeur, se concentrant sur un outil générique au lieu d'une implémentation spécifique à une norme, d'être transposable et de bénéficier à toute norme de compression, à quelques ajustements mineurs près.

Cette thèse se situe dans le cadre de la seconde approche et propose d'améliorer l'efficacité de compression via la prise en compte de dépendances. Les outils de codage visent à supprimer la corrélation entre les données, par exemple les images ou les pixels, afin de supprimer les informations redondantes. Malheureusement, ces outils sont également connus pour introduire des corrélations entre les décisions de codage de ces mêmes données, comme expliqué dans les Chapitres 1 et 2. Ces nouvelles corrélations sont des dépendances, et les prendre efficacement en compte permet d'obtenir un encodage efficace sur l'ensemble du signal vidéo. Le but de cette thèse est de se concentrer sur les décisions d'encodage, telles

que décrites dans Fig. 3, et de fournir une meilleure compréhension des dépendances, ainsi que des algorithmes efficaces d'optimisation globale.

## Structure de la thèse

**Chapitre 1:** Ce chapitre fournit des informations de base et décrit quelques principes de base du codage de source. Une description de la théorie débit-distorsion, de la quantification scalaire et du codage prédictif est fournie. Grâce à cette vue d'ensemble, les dépendances des décisions de codage connexes sont mises en évidence dans le contexte d'un schéma de codage classique par blocs. Notez que le terme *bloc* dans ce cas ne fait pas nécessairement référence aux blocs de pixels dans une image, mais simplement au groupe d'échantillons dans un signal. L'énoncé de ces dépendances inévitables montre les difficultés de l'optimisation exhaustive et brute force, ce qui conduit à une hypothèse simplifiée pour une solution d'encodage pratique.

**Chapitre 2:** Dans ce chapitre, vous trouverez un aperçu de la dernière norme de codage vidéo HEVC. Son schéma de codage est présenté avec une brève introduction des différents outils de codage disponibles dans la norme. Toutes les dépendances connexes sont exposées à la fin de ce chapitre et montrent un besoin critique de prise en compte des dépendances afin d'atteindre une efficacité de compression globale et optimale. Nous soulignons que ces dépendances sont liées au codage vidéo hybride puisqu'elles sont, pour la plupart d'entre elles, observables dans la plupart des normes de codage citées ci-dessus.

**Chapitre 3:** Le problème de l'optimisation globale en termes de performance débit-distorsion est d'abord considéré dans le contexte du codage Intra uniquement. La dépendance la plus couramment considérée est la propagation de distorsion, qui peut être modélisée ou introduite intuitivement dans le problème d'optimisation débit-distorsion. La plupart des méthodes de l'état de l'art présentées dans ce chapitre se concentrent sur cette dépendance. Cependant, nous expliquons que les autres dépendances peuvent également avoir une influence significative sur l'optimisation globale, en particulier dans le cas de débit élevé. Une étude est proposée pour évaluer l'efficacité de codage maximale, dans le contexte de l'optimisation conjointe de plusieurs modes de prédiction Intra. Une analyse permettant d'évaluer les possibilités d'une mise-en-œuvre peu complexe est également fournie.

**Chapitre 4:** La quantification adaptative optimale est l'un des points clés pour optimiser l'efficacité de codage des encodeurs vidéo. La compensation de mouvement est responsable d'un réseau de dépendances reliant tous les blocs d'un ensemble d'images. Dans ce chapitre,

ce réseau est décrit à l'aide d'un modèle de propagation des distorsions temporelles et d'une estimation efficace des probabilités des modes Inter et Skip. Des quantificateurs optimaux sont ensuite obtenus pour chaque bloc afin d'optimiser globalement le signal, en termes d'efficacité débit-distorsion. L'algorithme proposé surpasse plusieurs méthodes similaires de l'état de l'art. En plus de la démonstration analytique permettant d'obtenir les quantificateurs optimaux, nous proposons une analyse approfondie du comportement de l'algorithme.





# Introduction

## Preamble

Video-related technologies are past, present and undoubtedly future challenges. While a few decades ago, users mostly watched video contents on home TV or cinema, videos are now available and consumed from a variety of display devices, using IP or mobile networks. According to a recent study of the Cisco company in [Cis18], the video traffic over Global IP will account for 82% of all IP traffic by 2022, up from 75% in 2017. Note that in a previous report [Cis14], the company anticipated the video to account for 79% total IP traffic in 2018, which prove the reliability of these estimations. In a recent report [Eri18], Ericsson company announced a growth of video traffic in mobile networks around 35% annually for next 6 years.

These numbers can be explained by changes in usage and user habits. The growing consumption of video content is stimulated by the development of the *TV Everywhere* business model. The access to video content, using Internet and mobile applications, is conditioned by an authentication process instead of the use of specific display device such as the television. The concept is adopted by famous content providers such as Netflix, Youtube or Amazon. The growing popularity of e-sport also contributes to massively creating game streams. The Twitch platform, that mainly broadcasts game live streams, reports in [Twi] that during July 2018 around 1 million viewers at anytime is watching videos on their platform. Young generations (12-16years) are also active providers and consumers of videos through social media services as Facebook, Twitter, Snapchat and more recently TikTok. This dynamic environment is the reason for drastic changes in the way of consuming videos and the explanation of the growing worldwide video traffic.

In parallel to the tremendous content diversity and large number of solutions for accessibility, the video data volume growth is also explained by increasing end-user demands for quality and immersion. The resolution increased from [Standard Definition \(SD\)](#) to [High Definition \(HD\)](#) and we currently observe a growing popularity of [Ultra High Definition \(UHD\)](#) contents, matching the fact that all current TVs support [UHD](#). The development of

television and cameras manufacturers industries also allowed more fluidity in the display of contents. Other technological evolutions such as 3D, [High Dynamic Range \(HDR\)](#) and [Wide Color Gamut \(WCG\)](#) are also deployed in order to improve the [Quality of Experience \(QoE\)](#). Despite obvious benefits of these technicals progress, each of them implies a physical growth in the quantity of data to display.

The video data are more and more present in users everyday life and numerical size of video data is increasing. Optimal video compression, that consists into reducing the number of bits while minimizing the negative impact on visual quality, is the principal objective of this thesis. Even if networks and storage capacities are significantly higher than in the past, the conclusion is simple. Video compression is not only relevant: Video compression is a necessity.

## Context

Video compression is subject to a very dynamic environment from which it is difficult to escape. Compression standards impose some constraints and affect a very large number of companies, from encoder or decoder manufacturers to content providers. Therefore, it is important to know at all times what is happening inside the video compression community. Note that *code* and *compress* terms are used as synonyms in the remaining of this document.

## Video Compression Standard

The goal of a compression standard is to specify the bitstream syntax and the decoding process that allow to display the video on a device. Encoding consists into creating a bitstream compatible with a standard and aims to optimize the coding efficiency. The basic coding and decoding flow is depicted on the Fig. 3, along with the scope of the standard.

Currently, two lines of development for future video compression technology are in competition. The two standard organizations [International Organization for Standardization \(ISO\)/International Electrotechnical Commission \(IEC\)](#) and the [International Telecommunication Union \(ITU\)](#), are known to jointly developing video coding standards for over three decades. Their emerging contender is the industry-driven [Alliance for Open Media \(AOM\)](#).

### Standardization Organizations

The [Video Coding Experts Group \(VCEG\)](#) is a working group of the [ITU](#) responsible for the development of H.120, H.261 and H.263+ video coding standards, among others. The [Moving Picture Experts Group \(MPEG\)](#) is another working group formed by [ISO](#) and [IEC](#)

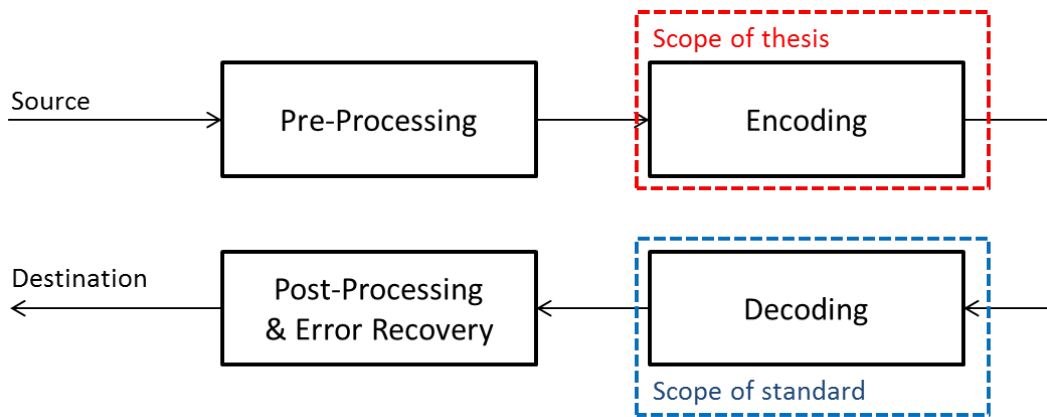


Fig. 3 Scope of the thesis and scope of a compression standard

for the standardization of audio and video coding and transmission. They are responsible for the MPEG-1 standard released in 1993 and MPEG-4 standard released in 1998. [VCEG](#) and [MPEG](#) groups first joint their efforts in video coding with the development of H.262/MPEG-2 part 2 in 1996.

The following partnerships between the two working groups were presented under project names such as [Joint Video Team \(JVT\)](#) for [Advanced Video Coding \(AVC\)](#) [[ITa](#), [ISOa](#)] in 2003 and [Joint Collaborative Team on Video Coding \(JCT-VC\)](#) for [High Efficiency Video Coding \(HEVC\)](#) [[ITb](#), [ISOb](#)] in 2013. While [AVC](#) is currently the most used video coding standard in the world, [HEVC](#) allow significant bitrate savings for equivalent quality and is subject to growing interest. The [Joint Video Exploration Team \(JVET\)](#), latest collaborative group between [VCEG](#) and [MPEG](#), has announced in April 2018 the development of [Versatile Video Coding \(VVC\)](#) that will be finalized in 2020. The new studied coding solutions [[CAS<sup>+</sup>17](#)] show the evidence of developing a new standard with coding capabilities beyond [HEVC](#).

### Alliance for Open Media

The [AOM](#) is an industry consortium aiming to provide royalty-free alternative to the video compression standards presented above. Several proprietary codecs, aiming to impose themselves on the market, have been initiated during the 90's with companies such as On2 (codecs VP3 to VP7) and RealNetworks (codec RealVideo). Another claim of the initiative is the desire to produce new standards faster and to speed up the innovation, compared to the standardization organization work speed.

[AOM](#) announced in March 2018 the release of AV1 version 1 specification in [[fOM](#)], an open source royalty-free video codec with enhanced video compression performance

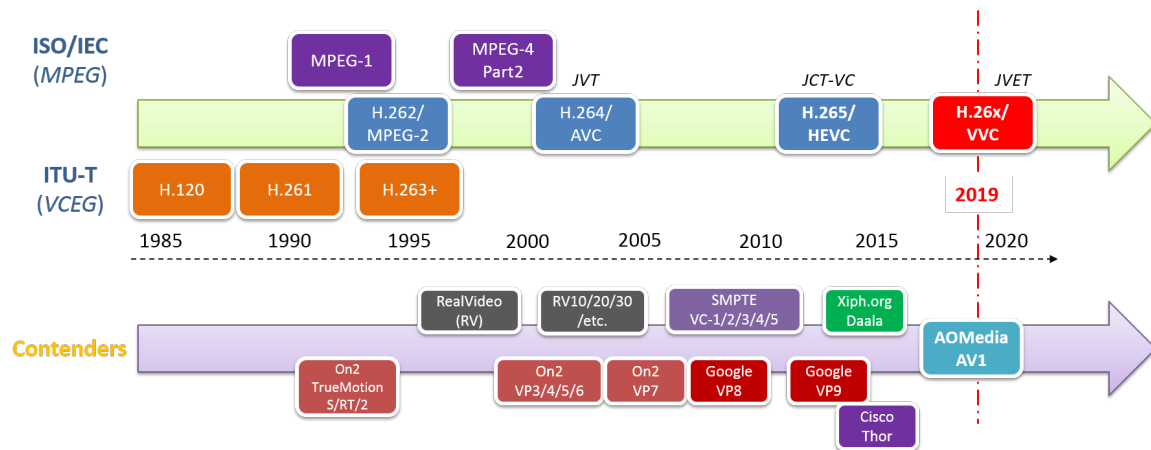


Fig. 4 History of Video Coding Standards (Modified Figure from [LTB18])

against state of the art. This emerging standard ambitions to compete the HEVC standard by proposing an efficient video compression scheme, without any limitations regarding to high cost or complex legal usage terms.

A summarized timeline of the video compression standards is proposed in Fig. 4.

## Encoder Design

A standard is designed to optimize the maximum coding efficiency by providing an optimized set of coding tools. However, there are no guarantee that an encoder exploits properly these coding tools and is able to provide the best quality. In industry, encoder vendors are specifically judged or competed on their capacity to optimize these encoding processes thanks to their proprietary algorithms, while they must comply to the same decoding process.

Consequently, the video coding world observes a competition of standards but also a competition of non-normative encoding algorithms. These algorithms are subject to several constraints due to the users application, e.g. in the case of [Digital Video Broadcasting \(DVB\)](#) a maximum bitrate constraint is set for continuous diffusion. Constraints in terms of computational complexity and delay are also very common for encoder design.

## Motivations

There are mainly two approaches to enhance video coding efficiency. The first way consists in developing more efficient coding tools within the standardization process to build the next generation video standard. New tools proposed by JVT for VVC aims to increase the coding efficiency by 40% to 50% compared to HEVC [LTB18]. However, this gain comes

with a complexity increase of 3 to 4 times the HEVC complexity at encoder side and 2 times at decoder side [SRS<sup>+</sup>16, AACP16, LTB18]. We point out that these numbers are often varying, since the reference model design is still under development phase. This approach is a long term solution and we usually observe a decade between the adoption of two video standards. Moreover, the required cost and time to the deployment of a new standard are important since there are plentiful decoding devices in the world compared to encoding devices.

The second approach aims to enhance the coding efficiency of existing standard encoders without changing the syntax of the decoder. Despite the variety of coding standards available and their difference in terms of compression efficiency, all of them use the same fundamental coding scheme. Such observation leads to the possibility for any encoder-side optimization, focusing on generic coding tool instead of specific standard implementation, to be transposable and benefits to any compression standard, with only minor adjustments.

This thesis focuses on the second approach and proposes to improve coding efficiency by considering dependencies. Coding tools often remove the correlation between data, e.g. frames or pixels, in order to suppress redundant information. Unfortunately, these tools are also known to introduce correlations between the coding decisions of these exact same data, as it is explained in Chapter 1 and Chapter 2. These newly introduced correlations are dependencies, and efficiently taking them into account allow to achieve efficient encoding over the entire video signal. The aim of this thesis is to focus on the encoding decisions, as depicted in Fig. 3, and to provide better insights about dependencies, along with efficient algorithms for global optimization.

## Outline

**Chapter 1:** This chapter provides background information and describes some fundamentals of the source coding. A description of the Rate-Distortion (R-D) theory, the scalar quantization and the predictive coding is provided. After this overview, related coding decisions dependencies are highlighted in the context of a classical block-based coding scheme. Note that the term *block* in this case does not necessarily refers to pixels blocks in a picture, but simply to samples group in a signal. Statement of these unavoidable dependencies exhibits the difficulties of exhaustive and brute force optimization, that further leads to simplified assumption for practical encoding solution.

**Chapter 2:** In this chapter, the latest released video coding standards is overviewed. The coding scheme is presented along with brief introduction of the different coding tools

available in the standard. All related dependencies are exposed in the end of this chapter and show a critical need for dependencies consideration in order to achieve global coding efficiency. We point out that these dependencies are related to Hybrid Video Coding since they are, for most of them, observable in most of the coding standards cited above.

**Chapter 3:** The problem of global optimization in terms of rate-distortion performance is first considered in the context of Intra-only coding. The most commonly considered dependency is the distortion propagation, that can be modeled or intuitively introduced into the [Rate Distortion Optimization \(RDO\)](#). Most of the state-of-the-art methods presented in this chapter focus on this dependency. However, we explain in this chapter that leftover dependencies may also have significant influence on global optimization, especially in the high-rate case. A study is proposed to evaluate the maximum achievable coding efficiency, in the context of joint optimization for Intra predictor. Analysis to evaluate the opportunities of low complexity implementation is also provided.

**Chapter 4:** Optimal adaptive quantization is one of the key points to optimize coding efficiency of video encoders. Motion compensation is responsible for a dependency network connecting all blocks of the same *GOP* together. In this chapter, this dependency network is modeled through a temporal distortion propagation model and an efficient use of Inter and Skip modes probabilities. Optimal quantizers are then designed per block in order to achieve the global optimization in terms of Rate-Distortion efficiency. The proposed algorithm outperforms several related methods from state-of-the-art. Moreover, along with the demonstration of optimal quantizer solution, we propose an in-depth analysis of the algorithm behavior.

# Chapter 1

## Source Coding

Video coding belongs to the field of source coding which is part of the information theory. In the following, we define a source signal as a random process that is a sequence of samples. When we refer to a *signal sample* or *source sample*, it must be treated as a random variable, continuous or discrete. We point out that samples are usually ordered, either temporally or spatially in the case of pixels from a video.

An important tool of modern coding systems is the entropy coding. This lossless tool usually aims to map symbols into a bitstream using a reversible operation. The mapping usually exploits statistical dependencies in order to reduce the transmission/coding rate. Among the entropy coding techniques, the two well known are Huffman coding presented by *Huffman* in [Huf52] and arithmetic coding presented by *Rissanen and Langdon* in [RG79]. These two methods significantly enhance the coding efficiency of video codecs. We do not further describe entropy coding in this text, since it is not the main topic of interest of this thesis. However, the reader is referred to the work of *Sayood* in [Say02] for further details.

In this chapter, some source coding fundamentals are reviewed. First, the R-D theory is addressed in Section 1.1 due to its importance in the encoding optimization process. Lossy and lossless compression techniques, respectively the quantization and predictive coding, are also discussed in Section 1.2 and Section 1.3. For further details on source coding, the reader is referred to the books of *Gray* [Gra90] and the one of *Wiegand and Schwarz* [WS11].

### 1.1 Rate-Distortion Theory

In the following, we consider a source signal  $\mathbf{X}$  that is an  $N$ -dimensional vector we indicate as  $\mathbf{X} = (X_1, \dots, X_i, \dots, X_N)$ , e.g. an image with  $N$  pixels. Each  $i^{\text{th}}$  sample of  $\mathbf{X}$ , designated by  $X_i$ , is a random variable.



In this section, a brief introduction of the R-D theory is given. First, we define the Distortion and the Rate, in the context of source coding. Second, the R-D function of a coding system is presented, which gives the best achievable coding efficiency for a given source. Then, we address the theoretical expression usually considered for the R-D function, hereafter called the R-D Shannon bound. Finally, the Lagrangian optimization for RDO is presented, which is usually used in practical video coding systems.

### 1.1.1 The Distortion

Source coding systems often use lossy coding techniques in order to achieve higher reduction of the data size with respect to lossless coding. These techniques introduce a measurable difference between the signal before coding, referred in this document as the *source signal*, and the signal after coding and decoding, named in this document the *reconstructed signal*. The measure of this difference is called the Distortion and is noted by  $D$ .

The Mean Square Error (MSE) is one of the most used distortion metric in the area of source coding. Its wide adoption into the coding community is due to the low computational cost and the ease of use in mathematical problems. For a source signal  $X$  and a reconstructed signal  $Y$  both composed of  $N$  samples, the MSE is defined by (1.1), with  $X_i$  being the  $i^{\text{th}}$  sample of the signal  $X$ .

$$MSE(X, Y) = \frac{1}{N} \sum_{i=1}^N \mathbb{E} \left[ (X_i - \hat{X}_i)^2 \right], \quad (1.1)$$

Since the topic of signal distortion is highly related to the use and nature of the signal, we choose to specify hereafter some images related metric. For representation matter in image coding, it is often preferred to represent the coding efficiency in terms of quality instead of distortion. The Peak Signal to Noise Ratio (PSNR) is a Visual Quality Metric (VQM) straightly computed from the MSE as expressed in (1.2).

$$PSNR(X, Y) = 10 \log_{10} \left( \frac{MAX^2}{MSE(X, Y)} \right) \quad (1.2)$$

In (1.2),  $MAX$  is the highest possible value on one sample. In the case of  $k$  bits coding per sample,  $MAX = 2^k - 1$ . Despite the benefits of the PSNR, MSE-based metrics may not properly evaluate the quality as a human observer would do. In the past decades, perceptual-oriented VQMs were proposed.

*Wang et al.* proposed the **Structural Similarity Index (SSIM)** metric [WBSS04] that figures in the most used **VQMs** and is considered more related to the perceived quality than the **PSNR**. The **SSIM** formula is given in (1.3). It is computed on a windows-basis for two signals  $X$  (source) and  $Y$  (reconstructed).

$$SSIM(X, Y) = \frac{(2\mu_X\mu_Y + c_1)(2\sigma_X\sigma_Y + c_2)(2cov_{XY} + c_3)}{(\mu_X^2 + \mu_Y^2 + c_1)(\sigma_X^2 + \sigma_Y^2 + c_2)(\sigma_X\sigma_Y + c_3)} \quad (1.3)$$

$\mu_X$  is the mean value of  $X$ ,  $\sigma_X^2$  is the variance of  $X$  and  $cov_{XY}$  is the covariance of  $X$  and  $Y$ . Each value is computed based on a given window.  $c_1$ ,  $c_2$  and  $c_3$  are model parameters, which are provided in the related paper of *Wang et al.* [WBSS04].

There are many other visual metrics more correlated with **Human Visual System (HVS)**. However, suitable perceptual **VQMs** is an entire field of research by itself. For in-depth understanding on perceptual **VQMs** topic, the reader is referred to the survey conducted by *Lin et al.* [LJK11].

## 1.1.2 The Rate

While viewers are mostly concerned by the observed quality, stream providers also care about the bitrate, commonly noted  $R$ . The rate is often expressed in bits per seconds, i.e. the amount of transmitted numerical data per second, or bits per samples. The rate of a source signal  $X$ , is defined here as the average rate of all samples, as described in (1.4). In the case of lossless coding, the minimum achievable rate for a sample  $X_i$ , that is a random variable, is equal to its entropy  $H(X_i)$  as defined by *Shannon* in [Sha48] and given in (1.5).

$$R(X) = \frac{1}{N} \sum_{i=1}^N R(X_i) \quad (1.4)$$

$$H(X_i) = -\mathbb{E}[\log_2 \mathbb{P}(X_i)] = -\sum_{a \in A} \mathbb{P}(X_i = a) \log_2(\mathbb{P}(X_i = a)) \quad (1.5)$$

$A$  is the dictionary, i.e. the set of possible values of  $X_i$ .  $\mathbb{P}(X_i = a)$  represents the probability that  $X_i$  takes the value  $a$ . When observing the entropy formula, we note that:

- In opposition to *sharp Probability Density Function (PDF)*, the *flatter* the random variable **PDF**, the bigger the entropy.

This observation is the basic justification for designing coding techniques that produces **PDF** as sharp as possible. Given a source distribution, a sharper **PDF** than the original may be obtained by using coarse approximation, e.g. sampling process, or by considering clustering to gather samples sharing similar properties.

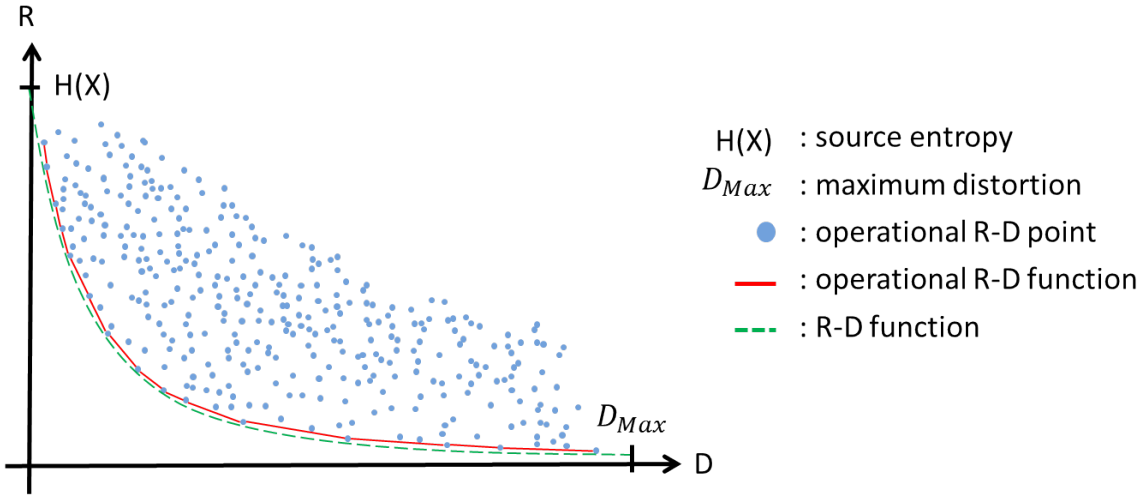


Fig. 1.1 Operational R-D function and the convex hull of operational R-D points as the modeled R-D function.

### 1.1.3 The Rate-Distortion Function

A coding system compliant with a given standard basically converts a source signal into a bitstream. A standard decoder is able to process the bitstream to produce the reconstructed signal. While a compression standard sets a maximum achievable coding efficiency for a given source signal, this optimal limit is rarely achieved in practice because of several constraints, e.g. the computational complexity and memory usage.

The encoding procedure estimates the optimal set of coding parameters  $\vec{p}^*$ , in terms of distortion (or rate), while matching constraints such as: latency, computational complexity, memory consumption, average rate (or distortion), maximum instantly allowed rate (or distortion). We note  $\vec{p} = (\vec{p}_1, \dots, \vec{p}_i, \dots, \vec{p}_N)$ , in the case each sample  $i$  of the  $N$ -dimensional vector source is coded with the coding parameters  $\vec{p}_i$ .

Rate distortion theory aims to find  $\vec{p}^*$  that minimizes  $D$  subject to a rate constraint  $R \leq R_T$ , with  $R_T$  being the target rate. An alternative approach is to optimize  $\vec{p}$  in order to minimize  $R$  with a distortion constraint  $D \leq D_T$ , with  $D_T$  being the target distortion (or quality). Whatever the considered problem, both solutions specify the lower bound of the so-named R-D function.

Setting  $\vec{p}$  to a given vector results into a unique encoding and one operational R-D point, i.e. a couple  $(D(\vec{p}), R(\vec{p}))$  estimated after the encoding procedure. On Fig. 1.1, we plot operational R-D points corresponding to different combinations of  $\vec{p}$ . We can observe a convex hull, called the R-D function, that is the boundary between realizable and non-realizable encodings.

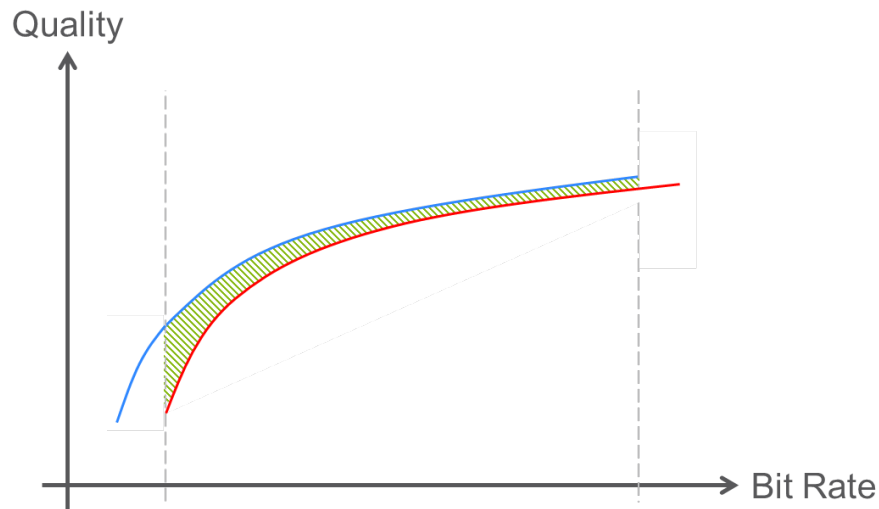


Fig. 1.2 Visual representation of the **BD-BR** metric to compare two coding systems based on their respective **RQ** curves.

It should be noted here that the operational **R-D** function is a piecewise constant function, drawn using affine functions. The operational curve may also be drawn using the Manhattan distance, that results into a staircase appearance. The second is sometimes preferred, because it is intuitively possible to achieve higher distortion at the same rate or equivalently higher rate with identical distortion. There is also no guarantee that a trade-off exists between two **R-D** points. However, the local tangent of the **R-D** function is easy to model and to differentiate, explaining why it remains most often the considered representation in **R-D** optimization problems.

In order to compare two coding systems, in the context of encoding algorithm evaluation for example, it may be intractable generating enough **R-D** points to estimate the operational **R-D** function. Consequently, only a few points are generated for each coding system, and their respective **R-D** functions are interpolated from these samples.

For comparison purpose of two coding systems, **R-D** curves are often replaced by **Rate-Quality (RQ)** curves. The approximated **RQ** functions, in the case of video coding, are usually compared using the **Bjontegaard-Delta Bit-Rate (BD-BR)** measure proposed by *Bjontegaard* [Bj01b]. This method compares the integral of **RQ** functions in the common range of rates, meaning that both functions are compared within an identical interval of integration. An example of the Bjontegaard measure between two **RQ** curves is presented in Fig. 1.2. Usually, the quality is expressed on a logarithm scale as in the example.

### 1.1.4 The Shannon Lower Bound

As shown previously, R-D functions can be obtained after several encodings. However, efficient encoding optimization often requires to know beforehand the relation between the rate and the distortion. Consequently, many algorithms of a coding system relies on a R-D model. The most common R-D model is the Shannon lower bound and is briefly discussed below. For further details about Shannon R-D function, the reader is referred to the works of *Cover and Thomas* [CT06] and *Wiegand and Schwarz* [WS11].

The *mutual information* is used to represent the amount of transmitted information, i.e. the rate. Let the mutual information between the source sample  $X_i$  and the reconstructed sample  $Y_i$ , written  $I(Y_i; X_i)$ , be defined by (1.6).

$$I(Y_i; X_i) = H(Y_i) - H(Y_i|X_i) \quad (1.6)$$

$H(Y_i)$  is the entropy defined in (1.5) and  $H(Y_i|X_i)$  is the conditional entropy.  $I(Y_i; X_i)$  measures the entropy reduction of the reconstructed sample when the source sample is observed. If samples are identical, implying  $H(Y_i|X_i) = 0$ , it results in a lossless coding process. As a consequence, the minimum achievable rate is the entropy of the source sample as shown in (1.7).

$$I(Y_i; X_i) = H(Y_i) = H(X_i) \quad (1.7)$$

It can be demonstrated, as done by *Wiegand and Schwarz* [WS11], that for *independent and identically distributed (iid)* memoryless sources and considering (1.6), the Shannon lower bound or Shannon R-D function is equal to (1.8).

$$R_i(D_i) = -\frac{1}{2} \log_2 \left( \frac{D_i}{c_i \sigma_i^2} \right) \quad (1.8)$$

With  $\sigma_i^2$  being the sample variance and  $c_i$  a constant depending on the sample distribution. Here are the possible  $c_i$  values for various distribution:

- Gaussian:  $c_i = 1$
- Laplacian:  $c_i = e/\pi$
- Uniform:  $c_i = 6/(\pi e)$

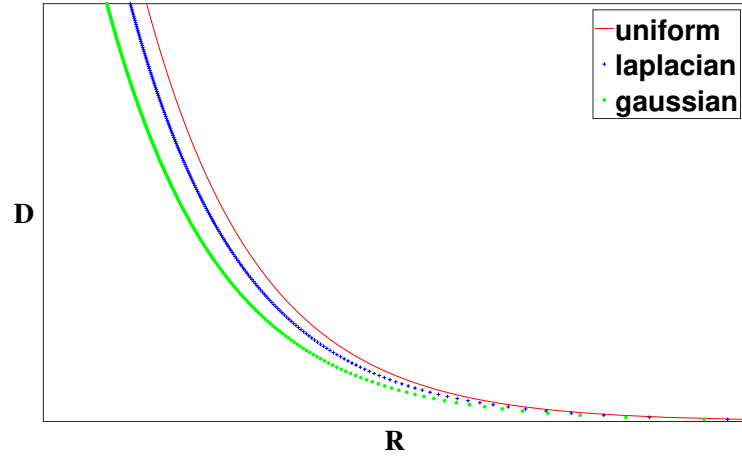


Fig. 1.3 Shannon bound for uniform, laplacian and gaussian distributions with  $\sigma_i^2 = 1$ .

The Shannon **R-D** functions for the three distributions are depicted in Fig. 1.3 with  $\sigma_i^2 = 1$ . In the case of small distortion, or equivalently high rate, the Shannon lower bound (1.8) is asymptotically equivalent whatever the source distribution. Such models are also tight to the observed **R-D** function for high rate as discussed by *Linder and Zamir* in [LZ94].

In case of Gaussian sources with memory, a more complex formulation of (1.8) is proposed in (1.9), with  $\rho$  being the correlation coefficient between successive samples. Interested readers may find the complete derivation of this formula given by *Wiegand and Schwarz* in [WS11].

$$R_i(D_i) = -\frac{1}{2} \log_2 \left( \frac{D_i}{(1 - \rho^2) \sigma_i^2} \right) \quad (1.9)$$

### 1.1.5 Rate-Distortion Optimization

In the following, we consider the problem of distortion minimization under rate constraint, named **Rate Distortion Optimization (RDO)**. The constrained problem is expressed in (1.10).

$$\begin{aligned} \min_{\vec{p}_i} \quad & \{D_i(\vec{p}_i)\} \\ \text{s.t.} \quad & R_i(\vec{p}_i) \leq R_T \end{aligned} \quad (1.10)$$

The decision core of an encoder aims to estimate the coding parameters  $\vec{p}_i$  that achieves **RDO** for the  $i^{\text{th}}$  sample. This constrained problem can be modeled thanks to the Lagrangian

multiplier method as proposed by *Everett* in [Eve63], with  $\lambda$  being the Lagrangian multiplier. The resulting unconstrained formulation of (1.10) to minimize is then defined as (1.11).

$$D_i(\vec{p}_i) + \lambda (R_i(\vec{p}_i) - R_T) \quad (1.11)$$

Since the target rate  $R_T$  is a constant, it does not impact the minimization, and the optimal set of coding parameters  $\vec{p}_i^*$  is then obtained through (1.12).

$$\vec{p}_i^* = \arg \min_{\vec{p}_i} \{J_i(\vec{p}_i)\} = \arg \min_{\vec{p}_i} \{D_i(\vec{p}_i) + \lambda R_i(\vec{p}_i)\} \quad (1.12)$$

$J_i(\vec{p}_i)$  is the R-D cost function the RDO tries to minimize. It should be noted that  $\vec{p}_i^*$  obtained in (1.12) with the correct  $\lambda$  value is also the optimal solution of (1.10). One important question is to define the optimal  $\lambda$  value. In the case of video coding,  $\lambda$  has been experimentally optimized by *Wiegand and Girod* [WG01] for H.263+ and by *Wiegand et al.* [WSJ+03] for AVC.

We note that the  $\lambda$  value considered in (1.12) controls the trade off between  $R_i$  and  $D_i$ , i.e. optimal  $\lambda$  is dependent of the R-D function. The R-D function is intuitively impacted by source signal characteristics, such as samples variance or source distribution, as shown in the Shannon bound expression (1.8). Consequently, the optimal  $\lambda$  value is constant if and only if the signal is stationary, i.e. source samples characteristics do not vary over time. Source signals such as audio, image or video are unlikely to be stationary.

Considering that each variation of  $\vec{p}_i$  induces a different coding efficiency, an usual RDO implementation is to exhaustively test all possible values of  $\vec{p}_i$  within a restricted set. Then the optimal coding parameters  $\vec{p}_i^*$  that minimize the R-D cost, potentially under some constraints, are chosen to compute the bitstream. This restricted set of possible encodings can be narrowed for several reasons:

- If the user or a parallel algorithm set some coding parameters beforehand.
- If parameters are made dependent on others' values, e.g. coding parameters of other samples.

In other terms, the search space of such exhaustive procedure is defined by a priori encoding configuration and encoder algorithms. Intuitively, one understands that larger search space leads to better coding efficiency, but also increased computational complexity. This problem is discussed later in Section 1.5.

For video coding purpose, a comprehensive description of the RDO is given by *Sullivan and Wiegand* in [SW98] and an overview of R-D methods is presented by *Ortega and Ramchandran* in [OR98]. Two distinct coding procedures, i.e. two decision cores or search strategies, are numerically compared by estimating their respective coding efficiency with respect to a quality score. An example of coding efficiencies comparison has been introduced in Section 1.1.3.

### 1.1.6 Link with the central limit theorem

The R-D function and related optimization processes focus on an overall evaluation of the coding system. It means the trade off is made between the total introduced distortion and the global rate of the signal  $X$ , i.e. we optimize  $X$  coding by finding  $\bar{p}^*$ . As presented in the beginning of this section,  $D$  and  $R$  are expressed as sum of samples distortions and samples rates. One problem is that previously proposed R-D models applies to a random variable, i.e. a sample  $i$  of the source signal, with a distribution that may be unknown.

According to the central limit theorem, summing a high number of *iid* random variables ( $N \rightarrow +\infty$ ) leads to a random variable that follows a Gaussian distribution. The consequence is that a vector of samples, that we conveniently name *block*, can be treated as a random variable and use R-D models such as (1.8) or (1.9). Because of this convergence into a simple distribution, efficient coding tools for Gaussian distributions can be very useful and efficient for coding systems that process vectors of samples instead of processing each sample independently.

Each sample  $X_i$  is a random variable with a distortion  $D_i$ , a variance  $\sigma_i^2$  and a rate  $R_i$  defined by the R-D Shannon bound. Basically, to use this theorem is equivalent to define a block of  $N$  samples, with a rate  $R_B$  and a distortion  $D_B$  defined as the functions in (1.4) and in (1.1), respectively. Then, according to the central limit theorem, the block can be considered as random variable  $\bar{X}$  that follows the Gaussian distribution with variance  $\sigma_B^2$ . We can write the following statements:

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N X_i \quad (1.13)$$

$$\sigma_B^2 = \frac{1}{N} \sum_{i=1}^N \sigma_i^2 \quad (1.14)$$



The related Shannon bound is then defined in (1.15).

$$R_B = -\frac{1}{2} \log_2 \left( \frac{D_B}{\sigma_B^2} \right) = -\frac{1}{2} \log_2 \left( \frac{\sum_{i=1}^N D_i}{\sum_{i=1}^N \sigma_i^2} \right) \quad (1.15)$$

The central limit theorem was first demonstrated by *De Moivre* [DM38] for a restricted case and was later generalized and popularized by *Laplace* [Lap12].

## 1.2 Quantization

In the context of signal processing, each sample of the signal takes a value within a dictionary  $A$ . The quantization is the process of mapping input signal samples from the dictionary  $A$  to a smaller dictionary  $B$ , i.e.  $Card(A) > Card(B)$ . The dictionary  $B$  results into a countable and finite set. The obtained signal can further be unmapped from  $B$  to  $A$  in order to obtain a reconstructed signal, different from the input signal. The mapping process is an irreversible approximation that introduces losses in the signal, named the quantization error. The algorithm or process that applies the quantization on the signal is named the quantizer.

In this section, we focus on the scalar quantization, which is extensively used in image and video coding. A brief overview of the scalar quantization is first provided, along with the optimality conditions for a quantizer to minimize the quantization error. The relationship between the distortion, the rate and the quantizer is analyzed, in order to explain hereafter the trade-off function of the quantizer in the RDO problem. Finally, the well-known high rate R-D function is presented, due to its importance in the next sections of this document.

### 1.2.1 Scalar Quantization Description

In the case of video coding, the scalar quantization is the classical considered method. In the following, the input signal  $X$  is considered as continuous and each sample  $X_i$  takes value from a continuous set. Conclusions of this section hold for a discrete source. In the following we only focus on the quantization of the random variable  $X_i$ . Let us define the following notations:

- $Q$  is the quantizer. We note  $Q(x)$  the quantization of one value  $x$  of the input signal, i.e.  $X_i = x$ .
- We note  $K = Card(B)$  the number of reconstruction levels, in the dictionary  $B$ . We point out that the quantizer output is necessarily a discrete source.
- $t_k$  is named the  $k^{th}$  decision threshold or interval boundary with  $k \in \{1, \dots, K\}$ .

- $I_k = [t_k; t_{k+1}[$  is the  $k^{\text{th}}$  interval of quantization.
- We note  $y_k \in B$  the  $k^{\text{th}}$  reconstruction level associated to  $I_k$ , i.e.  $Q(x) = y_k \forall x \in I_k$ .
- $\Delta_k = t_{k+1} - t_k$  is the interval size or the quantization step.

An example of the mapping for the uniform quantization is given in Fig. 1.4 (a). The uniform quantizer with dead-zone as presented in Fig. 1.4 (b) is also an efficient alternative, especially if the source signal to compress have a PDF that is symmetric around zero and reaches its peak value at zero.

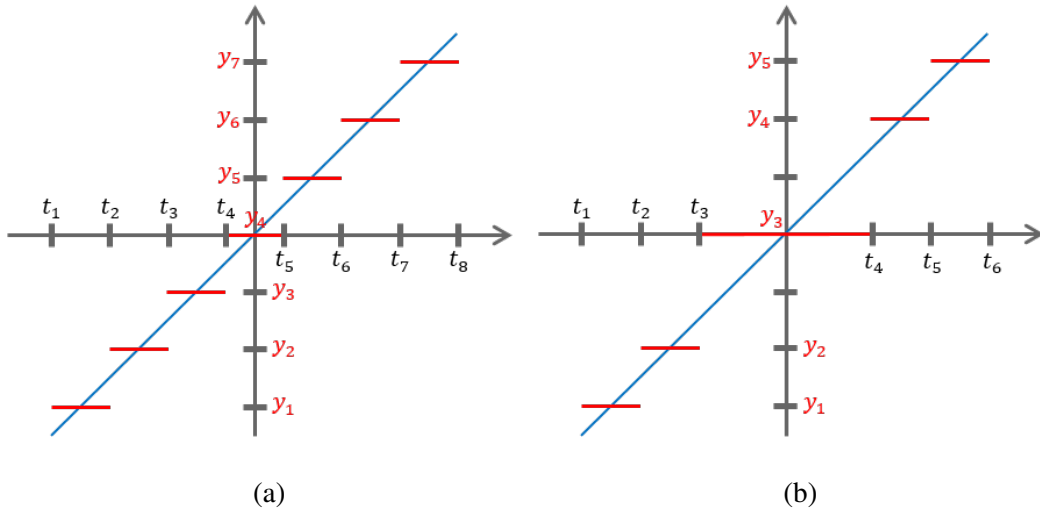


Fig. 1.4 Input and Output of Scalar Quantization for Uniform Quantizer (a) and Uniform Quantizer with Dead-zone (b)

Due to the quantization, there are losses and the quantization error  $D_i$  is measurable. The distortion of a scalar quantizer is defined by (1.16).

$$D_i(Q, f_{X_i}) = \sum_{k=1}^K \int_{t_k}^{t_{k+1}} f_{X_i}(x) D(x, Q(x)) dx \quad (1.16)$$

With  $f_{X_i}(x)$  being the PDF of  $X_i$ . The quantization error metric for one value  $x$  is designated by  $D(x, Q(x))$ . In the case of MSE, (1.16) is turned into (1.17).

$$D_i(Q, f_{X_i}) = \sum_{k=1}^K \int_{t_k}^{t_{k+1}} f_{X_i}(x) (x - Q(x))^2 dx \quad (1.17)$$

Two conditions are necessary for the design of an optimal quantizer in terms of distortion: the *generalized centroid condition* (1.19) and the *nearest neighbor condition* (1.20).

In the case of known decision thresholds, the distortion is minimized if the distortion of each interval  $D_k(y_k)$  is minimized. It leads to the optimal reconstruction level  $y_k^*$  in (1.18) and the optimal for the squared error distortion in (1.19).

$$y_k^* = \arg \min_{y_k} \{D_k(y_k)\} = \arg \min_{y_k} \left\{ \int_{t_k}^{t_{k+1}} f_{X_i}(x) D(x, y_k) dx \right\} \quad (1.18)$$

$$y_k^* = \frac{\int_{t_k}^{t_{k+1}} x f_{X_i}(x) dx}{\int_{t_k}^{t_{k+1}} f_{X_i}(x) dx} \quad (1.19)$$

In the case of known reconstruction levels, decision threshold  $t_k$  is optimal if the distortions of surrounding intervals are equal. It leads to the optimal decision threshold  $t_k^*$  in terms of MSE described in (1.20).

$$t_k^* = \frac{1}{2} (y_{k-1} + y_k) \quad (1.20)$$

Detailed description of optimality conditions is given by *Wiegand and Schwarz* [WS11] and by *Max* [Max60]. We can see in (1.19) and (1.20) that  $y_k^*$  and  $t_k^*$  depend on each other. An iterative solution is given by *Llyod* [Lyy82] and result in the well-known *Llyod-Max quantizer*.

## 1.2.2 Quantization Control for Rate-Distortion Optimization

Since the quantization is a lossy coding procedure, it may be involved in the RDO process. In the following, we consider the case of a uniform quantizer for which only the quantization step  $\Delta$  that can be optimized while the reconstruction levels  $t_k$  are set to the middle of their respective interval of quantization. As shown in (1.16) for smooth distribution, the distortion  $D(\Delta)$  increases when  $\Delta$  increases. In order to validate the R-D Shannon bound from Section 1.1.4, we also want to verify the rate behavior with respect to  $\Delta$ .

We consider the average rate to be well approximated by the entropy of the reconstructed signal. The rate achieved by the quantizer is consequently defined by (1.21), with  $\mathbb{P}(y_k)$  the probability of  $y_k$  defined by (1.22).

$$R_i(\Delta) = - \sum_{k=1}^K \mathbb{P}(y_k) \log_2(\mathbb{P}(y_k)) \quad (1.21)$$

$$\mathbb{P}(y_k) = \int_{t_k}^{t_{k+1}} f_{X_i}(x) dx \quad (1.22)$$

After some developments, the rate  $R_i(\Delta)$  may be decomposed<sup>1</sup> into (1.23).

$$R_i(\Delta) = H(\mathbf{X}_i) - \log_2(\Delta) \quad (1.23)$$

From (1.23), we can easily deduce that if the quantizer step is high, the rate tends to zero. On the opposite, if the quantizer step is low ( $\Delta \rightarrow 0$ ), the rate tends to infinity. However, this infinite rate value is purely theoretical. For pictures and videos,  $\mathbf{X}_i$  is a discrete random variable, hence there is a minimal value  $\Delta_{min}$  which is equal to the granularity of  $\mathbf{X}_i$ , i.e. that does not produce distortion. We define  $\mathbf{X}_i$  to be discretized in integer values and  $\Delta_{min} = 1$ . Consequently, the highest achievable rate  $R_{max}$  is estimated as the source entropy according to (1.24).

$$R_{max}(\Delta) = H(\mathbf{X}_i) - \log_2(\Delta_{min}) = H(\mathbf{X}_i) - \underbrace{\log_2(1)}_{=0} = H(\mathbf{X}_i) \quad (1.24)$$

The quantization step is proven to control the trade-off between distortion and rate, that is similar to the  $\lambda$  in the Lagrangian optimization for RDO expressed in (1.12). The quadratic relation between  $\Delta$  and  $\lambda$  has been verified in the context of video coding by *Sullivan and Wiegand* [SW98].

### 1.2.3 High-Rate R-D approximation

The High-Rate R-D function is a well known approximation used in video coding, mostly because of its mathematical tractability. As expressed in (1.23), in terms of rate-quantization, high rate is equivalent to  $\Delta \rightarrow 0$ , i.e.  $\mathbf{X}_i$  is uniformly distributed under each interval. Minimum and maximum values of  $\mathbf{X}_i$  are written  $x_{min}$  and  $x_{max}$ , respectively. For an uniform quantizer with  $K$  intervals, the quantization step size  $\Delta$  results in (1.25) and  $y_k$  probability in (1.26).

$$\Delta = \frac{x_{max} - x_{min}}{K} \quad (1.25)$$

$$\mathbb{P}(y_k) = \frac{1}{x_{max} - x_{min}} = \frac{1}{\Delta K} \quad (1.26)$$

The MSE distortion is expressed as:

$$D_i(Q) = \int_{x_{min}}^{x_{max}} f_{\mathbf{X}_i}(x) (x - Q(x))^2 dx. \quad (1.27)$$

<sup>1</sup>Detailed computations are provided in Appendix A.1

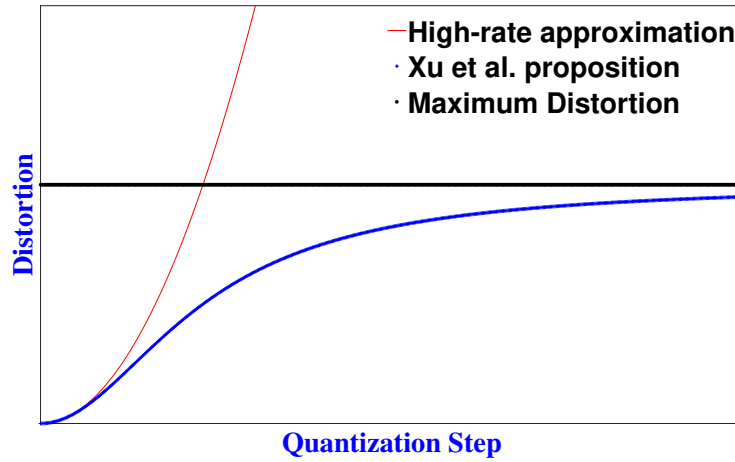


Fig. 1.5 High-Rate approximation and *Xu et al.* [XJGZ07] model of the Distortion-Quantization relationship, with distortion expressed in MSE, against the maximum achievable distortion  $\sigma_i^2$ .

In the case of uniform distribution within an interval of quantization, the optimal reconstruction level for the MSE is at the middle of the interval. The distortion is further simplified into the High-Rate approximation (1.30) with the following steps:

$$D_i(\Delta) = K \times \int_{-\Delta/2}^{\Delta/2} f_{X_i}(x) x^2 dx \quad (1.28)$$

$$D_i(\Delta) = K \times \int_{-\Delta/2}^{\Delta/2} \frac{1}{\Delta K} x^2 dx \quad (1.29)$$

$$D_i(\Delta) = \frac{\Delta^2}{12} \quad (1.30)$$

For the video coding application case, *Xu et al.* proposed in [XJGZ07] another model for  $D_i(\Delta)$  that is given in (1.31), with  $\sigma_i^2$  being the variance of the input sample.

$$D_i(\Delta) = \frac{\sigma_i^2 \Delta^2}{12\sigma_i^2 + \Delta^2} \quad (1.31)$$

This proposal is tight to (1.30) in the high-rate case, but does not suffer from the same inaccuracy in the low-rate case, as exposed in Fig. 1.5. Indeed, the distortion is limited by  $\sigma_i^2$  the input variance, but the high-rate approximation suggests to overcome this maximum at some point.

More complex models of  $D(\Delta)$  have also been proposed, such as the one suggested by *Ropert and Ropert* in [RR13]. The model expressed in (1.32) is based on the Laplacian distribution of coefficients obtained after **Discrete Cosine Transform (DCT)** and inequality constraints approximation introduced in [RR13].

$$D_i(\Delta) = \Delta^2 \left( \frac{2}{a^2} + \frac{2}{e^a - e^{-a}} \left( 1 - \frac{2}{a} \left( \ln \left( \frac{1 + e^a}{2} \right) + 1 \right) \right) \right), \quad (1.32)$$

with

$$a = \frac{\sqrt{2}\Delta}{\sigma_i}. \quad (1.33)$$

## 1.3 Predictive Coding

We stated in Section 1.1 that the entropy of an input signal gives the lower boundary achievable rate for its transmission in the case of lossless coding. However, this minimal limit only stands in the case of *iid* variables. If there are statistical dependencies, the lowest achievable rate is the conditional entropy, that is lower than the entropy of the input samples. In Section 1.2, it has been demonstrated that transmission rate may be reduced below entropy limit using quantization, but at the cost of creating distortion to the reconstructed signal.

In this section we present the predictive coding. The prediction allows to exploit the source statistical dependencies in order to reduce the achievable rate without introducing distortion. We first provide the main principle of a predictive encoder. Then the linear prediction is presented because of its wide use in video coding, including its optimality conditions and some properties. Finally, the joint use of prediction and quantization is discussed, along with the optimization based on predictors competition.

### 1.3.1 Principle of a Predictive Encoder

The basic structure of a prediction is presented in Fig. 1.6. Each sample  $X_i$  that belongs to the input signal  $X$  is predicted based on other samples of the signal, stored in an internal memory. We note  $\vec{p}_i$  the predictor parameter,  $\hat{X}_i(\vec{p}_i)$  the predicted value of  $X_i$ . The difference between source sample and its prediction yields a residue or prediction error  $\varepsilon_i(\vec{p}_i)$  expressed in (1.34). We point out that in the signal processing community, the residue is also called the innovation, i.e. the unpredictable part of the signal.

$$\varepsilon_i(\vec{p}_i) = X_i - \hat{X}_i(\vec{p}_i) \quad (1.34)$$

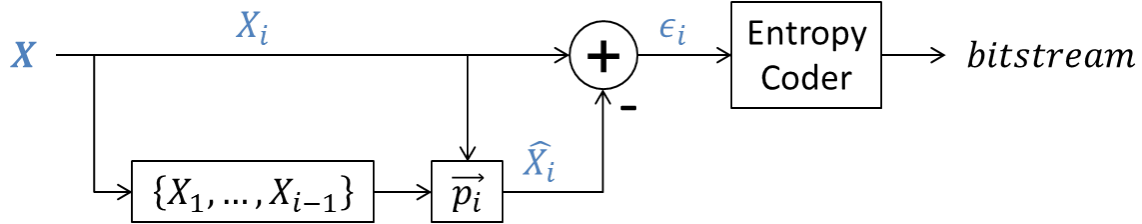


Fig. 1.6 Basic Encoder structure for predictive coding

Prediction is done using a memory that refers to the previously coded samples of the input signal. It should be taken into account that depending of the prediction requirements, such as the minimum number of samples to store in the memory, the implementation of a predictive coding can suffer from large memory requirements. A classical example is a video encoder that requires to store several pictures to apply the prediction process. Consequently, it may imply a large memory usage.

In a simple predictive coding scheme, only the residue is transmitted after the entropy coding. The decoder further reconstructs the sample by adding the received prediction error to the predicted value. The decoder scheme is presented in Fig. 1.7. Since the input signal, such as an image or an audio, is assumed to have high statistical dependencies, the entropy of the residue is much lower than the entropy of the original source signal.

The minimum achievable rate using a predictive coding system depends on the residue entropy. Hence, the optimal predictor from the encoder point of view minimizes its residue entropy. However, as explained by *Wiegand and Schwarz* in [WS11], the MSE minimization as proposed in (1.35) is a good criteria for predictor optimality.

$$\vec{p}_i^* = \arg \min_{\vec{p}_i} \epsilon_i^2(\vec{p}_i) = \arg \min_{\vec{p}_i} (x_i - \hat{x}_i(\vec{p}_i))^2 \quad (1.35)$$

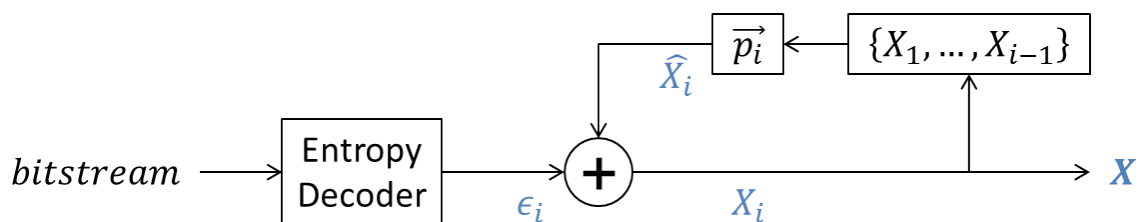


Fig. 1.7 Basic Decoder structure for predictive coding

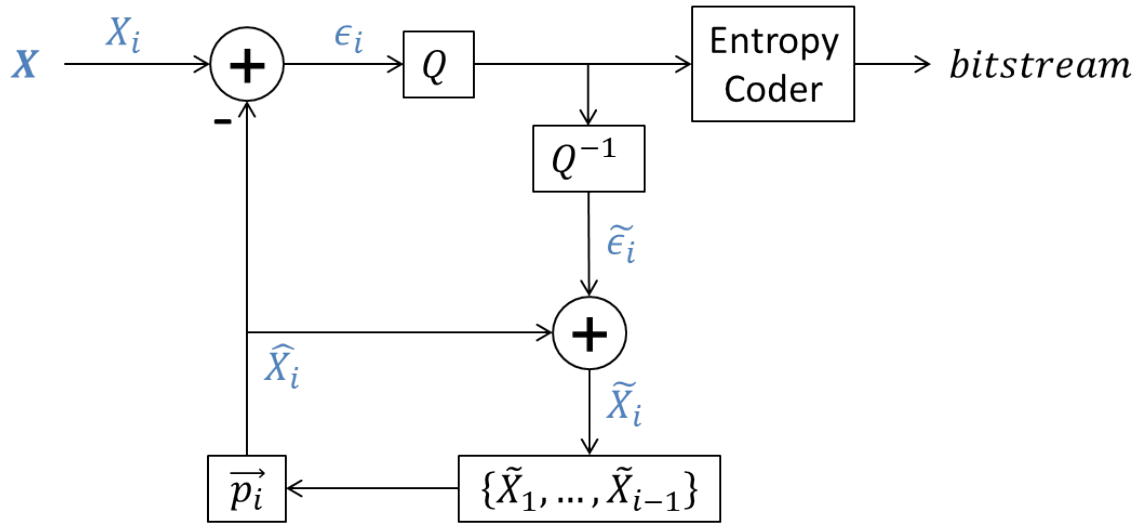


Fig. 1.8 Closed-loop encoder design for lossy predictive coding

### 1.3.2 Prediction in the RDO context

In the following we focus on the linear prediction, that consists into estimating discrete samples of an input signal based on a linear function of previous reconstructed samples. Linear prediction can be applied to scalars, one sample is predicted at a time, or be applied to vectors, one group of samples are predicted at a time. Many applications of predictive coding, such as the video coding standards make use of both prediction and quantization in order to achieve the RDO. Quantization is applied to residue samples instead of source ones. The corresponding lossy coding scheme is presented in Fig. 1.8.

One major difference between Fig. 1.6 and Fig. 1.8 is the feedback-loop that includes the reconstruction of the output samples at the encoder-side in order to fill the memory. In the case of lossy coding, source and reconstructed signals may be different. It becomes necessary to include the reconstruction in the encoder and only use reconstructed samples as reference for prediction. An encoder is often called a *codec*, portmanteau of coder-decoder, because of this feedback-loop. When considering dependencies in a coding system, it is important to notice the causalities such as the one between prediction and previously reconstructed samples, i.e. the causal relationship between prediction and reference's distortion.

In the context of predictive coding coupled with quantization, the RDO is the central problem. In the case of MSE measure, the distortion between  $N$  source samples  $X_i$  and reconstructed samples  $\tilde{X}_i$  is equal to the distortion between its residue  $\epsilon_i$  and quantized residue  $\epsilon'_i$  as shown in (1.36).  $\hat{X}_i$  is the predicted sample of  $X_i$ .



$$\frac{1}{N} \sum_{i=1}^N (X_i - \tilde{X}_i)^2 = \frac{1}{N} \sum_{i=1}^N (\varepsilon_i + \hat{X}_i - (\tilde{\varepsilon}_i + \hat{X}_i))^2 = \frac{1}{N} \sum_{i=1}^N (\varepsilon_i - \tilde{\varepsilon}_i)^2 \quad (1.36)$$

A necessary and sufficient condition of optimal linear predictor is the orthogonality principle. It states that the prediction residue is uncorrelated with the observed data, i.e. reference and prediction error are uncorrelated. This observation is an important start point for encoding optimization of predictive coding system. It enables a good mathematical tractability of the relationship between distortions of the current sample and its prediction reference, as shown in next chapters of this document.

Comprehensive treatments of the orthogonality principle and more generally the estimation theory are given by *Kay* in [Kay93]. For further reading about the linear prediction, we also refer the reader to the well-known tutorial provided by *Makhoul* in [Mak75].

### 1.3.3 Optimization of Predictive Coding

We previously assumed that the encoder and decoder are based on unique and deterministic predictor. In order to achieve the RDO, a common utilization of the predictive coding is to use several predictors and to compare them in terms of R-D efficiency, as proposed in Section 1.1.3. This solution allows the encoder to be more adaptive and handle efficiently different source distribution. It also requires multiple prediction to estimate the best predictor, which may drastically increase the computational complexity of the encoder. In such predictor competition situation, the decoder must be informed with the chosen predictor. It induces the transmission of additional data, different from the residue, called *side informations*.

Predictive coding may be assimilated to parametric coding, widely used in the area of audio coding. For the reader interest, an overview of parametric coding for audio is provided by *Purnhagen* in [Pur99] and an application case for stereo audio is addressed by *Breebaart et al.* in [BvdPKS05]. Parametric coding is defined as modeling an input random variable, written  $X_i$ , with a parameters vector  $\vec{p}_i$  defined in (1.37).

$$\vec{p}_i = \{p_i(1), \dots, p_i(j), \dots, p_i(N_{param})\} \quad (1.37)$$

with  $N_{param}$  being the number of parameters. In the following, we call the parameters vector as the prediction vector. Each parameter takes value in its own set  $P_j$ , i.e.  $p_i(j) \in P_j$  and increases the total bitrate by  $R_i(p_i(j))$ . An optimal prediction vector minimizes the residue energy at the cost of a syntax bitrate defined in (1.38).

$$R_i(\vec{p}_i) = \sum_{j=1}^{N_{param}} R_i(p_i(j)) \quad (1.38)$$

There are two different solutions to improve such predictive system in terms of residue energy minimization. The first one consists in increasing the size of prediction vector  $N_{param}$  by adding additional parameters in order to better model the input signal in a more complex form. However, additional coding parameters induce an extra cost, in terms of bitrate, because of supplementary side informations to transmit to the decoder. Second, increasing the possible values of a parameter, i.e. to extend the subspace size or granularity of  $P_j$ , in order to enable a more accurate selection of the optimal prediction parameter value.

Both solutions are most likely to provide better prediction and to reduce the entropy of the residue. However, it implies a rate increase due to the diversity and the increasing entropy of these side informations. The control of these *communicating vessels*, illustrated in Fig. 1.9, between the syntax bitrate and the residue bitrate is one of the key to achieve encoding optimization in modern codecs. We can see on the graph that the Predictor B, requires more syntax bitrate than the Predictor A, but estimates more accurately the source signal, resulting into lower residue energy. However, the comparison of the final R-D functions shows that in low rates context, the syntax overhead becomes non-negligible and the predictor B results into worst coding performance than the predictor A.

## 1.4 Dependencies related to Source Coding

Some fundamentals about source coding coding has been presented in the last subsections. The coding process can be summarized in the following steps:

1. Input signal is divided into individual samples, or group of samples, defined as blocks.
2. Blocks are then predicted based on available reconstructed samples in memory. The predicted block is subtracted to the source one, resulting in residue.
3. The residue is processed by quantization and/or lossless coding methods, before transmission to the decoder.
4. Inverse operations are further applied on the transmitted residue. It allows to reconstruct an approximate version of the source samples and store them in the internal memory, at both the encoder-side and decoder-side.

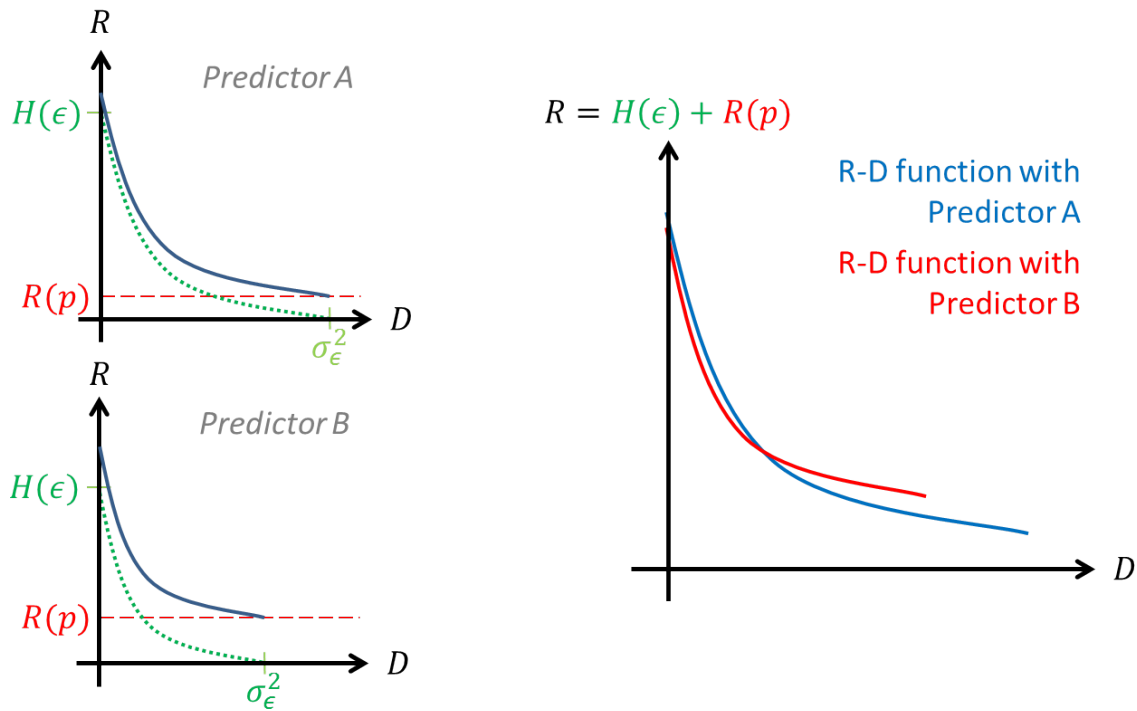


Fig. 1.9 Illustration of the syntax bitrate and the residue energy relationship

Source coding aims to achieve a global optimization under constraint, i.e. reaches the highest coding efficiency for the complete source signal while meeting requirements. These requirements can be the computational complexity, the encoding delay or output signal characteristics (quality or rate). However, the global optimization is not straightforward for large size signals. Because of the considerable amount of data, it is intractable to process all samples at once. This is the first reason for dividing the signal into blocks that are processed sequentially.

We consider a predictive coding scheme using quantization process such as the one presented in Fig. 1.8. An important observation must be taken into account: the prediction step introduces a sample dependency. This dependency is also stated as the distortion propagation in this document. We consider the sample  $X_i$  that is predicted from the previous reconstructed sample  $\tilde{X}_{i-1}$  as shown on (1.39). We point out that  $X_i$  may be a scalar or a vector.

$$\begin{aligned}
 \hat{X}_i &= \tilde{X}_{i-1} \\
 &= \hat{X}_{i-1} + \tilde{\epsilon}_{i-1} \\
 &= X_{i-1} - \epsilon_{i-1} + \tilde{\epsilon}_{i-1}
 \end{aligned} \tag{1.39}$$

$\varepsilon_i$  stands for the residue before quantization and  $\tilde{\varepsilon}_i$  is the residue after quantization as defined in Fig. 1.8. We focus on  $\sigma_i^2$  that is the variance of prediction error, i.e. the maximum achievable distortion according to the Shannon bound described in Section 1.1.4. The variance of prediction error is first introduced as in (1.40).

$$\begin{aligned}
\sigma_i^2 &= \mathbb{E} [\varepsilon_i^2] \\
&= \mathbb{E} [(X_i - \hat{X}_i)^2] \\
&= \mathbb{E} [(X_i - \tilde{X}_{i-1})^2] \\
&= \mathbb{E} [(X_i - X_{i-1} + d_{i-1})^2]
\end{aligned} \tag{1.40}$$

$d_{i-1}$  is the distortion introduced on  $X_{i-1}$  according to (1.36). The variance of prediction error expressed in (1.40) can be developed in order to obtain (1.41).

$$\sigma_i^2 = \underbrace{\mathbb{E} [X_i^2] + \mathbb{E} [X_{i-1}^2] - 2\mathbb{E} [X_i X_{i-1}]}_{=\mathbb{E}[(X_i - X_{i-1})^2]} + \underbrace{\mathbb{E} [d_{i-1}^2]}_{=D_{i-1}} + \underbrace{2\mathbb{E} [X_i d_{i-1}]}_{=0} - \underbrace{2\mathbb{E} [X_{i-1} d_{i-1}]}_{=C_{i-1}} \tag{1.41}$$

Consequently, we see that  $\sigma_i^2$  depends on the distortion of its reference. The term  $2\mathbb{E} [X_i d_{i-1}]$  is supposed to be zero according to the orthogonality principle of optimal linear predictor, as introduced in Section 1.3.2. We obtain a variance formula as (1.42).

$$\sigma_i^2 = \mathbb{E} [(X_i - X_{i-1})^2] + D_{i-1} - C_{i-1} \tag{1.42}$$

The term  $C_{i-1}$  may be perceived as the correlation measure between the reference and its distortion. As an intuitive explanation, one considers the case of the reference distortion being a random noise uncorrelated with its source samples. Consequently,  $C_{i-1}$  tends to zero and the injected noise may be transmitted to  $X_i$  prediction, i.e.  $D_{i-1}$  impacts is not diminished. Note that in the high-rate case, distortion may be assimilated to random noise.

The formulation in (1.42) shows that two samples of a signal related by a prediction process should not be optimized independently. Moreover, if several predictions are cascaded the joint optimization may become intractable, because it would be equivalent to optimize the whole signal at once. In the next subsection, we estimate the complexity of such a joint optimization approach.

## 1.5 Global Optimization in Source Coding

### 1.5.1 Joint Optimization versus Independent Optimization

In a predictive coding standard, the **RDO** technique is applied sequentially to parts of the signal, named blocks in the following for convenience. A block may be a frame in a video, a frequency in an audio signal or a group of pixel in a picture. We note  $B_i$  the  $i^{\text{th}}$  coded block,  $\vec{p}_i$  the vector defining the set of coding parameters to estimate for  $B_i$  and  $J_i$  the local **R-D** cost of  $B_i$ . For a signal composed of  $N$  blocks, the global optimum is obtained by solving (1.43).

$$\min_{\vec{p}_1, \dots, \vec{p}_N} \left\{ \sum_{i=1}^N J_i(\vec{p}_1, \dots, \vec{p}_N) \right\} \quad (1.43)$$

$\{\vec{p}_1, \dots, \vec{p}_N\}$  represent all the set of coding parameters from  $B_1$  to  $B_N$ . Without loss of generality, (1.43) can be rewritten into (1.44) by using the causality of a block-based compression scheme.

$$\min_{\vec{p}_1, \dots, \vec{p}_N} \left\{ \sum_{i=1}^N J_i(\vec{p}_1, \dots, \vec{p}_i) \right\} \quad (1.44)$$

Equation (1.44) is usually simplified based on the common assumption of independence between blocks, assuming that  $J_i$  only depends on  $\vec{p}_i$ . Under this hypothesis, one defines the **Independent Rate-Distortion Optimization (Independent-RDO)** minimization problem in (1.45).

$$\min_{\vec{p}_1, \dots, \vec{p}_N} \left\{ \sum_{i=1}^N J_i(\vec{p}_i) \right\} = \sum_{i=1}^N \min_{\vec{p}_i} \{J_i(\vec{p}_i)\} \quad (1.45)$$

Although the independence assumption significantly simplifies the computational complexity, as further explained in the complexity formalization hereafter, it may significantly reduce the coding efficiency of the complete signal. Methods that jointly optimize multiple blocks or  $\vec{p}_i$  components are stated **Joint Rate-Distortion Optimization (JRDO)** methods in this document.

### 1.5.2 Complexity formalization

For  $B_i$ , the set of parameters  $\vec{p}_i$  is the vector of coding decisions.  $p_i(j)$  is defined in (1.37) as the value of the  $j_{\text{th}}$  coding parameter of  $\vec{p}_i$  and  $N_{\text{param}}$  is the total number of parameters used to code a block, i.e. the vector size of  $\vec{p}_i$ .

Examples of coding parameter may be the prediction mode or the quantization parameter. Each parameter is defined into a different space  $p_i(j) \in P_j, \forall i \in \{1, \dots, N\}$  and  $\forall j \in \{1, \dots, N_{param}\}$ . The coding decision set  $\vec{p}_i$  is defined within vector space  $P$  defines in (1.46).

$$P = P_1 \times P_2 \times \dots \times P_{N_{param}} \quad (1.46)$$

We consider the RDO process as an exhaustive search to optimize  $J_i(\vec{p}_i)$ . The basic number of  $B_i$  encodings to determine  $\vec{p}_i^*$  is then equal to  $Card(P)$  defined in (1.47).

$$Card(P) = \prod_{j=1}^{N_{param}} Card(P_j) \quad (1.47)$$

For a given block, the complexity worst case (exhaustive search) is then equal to  $Card(P)$ . In case  $N_{param}$  is high, the search space described by  $P$  is usually not fully analyzed because of the required computational complexity. In order to limit the complexity of optimizing one block, each coding parameter can be optimized independently from others. It allows to reduce the search space and consequently the computational complexity. In such independent case, (1.47) is turned into (1.48).

$$Card(P) = \sum_{j=1}^{N_{param}} Card(P_j) \quad (1.48)$$

In the following, the complexity of analyzing  $B_i$  is noted  $Cpx(B_i) = Card(P)$ . By considering the dependencies between blocks exposed in Section 1.4 and the JRDO equation written in 1.44, the complexity  $Cpx(X)$  of optimizing the whole signal  $X$  is equal to (1.49).

$$Cpx(X) = \prod_{i=1}^N Cpx(B_i) = Cpx(B_i)^N \quad (1.49)$$

Obviously,  $Cpx(X)$  may easily become unrealistic. This is one historical reason for the use of independent assumption between blocks in most encoding systems. According to (1.45) the complexity of exhaustively optimizing  $X$  is defined in (1.50) and ensures more tractable computational complexity.

$$Cpx(X) = \sum_{i=1}^N Cpx(B_i) = N \times Cpx(B_i) \quad (1.50)$$

At this point, a quick example is given to illustrates the complexity of joint optimization compared to independent optimization. By considering one parameter, i.e.  $\vec{p}_i$  is 1-

dimensional vector, that takes value over 35 possibilities, and the joint optimization of 4 distinct coding blocks. The corresponding ratio of complexity exposed in (1.51) assumes that joint optimization requires more than 10000 times the computations of the independent optimization, which may be intractable for real-time application.

$$\frac{\prod_{i=1}^N Cpx(B_i)}{\sum_{i=1}^N Cpx(B_i)} = \frac{35^4}{35 \times 4} = 10718.75 \quad (1.51)$$

In the next Chapter, we discuss the hybrid video coding standard [HEVC](#) and its related dependencies for global optimization purpose.

# Chapter 2

## The hybrid video coding standard: H.265/HEVC

### Introduction

The HEVC standard [ITb, ISOb] was released in 2013 by the JCT-VC established jointly by the ITU-T VCEG and the ISO MPEG. HEVC enables up to 50% bitrate savings compared to AVC [ITa, ISOa] for equal perceptual video quality. *Le Tanou et al.* announce in [LTB18] an increase in complexity of a factor 4x and 2x for encoding and decoding, respectively. An overview of HEVC is provided by *Sullivan et al.* in [SOHW12] and an overview of AVC is provided *Wiegand et al.* in [WSBL03]. *Ohm et al.* proposed in [OSS<sup>+</sup>12] a comparison of the coding efficiency for video coding standards.

Like its predecessor, HEVC is a block-based coding standard. The video sequence is divided into Group of Pictures (GOPs), each GOP being composed of several frames. Each frame is then divided into blocks of pixels. These blocks are further processed using coding tools such as the quantization or the prediction, respectively introduced in Section 1.2 and Section 1.3. MPEG standards are also called *hybrid* as they combine temporal prediction between pictures of the video sequence with transform coding techniques applied on the prediction error.

The architecture of an HEVC video encoder is presented in Fig. 2.1. Frames are first sequenced, depending of the requested coding scheme that may be decided prior to the encoding procedure. This stage is called the sequencing and defines some hierarchy between pictures. Each picture is then split into blocks, using a QuadTree procedure in the case of HEVC, as described by *Kim et al.* in [KML<sup>+</sup>12]. A predicted signal is generated for each block, using either the Intra-frame prediction or the Inter-frame prediction, named the Motion





In the next parts of this section, a non-exhaustive list of coding tools used in HEVC are summarized. Detailed descriptions of HEVC are provided by *Sze et al.* in [SBS14] and *Wien* in [Wie15].

## 2.1 Frames Sequencing

Coded video streams are divided into GOPs, that are composed of coded frames following a given sequencing configuration. Frames are partitioned into one or several slices and each slice can be independently decoded from other slices in the picture. Slices can be among three major types and the GOP structure defines how frames are arranged in term of referencing for prediction. In the following of this document, we do not consider slice segmentation of frames and each one is only composed of one slice. When designing the GOP structure, each frame is defined by the following parameters:

- Slice type may be either I, P or B.
- **Picture Order Count (POC)** is the frame number in display order.
- The frame number in coding/decoding order.
- Reference Pictures indicate slices that are used as reference for motion compensation.
- **Quantization Parameter (QP)** that is dependent of the frames hierarchy.

The different types of slices and the usual GOP structures are defined in the following sections.

### 2.1.1 Slices Types

**I-slice:** Intra coded slice, is coded only using the Intra modes depicted in Section 2.3. An I-slice only refers to already coded samples in the current slice for prediction and never to other slices samples. Consequently, it can be coded and decoded independently from other slices. However, I-slices are usually the less recurrent slice type in a GOP since the bit cost of Intra modes is usually higher than the bit cost of Inter modes. An I-slice can also be part of an **Instantaneous Decoder Refresh (IDR)** frame. IDR frame have same properties than a frame only composed of I-slices, but additionally carry high level syntax elements set for all following slices, until the next IDR.

**P-slice:** Predictive coded slice, may use Intra or Inter coding modes,. The prediction is restricted to only refer to one previously coded slice. Usually this slice corresponds to a past frame, in terms of display order (**POC**), but the possibility to use slice in future frame is also allowed. P-slices are more efficient in terms of coding efficiency than I-slices, but require to decode at least one slice before being coded or decoded.

**B-slice:** Bi-directional coded slice, may use Intra or Inter coding modes, but suffer less restriction than P-slices. B-slices may refer to slices in past and/or future frames using forward or backward motion compensation, respectively. This type of slices can also use bi-prediction, i.e. consider two reference slices at once. *Generalized B-slices*, written P/B-slices, are introduced in **AVC** in order to allow forward/forward and backward/backward bi-prediction in addition to the forward/backward original use. B-slices are the most efficient in terms of coding efficiency, thanks to the numerous coding possibilities, but also the most computationally complex.

For simplification purpose, we use the terminology I-frame, P-frame and B-frame to refer to frames composed of an I-slice, P-slice or B-slice, respectively. The notion of **GOP** often refers to two different coding structures and there is a common confusion about it. In the first definition, a **GOP** refers to a periodic coding structure of successive pictures that is used to encode the video sequence. The second definition, the **GOP** term refers to the coding structure between two successive I-frames. Note that the periodic structure is repeated an integer number of times between two I-frames. In this thesis, we aim to achieve global optimality and jointly optimize the larger number of frames. Hence, we choose to use the **GOP** term to designate successive pictures between two I-frames.

### 2.1.2 Common Coding Structures

We present in the following the four most used scenarios and related **GOP** structures in current applications of video codings.

- **All-Intra (AI)** only considers I-frames as depicted in Fig. 2.2. Such scheme is usually less complex than others by avoiding the **Motion Estimation (ME)**. I-frame also have the useful property of being editable on a frame-by-frame basis, without extensive decoding. Due to the almost zero-latency of this configuration, it is often used for primary contribution television using satellite transmission. Post production is also a common use case of **AI** configuration due to its ease of edition.
- **Random-Access (RA)** is the coding scheme that enables the highest coding efficiency, usually at the cost of high computational complexity and some delay. **RA** uses the

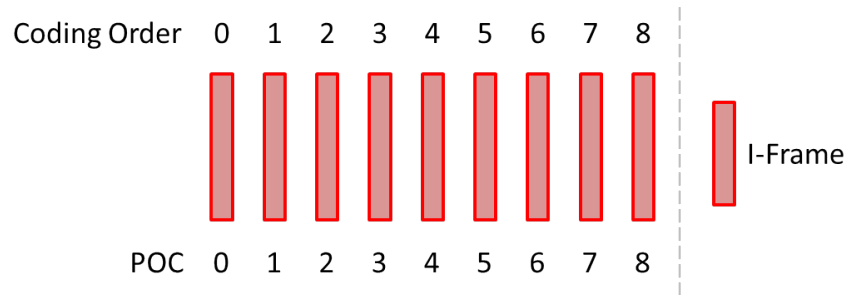


Fig. 2.2 All-Intra Configuration

hierarchical B structure, that is presented by *Schwarz et al.* in [SMW05] and depicted in Fig. 2.3. Classical use cases of such scheme are the [Internet Protocol Television \(IPTV\)](#) and [Over-The-Top \(OTT\)](#) services.

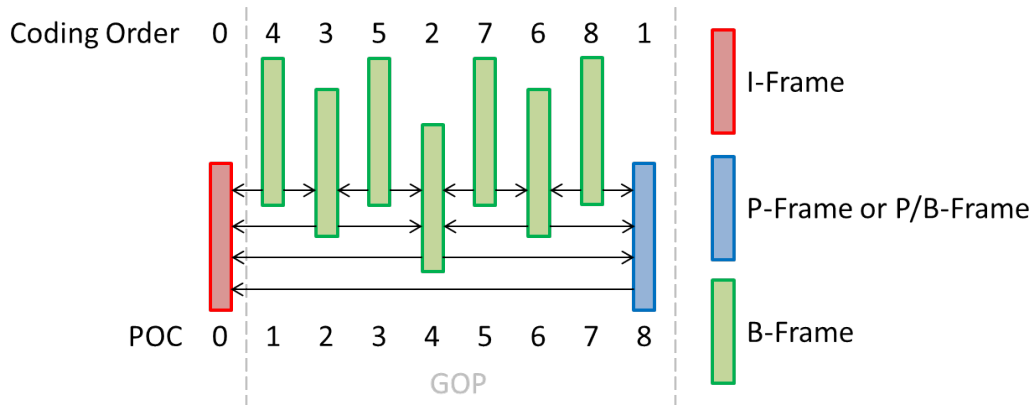


Fig. 2.3 Random Access Configuration

- **Low-Delay P (LDP)** classically starts with one I-frame followed by a large number of P-frames. As shown by the coding scheme illustrated in Fig. 2.4, frames are not reordered and POC and coding order are identical. As long as the encoding frame-rate matches the display frame-rate, such coding structure does not introduce any additional delay in the video transmission. The higher coding efficiency compared to the [AI](#) structure and the low latency compared to the [RA](#) structure makes it an obvious solution for the video conferencing use case.
- **Low-Delay B (LDB)** is almost identical to the [LDP](#), but consider B-frames instead of P-frames. Thanks to the bi-prediction, highest coding efficiency may be achieved compared to [LDP](#). No additional delay is introduced, i.e. POC and coding order remain identical, but the encoding process may be more complex for each frame, due to the additional coding options.

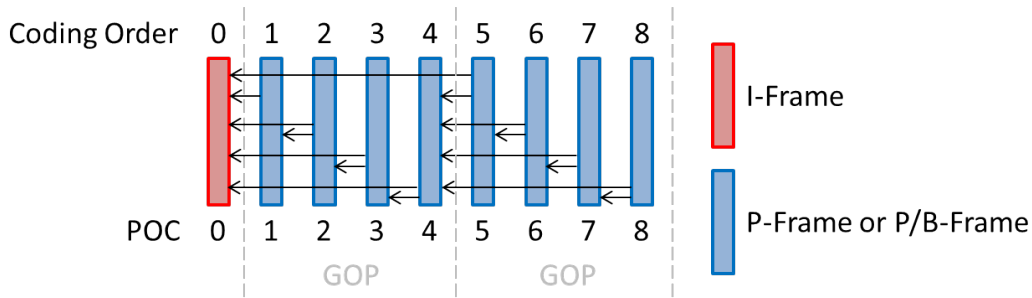


Fig. 2.4 Low Delay Configuration

HEVC uses a complex yet efficient referencing system. *Sjöberg et al.* proposed a detailed description of the reference picture management in [SCF<sup>+</sup>12].

## 2.2 QuadTree Partitioning

In this subsection we present three of the unit types used to partition a frame during an HEVC encoding: **Coding Tree Unit (CTU)**, **Coding Unit (CU)** and **Prediction Unit (PU)**. A first framework using different concepts of units have been proposed by *Han et al.* in [HMK<sup>+</sup>10]. One general remark concerning the partitioning is that large partitions work well on smooth areas, whereas small partitions are useful for highly-textured areas. A complete description of the HEVC partitioning and the coding efficiency of partition parameters is provided by *Kim et al.* in [KML<sup>+</sup>12].

### 2.2.1 Coding Tree Units and Coding Units

In HEVC, each frame is uniformly partitioned in CTUs, equivalent to **MacroBlocks (MBs)** in AVC. CTUs are sequentially compressed in raster scan order. Then, each CTU can recursively be further sub-divided in multiple CUs, following a QuadTree structure. Fig. 2.5 (a) shows an illustration of the partitioning of a CTU in HEVC. CUs within a CTU are coded in a recursive Z-scan order. The size of a CTU can be square of sizes 64x64, 32x32 or 16x16 and is set for the entire coding procedure, up to the next IDR picture.

CUs are of size  $2N \times 2N$  with  $N \in \{32, 16, 8, 4\}$ . Video coding community often referred to the size of a CU as the *depth* of the QuadTree. The largest CTU size (Ex. 64x64) is equivalent to the minimum depth 0 and the 8x8 size (smallest) is equivalent to the maximum CU depth 3.

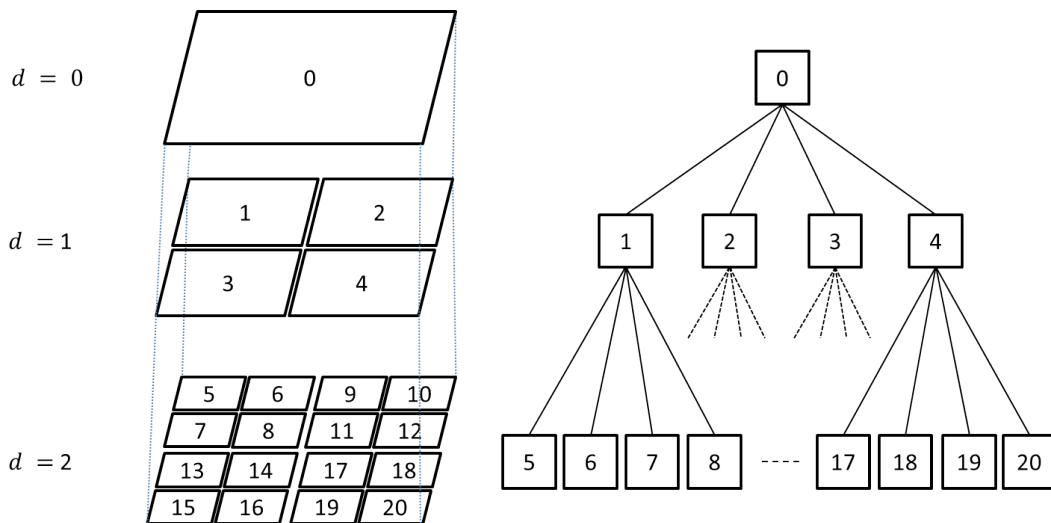


Fig. 2.5 Illustration of the recursive QuadTree Partitioning in HEVC

## 2.2.2 Prediction Units

Each CU at a given depth of the QuadTree can be predicted in one, two or four partitions, named PUs. The possible partitions for a PU are presented in Fig. 2.6.

The prediction is applied on each PU independently, whatever the number of PUs within the CU. However, all PUs that belong to the same CU use the same type of prediction (Intra or Inter). We point out that Intra coding modes only support the squared PUs, hence  $2N \times 2N$  or  $N \times N$ .

## 2.3 Intra Coding

Intra prediction consists in sequentially predicting the source signal from neighboring reconstructed pixels within the same frame, used as reference. Prediction is built by copying or interpolating reference pixels onto target pixels, according to a rule specified by the predictor. In the case of Intra coding, reference pixels used to predict the current PU are depicted in Fig. 2.7.

In HEVC, 35 possible predictors for Intra coding are available and presented in Fig. 2.7. The DC-mode uses the average value of reference pixels and the Planar-mode is a bilinear interpolation designed to preserve continuities along block boundaries. The 33 angular modes represent a direction of projection to the reference pixels. An Intra coded CU of size  $2N \times 2N$  is always composed of one PU of the same size, except for the highest allowed CU depth for which 4 PUs of size  $N \times N$  can be used.

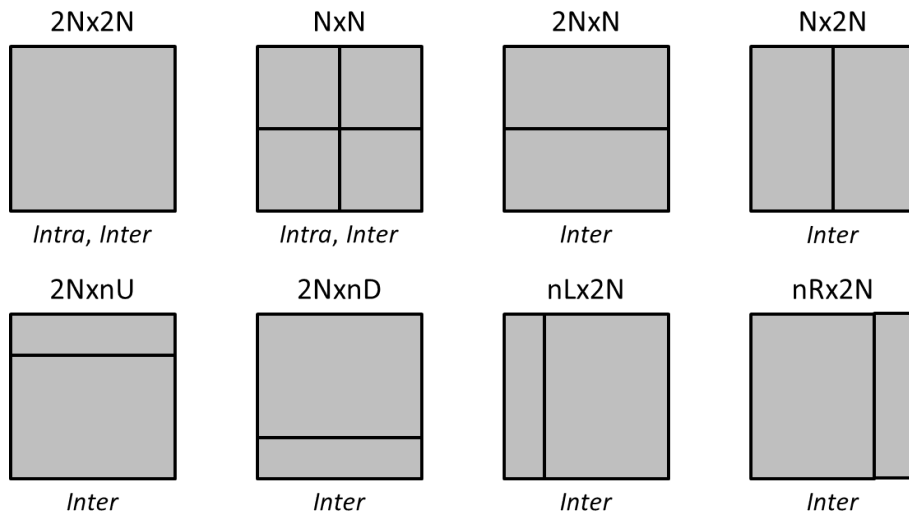


Fig. 2.6 Available Prediction Unit partitions in HEVC

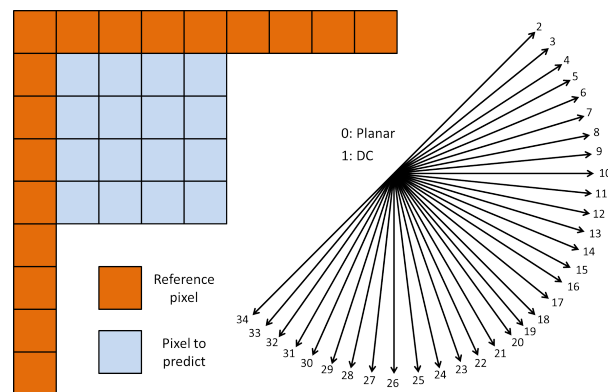


Fig. 2.7 Intra Prediction in HEVC

It must be noted that a second Quadtree, named the residual QuadTree is applied on each PU. Prediction information is carried at the PU-level. However, in the specific case of Intra coding, prediction is effectively processed on units defined by this residual QuadTree, named Transform Units (TUs). This process is highly effective to code gradual luminance/chrominance variation in a consistent direction.

Reconstructed samples used for prediction are available in the DPB at the decoder side, that is further added to the residue. However, the decoder also needs to receive the Intra predictor, in order to process the prediction. Coding the Intra predictor can take advantage of statistical redundancies with spatially previous coding modes due to the local spatial correlation. The Intra predictor can be coded in two ways.

The first solution to transmit the coding mode is based on the Most Probable Modes (MPMs). Using the surrounding PUs coding modes, i.e. left and above PUs modes, a set

of 3 **MPMs** is selected. In case the considered Intra predictor matches one of the predictors included in the **MPMs**, only the index within this reduced set is transmitted. In the case **MPMs** are not used, the second option is to transmit the Intra predictor using a fixed length code of 5 bits.

For a complete explanation on Intra coding in HEVC standard, the reader is referred to the work of *Lainema et al.* in [LBH<sup>+</sup>12].

## 2.4 Inter Coding

Inter coding is briefly presented by focusing on two important processes: the motion estimation/compensation and the motion vector prediction. Also, a description of special coding modes *Skip* and *Merge* is provided.

### 2.4.1 Motion Estimation/Compensation

Two solutions exist to deal with the temporal redundancy present in a video signal, the frame (or block) difference and the **Displaced Frame Difference (DFD)**. In the case of frame difference, the co-located block in the reference picture is used as a predictor, while **DFD** considers a motion compensated block. In natural sequences, motion is present due to moving objects, shot transitions or camera movements.

The **ME** consists into estimating, through block matching technique, the apparent motion of the current block one tries to predict. It aims to find corresponding points between the current block and the reference frame in order to obtain the optimal predictor, associated to a **Motion Vector (MV)**. Once the **MV** is estimated, the reference frame is compensated in order to be re-aligned with the current frame. Hence, the Inter prediction is also named the **Motion Compensated Prediction (MCP)** and is represented in the Fig. 2.8. In **HEVC**, the **MCP** is applied at the **PU** level. The **ME**, that is highly computationally complex, must be applied independently for each **PU**. That explains why it is often considered as a complexity bottleneck in an encoder.

### 2.4.2 Motion Vector Prediction

The **MV**, similarly to the Intra predictor, is not transmitted to the entropy coder as raw data. A prediction process named the **Advanced Motion Vector Prediction (AMVP)** is used in order to obtain the predicted **MV**, and the encoder only transmits the **Motion Vector Difference (MVD)** that is the prediction error between the actual **MV** and its corresponding predicted **MV**. In **HEVC**, two candidates are available and the **AMVP** chooses the predicted



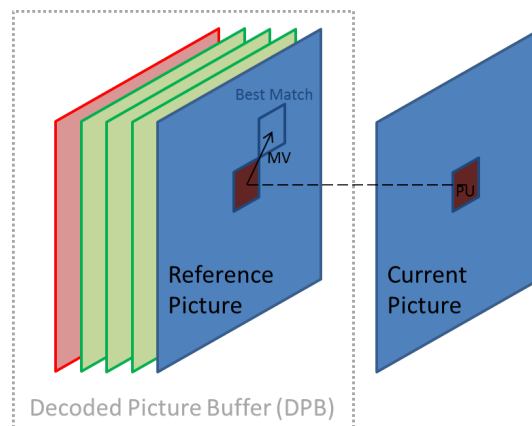


Fig. 2.8 Motion Compensation Illustration

**MV** among these candidates. The index of the predicted **MV** in the list, signaled by a boolean flag for **AMVP**, is required for the decoding.

The concept of using the motion prediction is similar to the use of **MPMs** in Intra coding. One assumes the motion to be locally homogeneous either spatially or temporally. The case of consistent motion on the spatial dimension is taken into account by adding motion vectors from the left and above **PUs** within the same frame to the candidate list. If there is less than 2 spatial candidates, the **MV** of the co-located block in the reference frame may be used as a candidate temporal predictor.

### 2.4.3 Merge and Skip modes

Two particular coding modes may be identified among Inter coding modes: Merge<sup>1</sup> and Skip. Inter coding consists in motion compensation and **AMVP** in order to transmit the residual samples and the motion syntax data to the decoder, including the predicted **MV** index and the **MVD**. However, sometimes the motion may be consistent on a large spatial area, e.g. during a camera panning. The predicted **PU** may also be identical to the current one, e.g. static background scene. Merge and Skip modes have been designed to take advantages of these situations.

As for the **AMVP**, several candidates in a list are considered as potential predicted **MV**. The list is derived from spatial and temporal neighboring **MVs**, but five candidates are considered compared to the two in **AMVP**. In the case of Merge mode, each potential predicted **MV** is evaluated in order to obtain the best predictor, i.e. the **MVD** is set to zero. The high coding efficiency of this mode comes from the possibility to *merge* large

<sup>1</sup>The Merge mode was introduced in the **HEVC** standard and works similarly to the Direct mode introduced in the **AVC** standard.

number of PUs and only define once their motion, since each PU inherits the motion from neighboring PUs. The design and coding efficiency of the Merge mode are detailed by *Helle et al.* in [HOB<sup>+</sup>12].

The Skip mode, as the Merge mode, does not involve any ME nor include motion data outside the predicted MV index. The additional feature of the Skip mode is to not transmit any residue. Consequently, the predicted signal is equivalent to the reconstructed one. Due to its design, the Skip mode is signaled as a particular Merge mode case. However, the Skip mode have a particular behavior in terms of R-D and complexity. Its computational complexity is lower than other modes, its rate is theoretically zero and its resulting distortion cannot be altered using other parameters such as the quantization step  $\Delta$ . For these reasons, it is often referred as one of the three main modes when designing encoding optimization algorithms: Intra, Inter and Skip.

## 2.5 Transform and Quantization

Each CU is split into PUs in order to be efficiently predicted, but is also split into TUs following a second QuadTree named the Residual QuadTree (RQT). A TU is a squared block of size 4x4, 8x8, 16x16 or 32x32, composed of coefficients resulting from DCT functions applied on the residue. The DCT-II is presented by *Ahmed et al.* in [ATR74] and an analysis of the DCT coefficients in image processing is provided by *Lam and Goodman* in [LW00]. The Discrete Sine Transform (DST)-VII may also be used in place of the DCT-II for the particular case of 4x4 Intra prediction.

DCT and DST have convenient properties for the video coding purpose:

- The transformation is orthogonal and reversible, thus the inverse operation may be implemented at the decoder side with transpose transform. It should be noted that for standard compliance matters, a fixed point approximation is considered on transforms, to get a full fixed point reversibility.
- Most often, energy is more compacted in the frequency domain than in the spatial domain. Spatial information is for the most part included in the low frequencies and often the highest frequencies do not carry any information. It results into using less bits to transmit the same information.
- The HVS happens to be more sensitive to low frequencies variations than high frequencies variations. When using the quantization, this property allow to quantize more aggressively high frequencies than the low ones and thus save rate by introducing barely noticeable distortion.

Once the residue is transformed, the obtained transform coefficients are quantized according to the **QP** value. **QP** is set at the **CU** level and takes integer value in a set as  $QP \in \{0, \dots, 51\}$ . Small values result into low distortion, while high values correspond to high distortion. The **QP** corresponds to a quantization step-size  $\Delta$ , the distance between two reconstructed level, as defined by the function 2.1.

$$\Delta^2 = 2^{\frac{QP-4}{3}} \quad (2.1)$$

Finally, quantized transform residues are processed and arranged by an entropy coder before being transmitted to the decoder. Detailed information about the transform coefficient coding is provided by *Sole et al.* in [SJM<sup>+</sup>12a].

For encoding optimization purpose, some algorithms named **Adaptive Quantization (AQ)** methods may be used. The concept is to apply different **QP** for each **CU** in order to better correlate with source distribution or aim for global optimization. In order to signal the different **QPs**, a quantization parameter is predicted based on above and left **CUs** respective **QP**, and only the difference with this predicted value is transmitted, named the *delta quantizer*.

## 2.6 Entropy Coding

The different tools available in **HEVC** and presented above produce a number of parameters, named **Syntax Elements (SEs)**, that are required for the decoding process. In **HEVC** the only entropy coding method considered to process the **SEs** is named the **CABAC**, that is overviewed by *Marpe et al.* in [MSW10] under the **AVC** context. *Sze and Budagavi* present in [SB12] an improved version of **CABAC** in the context of **HEVC**, that aims to overcome the throughput limitations of **CABAC**. The design of the **CABAC** can be decomposed into three key operations as depicted in Fig. 2.9: Binarization, Context Modeling and Arithmetic Coding.

Each **SE** is first binarized into symbols named bins if necessary, using different methods listed in [SB12] that depends of the **SE** type. Second, each bin may be context coded or bypass coded as for non-**MPM** Intra modes (except the first bin). In the case of context coding, the bin's value probability is estimated based on the previously coded bins. In the case of bypass coding, the probability is supposed to be 0.5. Whatever the probability, arithmetic coding is used in order to obtain the bitstream. If context coded is used, a feedback-loop updates the context for next probability estimations.

In previous compression standards, such as **AVC**, another entropy coding method named **Context-Adaptive Variable-Length Coding (CAVLC)** and based on **Variable-Length Coding**

(VLC) were also used. However, VLC limits the output code to an integer number of bits for each symbol. Arithmetic coding theoretically achieves floating number of bits for each symbol. Consequently, even if the CABAC entropy coder estimates an integer number of bits, the per-symbol estimation is closer to the actual entropy value, hence more accurate. Thus, CABAC significantly outperforms CAVLC and is the only considered method in HEVC.

## 2.7 Common Test Conditions

In order to allow fair comparison between video coding contributions, in the context of standardization, some Common Test Conditions (CTC) are provided along with each standard. The CTC are a set of requirements to meet, so that different coding tools can be compared. CTC are usually based on the common coding structures described in Section 2.1.2. In the case of algorithms that do not modify a standard, it is also recommended to follow CTC recommendations for fair comparison. In the context of HEVC, Bossen presents the CTC in [Bos13] that defines the following rules to follow:

- 24 Sequences spread into 6 classes (A, B, C, D, E and F). Each class represents a particular resolution (classA-E) or content type (classF is "screen content").
- Resolution, framerate, bitdepth and number of frames to encode are fixed for each sequence
- The Intra Period, i.e. the frequency of I-frame (Frame only composed of I-Slices) is also fixed and defined as a function of the framerate. See Table. 2.1 for the corresponding rules.
- Base QP are set to 22, 27, 32 and 37 for R-D curves achievement. Offsets between frames are also set based on the coding structure.

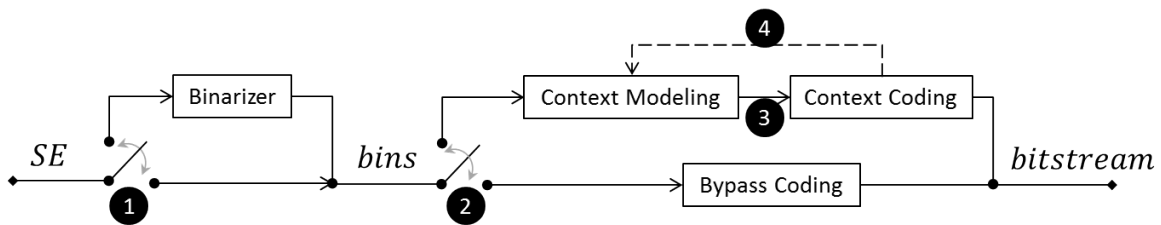


Fig. 2.9 Framework of CABAC. ❶:  $SE$ s may be binary before CABAC. ❷:  $SE$ s are coded in Bypass mode or Context mode. ❸: Bin's probability is transferred for arithmetic coding. ❹: Context is updated for probability estimation.

Table 2.1 I-Frame period based on Framerate as recommended by the CTC proposed by Bossen in [Bos13]

Framerate (frame/sec)	I-Period (in frames)
20	16
24	24
30	32
50	48
60	64

Experimental results presented in the following of this document follow the general CTC of HEVC presented above, if not mentioned otherwise.

## 2.8 Dependencies related to Hybrid Video Coding

In previous sections, we overview some of the coding tools supported by the HEVC standard. Several of these tools introduce dependencies, i.e. achievable coding efficiency may be dependent of previous coding decisions if not all. Because of complexity constraints, numerous encoding decisions are optimized locally and independently from each other. However, these dependencies suggest that global optimization can only be achieved by considering joint decisions. We depict in the following the various dependencies related to the hybrid video coding scheme.

### 2.8.1 Dependency related to samples prediction

Block-based hybrid video coding scheme like HEVC makes use of predictive coding and quantization, among other tools. Dependencies described in Section 1.4 are thus present in the context of HEVC and must be taken into account for global optimization purpose. However, the prediction in HEVC is either spatial (Intra coding) or temporal (Inter coding), resulting in spatial and temporal distortion propagation, respectively.

In case of Intra coding, the distortion is propagated from right and bottom samples borders of the reference block. This spatial distortion propagation is often compensated by trying to reduce the initial distortion made on reference samples. Indeed, the reconstructed pixels used for reference may be first filtered using a three-tap smoothing filter. However, filter makes more complex the modeling of the distortion propagation from PU to PU.

In case of Inter coding, the MCP is a copy-paste process with sub-pel accuracy and a given amount of residue transmitted after quantization. Consequently, an important part of

the reference samples distortion may be projected on the current PU. Under the assumption that distortion is uniformly distributed among pixels, the overlapping of MCP over several reference CU can easily be taken into account. After a frame encoding, the post-processing Deblocking filter may affect the border samples of each block. Because of it, temporal distortion propagation can be difficult to model. However, the Deblocking filter influence is often ignored to simplify the problem.

These two dependencies will be named the spatial distortion propagation and the temporal distortion propagation in the following chapters.

## 2.8.2 Dependency related to side informations prediction

As presented in Section 2.3, the Intra predictor may be transmitted using the MPM methods. MPM is a set of three spatial predictors based on the prediction modes used for PUs on the left and above of the current one. The signalization of the Intra predictor requires less bits using MPM than the bypass coding for the non-MPM modes. Consequently, re-using identical predictors is slightly favored. This is a supplementary dependency inherent to HEVC since the syntax cost of Intra predictor depends on the previous coding decisions.

A similar observation is made on the coding efficiency for motion information. The MV resulting from the ME is predicted from spatial or temporal neighboring MVs, in order to limit the transmitted motion data for MVD. Moreover, Merge and Skip modes efficiency rely on the motion homogeneity. Whatever the chosen motion vector for a given PU, it influences the coding efficiency of Inter coding modes for the spatially close PUs.

It must be taken into account that coding an Intra CU within a group of several Inter CUs breaks the motion homogeneity. Since an Intra PU does not carry any motion information, the *virtual* MV considered for predicting other PUs' MVs is set to zero motion vector. This broken homogeneity may result into reducing the coding efficiency of several PUs, i.e. the prediction quality of their MVs. Nevertheless, the predicted MVs competition proposed in HEVC is supposed to limit cases of inefficient MV prediction.

QP is another syntax element predicted based on the values used on neighboring CUs. In the context of AQ, only the delta quantizer is transmitted, i.e. the difference between current QP and a QP that is predicted based on available neighboring ones. It should be taken into account when designing such method that heterogeneous field of QPs may result into performance losses, because of the syntax cost overhead. This dependency is named the quantizer signalization dependency.

### 2.8.3 Dependency related to cross-processing

Individual CUs are processed sequentially using several coding tools, e.g. prediction, transformation and quantization. The sequential processing introduces a dependency between the different coding steps, i.e. the efficiency of one coding step depends on the result from a previously applied coding decision. An example of this dependency related to cross-processing and how it may be exploited is given as follow:

- We know transformation better compacts the energy of a signal, here the residue, with very little high frequencies
- We observe that the optimization of the transform coding obviously depends on the residue output of the prediction
- Thus, we assume that prediction decision that minimizes the residue variance intrinsically leads to minimize high frequencies. Consequently, it leads to highly efficient transformation

Dependency related to cross-processing increases exponentially the number of parameter combinations for one coding unit. Thus, the exhaustive joint optimization of coding steps for a single coding unit is highly complex. Some joint optimization of different coding steps have been considered by design in HEVC, examples:

- Transformed residue coefficients can be scanned in different order in case of Intra coding. This scan order is set based on the prediction angular direction and this simplification is named **Mode Dependent Coding Scan (MDCS)**
- **DST-VII** is used to replace **DCT-II** only if the prediction block of size 4x4 is Intra coded. This choice is due to the statistical properties of an Intra 4x4 residual block.
- Quantization matrices enabling frequency dependent scaling are allowed in HEVC. When default matrices are used, they depend on the Intra or Inter decision, the TU size and the color component
- etc.

All of these a priori constraints made in the standard saves substantial computational complexity by reducing the total number of considered combinatorics. However, non-normative constraints may be necessary in order to optimize coding decisions with reasonable complexity.

### 2.8.4 Dependency related to Entropic Coding

The last dependency is considered from a more global point of view. During the entropy coding procedure, each contextualized bin have a syntax cost that depends on its value probability. Arithmetic coding tends toward the entropy limit, and the entropy is the lowest when probability distribution is sharp. Hence, we conclude that the less variations is allowed in terms of [SE](#) values, the lower is the syntax cost. Such constraint limits the encoder versatility and may degrade residue coding efficiency, hence it should be considered with caution. Note that different context models can be used for different bins, the selected context being based on the type of syntax element, neighboring information, etc. Consequently, current coding parameters may also affect the context modeling of neighboring [CUs](#).

## Conclusion

In this Chapter, we described some of coding tools considered in the context of the [HEVC](#) standard. Thanks to this overview, we are able to identify dependencies between coding decisions, introduced by the standard constraints. Some dependencies are the distortion propagation over coding units, spatially or temporally, induced by the prediction process. As it is described in the following chapters, a large number of state of the art methods focused on these dependencies. Remaining dependencies are related to the cross-processing and the signalization of coding parameters, that uses prediction, context modeling and arithmetic coding. In the next chapter, we focus on the [HEVC](#) Intra coding and aim to estimate interests, in terms of coding efficiency, to take into account dependencies for [RDO](#).





# Chapter 3

## Coding dependencies in Intra coding

As shown in Chapter 2, several dependencies may affect the Intra coding efficiency in the context of HEVC. These dependencies are identified as the spatial distortion propagation, the MPM dependency and the entropy coding dependency. The dependency related to the signalization of the quantizer, i.e. the potential overhead for syntax element coding, exists only when the AQ is enabled. This latter is discussed in Section 3.4 since no AQ method is considered otherwise.

In this chapter, we consider the problem of global optimization in terms of rate-distortion performance taking into account these dependencies. Methods studied or proposed in the next sections focus on the consideration of the dependencies between coding units subject to an Intra-only coding scheme. The most common dependency to consider is the distortion propagation, that can be modeled or intuitively introduced within the RDO process. Most of state-of-the-art methods belong to this category of dependent optimization.

We first propose an experiment aiming to estimate the upper-bound coding efficiency that can be obtained when no distortion is made on reference samples used for Intra prediction, i.e. if there is no spatial distortion propagation. The HEVC encoder is modified in order to use the source samples for prediction, instead of the reconstructed samples. Obviously, the produced bitstream cannot be decoded due to the prediction mismatch between encoder and decoder. However, the distortion and bitrate can still be estimated during the RDO process. The resulting coding performance, i.e. the R-D couple, is stated as the maximum coding efficiency (upper-bound), assuming that nullify distortion onto reference is achievable and free of signalization overhead cost. BD-BR results of this experiment against the classical encoding in AI configuration are given in Table. 3.1, under the CTC. Five QPs are used to estimate gains: {22, 27, 32, 37, 42}.

It should be noted that the results presented in Table. 3.1 are not decodable. However, considering the large amount of bitrate savings, i.e.  $-25.20\%$  in average, we assume that

Table 3.1 Virtual YUV-BD-BR (PSNR) of using source as prediction reference against reconstructed in HEVC Test Model (HM)16.12.

	Maximum	Minimum	Average
Class A	-30.65%	-7.36%	-21.65%
Class B	-35.68%	-16.60%	-25.82%
Class C	-40.21%	-14.47%	-27.89%
Class D	-29.96%	-11.01%	-18.90%
Class E	-41.49%	-14.50%	-30.83%
All	<b>-41.19%</b>	<b>-7.36%</b>	<b>-25.20%</b>

significant R-D gains could be achieved by optimizing the distortion made onto samples used as reference. These results and conclusion were the basic justification for the work described in this Chapter. Despite the interest of considering spatial distortion propagation, we will explain later that other existing dependencies may also impact the global optimization, especially in the high-rate case. Some results of this chapter have been published in [BLTR<sup>+</sup>17] and [BLTR<sup>+</sup>18a].

The remainder of this chapter is organized as follows. Section 3.1 presents the state-of-the-art methods that consider the spatial distortion propagation during the RDO for achieving global optimization over the whole picture. Then, a study aiming to evaluate the maximum achievable coding efficiency, in the context of joint optimization, is presented in Section 3.2. We focus on jointly optimizing multiple prediction modes, related to different PUs. Section 3.3 depicts supplementary experiments and analysis to evaluate the opportunities of low complexity implementation and the respective impact of each dependency. Finally, the first method designed for prediction mode optimization in Section 3.2 is extended to the joint optimization of local quantizers in Section 3.4.

### 3.1 Previous Methods

In the context of a block-based coding scheme, the DCT coefficients quantization introduces error that is higher near the block boundaries, as explained by *Robertson and Stevenson* in [RS01]. The biased distortion distribution is also a justification for using the deblocking filter, as stated by *List et al.* in [LJL<sup>+</sup>03]. We point out that this statement is not limited to the Intra prediction and may also apply to the case of motion compensated prediction. Indeed, the correlation between two pixels is inversely relative to the distance between them. As the Intra prediction is basically a projection of neighbors pixels of the block, the prediction error in the current block naturally increases for pixels near the bottom-right corner, because they



Fig. 3.1 Spatial distribution of the prediction error  $\epsilon = (x - \hat{x})^2$  with  $x$  the source sample and  $\hat{x}$  the predicted sample, using diagonal-left predictor (mode number 18)

are less correlated with reference pixels. An illustration of this prediction error distribution is given in Fig. 3.1.

Consequently, the prediction error distribution within a block mixed with the quantization error property explain why the distortion is higher on the right and bottom boundaries of the block. Note that for a given quantizer, the distortion is bounded by the quantization error, as the reconstructed signal is added to the prediction error. Furthermore, because these pixels are further used as reference for the neighboring blocks prediction, an evident loss in terms of global optimality is exhibited. The HEVC standard may partially solve this issue by filtering the reference pixels before prediction in order to avoid generating wrong directional edges or structures to the predicted block. In next sections, we discuss other methods that have been proposed to overcome this issue.

In the following, we divide the related methods in two categories. The first one is stated *minimization of reference samples distortion*. Methods in this category consists into designing a solution that aims to compensate dependencies and reduce their impact on global coding efficiency. Instead of using models and hypotheses, such solutions are usually based on an exhaustive search of optimality or they modify local decisions in order to minimize distortion on samples further used as reference for prediction. The second category of methods is stated *distortion propagation modeling* and depicts the methods that model the dependencies and achieve global optimization through analytical solutions. These solutions use hypotheses and mathematical solving and usually maintain the computational complexity acceptable.

### 3.1.1 Minimization of reference samples distortion

HEVC standard already implements some filters specifically designed to minimize the impact of reference samples distortion. As mentioned above, a three-tap smoothing filter may be

applied to the reference samples before prediction. Note that the larger is the PU size, the more prediction modes use filtered reference samples. This type of smoothing is used to prevent the propagation of wrong directional edges. Adapting the filtering selectively based on the block size and prediction modes reduce contouring artifacts as stated by *Wien* in [Wie03]. Another smoothing process is applied on first row and/or column of prediction samples for DC, horizontal and vertical prediction directions. This solution enables 0.4% bitrate savings as reported by *Lainema et al.* in [LBH<sup>+</sup>12].

Another solution to minimize the distortion on block boundaries, that should highly improve the global efficiency of the encoding, is to use an adaptive quantization matrix. An equal expected-value rule is proposed by *Sullivan* in [Sul05], that aims to adapt the dead-zone parameter in order to keep the average error unchanged before and after quantization. Sullivan method enables up to 1dB gain in the Joint Model (AVC Reference Software) (JM) 8.6, that is the reference model of AVC. *Tanizawa and Chujoh* demonstrate the benefits of competing multiple quantization matrices for each MB in [TC06], and present bitrate savings in AVC up to 6.57%.

In [YHY09], *Yu et al.* propose to constraint the quantization error to be evenly distributed among all coefficients, in order to achieve evenly distributed distortion in the pixel domain. The proposed minimization problem is exposed in (3.1), with  $D_i$  and  $\sigma_i^2$  being the distortion and variance of the  $i^{\text{th}}$  coefficient from the 4x4 transformed residual block and  $D_T$  the target coefficient distortion.

$$\begin{aligned} \min_{D_1, \dots, D_{16}} \quad & \sum_i \ln \frac{\sigma_i^2}{D_i} \\ \text{subject to} \quad & D_i = D_T, \\ & D_i < \sigma_i^2, \end{aligned} \tag{3.1}$$

The problem of (3.1) is solved using sequential quadratic programming and integrated into a two-pass encoding. The first pass is used to determine optimal quantization step sizes per coefficient, and the second pass actually applies the quantization steps to the coefficients. Authors exhibit up to 12% bitrate savings in the context of AVC, but modifications of context modeling in CABAC are mentioned but not described. Such modifications change the decoding process and do not fall into the scope of this thesis.

Despite the obvious efficiency of these methods, they focus on balancing the distortion inside a block and minimizing the boundaries distortion, while no observation or hypotheses is made on how such distortion may propagate to other blocks. These methods assume that

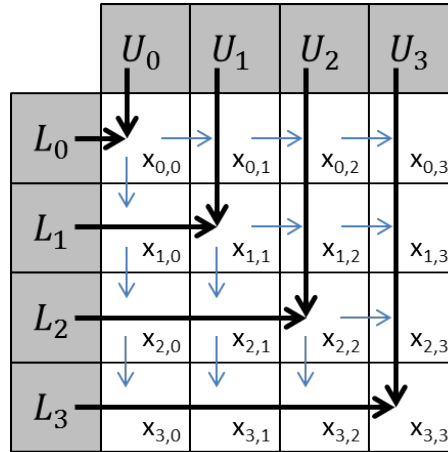


Fig. 3.2 Illustration of Weighted Cross Prediction

constraining the distortion distribution to be flat within the considered block reduces the spatial distortion propagation, but there is no guarantee that it is the case.

In the context of *AVC*, 16x16 Intra blocks may be predicted using the Planar mode, specifically designed to represent smoothly-varying regions. However, 4x4 blocks do not have access to such tool in *AVC*. In order to address the situation of smoothly-varying and small regions, i.e. 4x4 blocks, *Yu et al.* propose in [YGCZ08] to replace the DC prediction mode for such blocks. Their method is named the **Distance-based Weighting Prediction (DWP)** and consists in weighting the impact of reference pixels based on the inverse relative distance with the current pixel they try to predict.  $\hat{X}_{i,j}$  is the predicted pixel at position  $i, j$  in the block, with  $i$  the line and  $j$  the column.  $L_i$  is the left reference pixel on the  $i^{th}$  line and  $U_j$  is the upper reference pixel on the  $j^{th}$  column. (3.2) describes the proposed new mode of prediction.

$$\hat{X}_{i,j} = (L_i \times (i + 1) + U_j \times (j + 1) + 2) / (i + j + 2) \quad (3.2)$$

This approach does not consequently improve prediction of pixels lying in the top-left to bottom right diagonal of the block, but it has significant interest for pixel far from this diagonal, i.e. spatially closer to one reference block than the other. Average bitrate savings of 2.2% are enabled and a low-complexity version using integer approximation manages to enable 1.8% average bitrate savings.

In [WPU<sup>+</sup>09], *Wang et al.* design an alternative to the **DWP**, named the **Weighted Cross Prediction (WCP)**, by using a more complex linear combination of reference pixels to predict the current pixel. This model enables 0.6% average bitrate savings at a lowest complexity than the integer approximation of **DWP**. **WCP** is illustrated in Fig. 3.2.

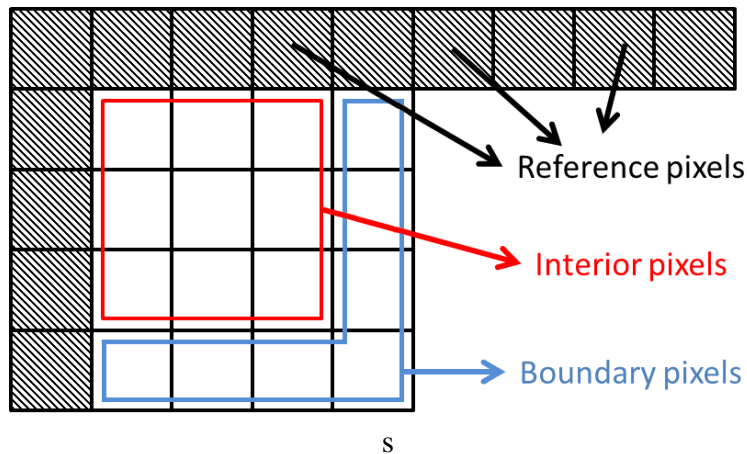


Fig. 3.3 Joint Line and Pixel Prediction: Illustration of the two groups pixels

Zhang *et al.* acknowledge in [ZMG10] the possible bad performance of line-based prediction for bottom-right pixels due to the poor correlation between those pixels and reference pixels. To tackle this issue, they introduce a new Intra prediction mode in AVC named **Joint Line and Pixel Prediction (JLPP)**. It consists in processing separately the right and bottom boundary pixels from others. The two groups of pixels are depicted in Fig. 3.3. When using JLPP, the block composed of interior pixels is predicted, transformed and quantized in similar way to AVC Intra coding. The remaining pixels are predicted one by one using surrounding pixels and they are applied spatial quantization only (no transformation). This additional mode enables 2.55% average bitrate savings but increases the computational complexity of encoding to 170%. However, authors state that a low complexity implementation at the encoder-side is possible and would require 60% of the anchor encoding time.

Methods presented in [YGCZ08, WPU<sup>+</sup>09, ZMG10], that are described above allow a better coding efficiency using additional prediction modes. These new tools compensate Intra prediction drawbacks, i.e. the non-uniform distortion distribution within a block. However, such method requires to change the decoding process and the standard, that is not the scope of this thesis.

In [YCJ08], You *et al.* manage to achieve bitrate reduction without changing the decoding part of AVC. The proposition focuses on the 4x4 prediction modes and consists in penalizing more aggressively, into the RDO, the distortion created on the pixels usable for reference. The classical R-D cost minimized for each 4x4 block is replaced by (3.3), with  $D_{bound}$  being the distortion on pixels usable as reference for further prediction. This solution is a multi-pass approach with tested values of  $\alpha$  in the set  $[0, 1, 2]$ . Even if the model enables 1.70% average

bitrate savings without modifying the standard, it implies important complexity increase due to the multi-passes.

$$J = D + \lambda \times R + \alpha \times D_{bound} \quad (3.3)$$

### 3.1.2 Distortion propagation modeling

Motivated by the increasing distortion on block boundaries, that may substantially decrease the Intra coding efficiency, *Pang et al.* introduce the **Inter-Block Dependency Model (IBDM)** in [PAZ<sup>+</sup>11] for the purpose of distortion redistribution over coefficients. This model makes use of the linear relationship between the residue variance in spatial domain and transformed domain, and also relies on the orthogonality principle presented in Section 1.3.2. The **IBDM** is defined in (3.4).

$$\sigma_C^2 = \mathbb{E} \left[ (X - \hat{X})^2 \right] \cdot A + D_{ref} \cdot A \quad (3.4)$$

$\sigma_C^2$  is the variance matrix of the coefficient residue in transformed domain,  $\cdot$  is matrix multiplication,  $X$  and  $\hat{X}$  are the source signal and its prediction value, respectively.  $D_{ref}$  is the average distortion made on the reference pixels (up and left) and  $A$  is a transform related matrix. The **IBDM** is experimentally verified and further estimated off-line as a linear function of  $D_{ref}$ , expressed in (3.5) with  $B$  and  $\sigma_{C0}^2$  being constant matrices.

$$\sigma_C^2 = B \cdot D_{ref} + \sigma_{C0}^2 \quad (3.5)$$

The minimization problem is finally expressed in (3.6), that is solved using an iterative approach.  $D_{i,j}^f$  is the coefficient distortion in transform domain at position  $(i, j)$  for a block of size  $N \times N$ .  $D_T$  is the target distortion, that is in the case of distortion redistribution problem, the distortion obtained after the first **RDO** iteration.

$$\begin{aligned} \min_{D^f} \quad & \sum_{i=1}^N \sum_{j=1}^N \log \left( \sigma_{i,j}^2 / D_{i,j}^f \right) \\ \text{s.t.} \quad & \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N D_{i,j}^f \leq D_T \end{aligned} \quad (3.6)$$

Note that the term to minimize,  $\log \left( \sigma_{i,j}^2 / D_{i,j}^f \right)$ , estimates the rate and is derived from the Shannon bound. Based on the reformulated **RDO**, authors in [PAZ<sup>+</sup>11] enable gains up to 2 dB in the reference model of **AVC**. However, 16x16 and 8x8 Intra coding modes were disabled and the experiments were only conducted on 4 sequences, which is insufficient to assess the performance of the method. *Pang et al.* reuse the **IBDM** for **Rate Control (RC)**



purpose in [PAZ<sup>+</sup>13b], in the context of **Audio-Video coding Standard (AVS)**. The solution is still iteratively solved.

*Sun et al.* in [SAD<sup>+</sup>12] state that the initial use of the **IBDM** only aims to redistribute the distortion within a block to avoid distortion propagation, and does not actually minimize the boundary distortion. They consequently modify the constraint in (3.6) and rewrite it as in (3.7).

$$\begin{aligned} \min_D \quad & \sum_{i=1}^N \sum_{j=1}^N \log \left( \sigma_{i,j}^2 / D_{i,j} \right) \\ \text{s.t.} \quad & \omega \left( \sum_{i=1}^N D_{i,N} + \sum_{j=1}^{N-1} D_{N,j} \right) + \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} D_{i,j} \leq D_T \end{aligned} \quad (3.7)$$

$\omega$  is a weighting factor used to penalize the distortion injected on the right and bottom boundaries. (3.7) is solved iteratively with  $\omega$  being updated at each step according to (3.8). The iteration stops when  $\omega$  is higher than a predetermined threshold.

$$\omega = \frac{\sum_{i=1}^N \sum_{j=1}^N D_{i,j}}{\sum_{i=1}^N D_{i,N} + \sum_{j=1}^{N-1} D_{N,j}} \quad (3.8)$$

The index  $j$  stops at  $N - 1$  in order to avoid adding  $D_{N,N}$  twice. *Sun et al.* implement this proposal for **HEVC** in the reference model **HM4.0** and it enables significant coding efficiency improvements. Unfortunately, no information is provided about the required number of iterations, or the computational complexity overhead.

*Qingbo et al.* propose in [QXB<sup>+</sup>14] a solution to address the problem of dependent optimization in **AVC** Intra coding. They use accurate lagrangian estimation and multiple lagrangian competition. Based on the Markov property of quantization errors, discussed by *Arnstein* in [Arn75], and the supposed optimality of future decisions, authors reduce the block distortion as a linear function of its neighbor distortion as expressed in (3.9).

$$D_{i+1} = aD_i + b \quad (3.9)$$

$a$  and  $b$  are defined as linear functions of the quantization value, that are optimized prior to the encoding with off-line training. Based on this dependency model, the  $\lambda$  computation is expressed in (3.10).

$$\lambda^* = \frac{\frac{d[D_i + D_{i+1}]}{d\Delta}}{\frac{d[R_i + R_{i+1}]}{d\Delta}} \quad (3.10)$$

In order to make the model more robust to image contents variation, a framework named **Multiple Lagrangian Multiplier (MLM)** is proposed. The concept is to optimize coding decision according to each proposed lagrangian value. The resulting **R-D** costs can further be compared in order to extract the coding decision that achieves global optimization, in the limits of the proposed dependency model. The considered lagrangian values correspond to different  $(a, b)$  couples trained to match different source content characteristics. When considering one lagrangian, the model enables 0.46% average bitrate savings, in the context of **AVC** for low-bitrate test case. This result increases to 0.99% and 1.15% using **MLM** framework when putting in competition 2 and 4 lagrangian values, respectively.

### Addressing the dependency related to Entropic Coding

Most of the studies cited before focus on improving the reconstructed quality of reference pixels in order to compensate the potential distortion propagation. This popularity is consistent with the obvious dependency inherent to the Intra coding scheme in MPEG standard. However, some papers also focus on a less obvious dependency related to the **CABAC**. The study we present below differs from the previous categories, by the considered dependency, and is consequently presented apart.

*Im et al.* modify the rate estimation made by **CABAC** in [IGL12], in order to take into account the non-integer bits  $f$  that are shared between consecutive groups of bins. The modified **R-D** cost is defined in (3.11). Authors also extended this concept to **HEVC** in [IGC15]. By avoiding an integer approximation of the rate, such solution enable **BD-BR** improvements of  $-0.82\%$  and  $-2.75\%$  for Intra coding only of **AVC** and **HEVC**, respectively. Note that in [IGC15] more important gains are enabled for IPPPP and IBBBP **GOP** structures.

$$J = D + \lambda \times (R + f) \quad (3.11)$$

This method has the interest of addressing the **CABAC** dependency and thus optimizing the coding procedure at the frame-level. It is consequently more efficient than a joint optimization on an area smaller than a frame. Nevertheless, it cannot address other cited dependencies, such as the spatial distortion propagation or the **MPM** dependency.

## 3.2 Inter-Block Dependencies Consideration for Intra Coding Optimization

The studies cited in Section 3.1 are **JRDO** methods that improve the coding efficiency when compared to *Independent-RDO* approach. Improvements are achieved by formalizing

dependencies in theoretical models or exhaustively searching for global optimality, under constraints. However, such models are often using coarse assumptions resulting in simplified dependency models. To the best of our knowledge, there is no reference proving the maximum achievable gain of a coding decision model that considers intrinsic inter-block dependencies.

In this section we aim to make a quantitative evaluation of the benefits of **JRDO** methods for Intra coding. Specifically, we focus on considering joint optimization of multiple blocks, e.g. **CUs** or **PUs**. Based on the complexity formalization of exhaustive **JRDO** methods exposed in Section 1.5, we limit the search space of the solution to a reasonable use case in order to estimate the maximum achievable coding efficiency.

We propose to evaluate the maximum achievable gain of exhaustive joint optimization of multiple **CUs** applied to intra prediction mode decision. From dependencies identified in Section 2.8, we address the spatial distortion propagation, the **MPM** dependency and the entropy coding dependency. Since the latest **MPEG** compression standards are based on similar concepts, and thus similar dependencies, we chose to confirm our approach in both **HEVC** and **AVC** standards. We introduce two **JRDO** models in Section 3.2.1: *Dual-JRDO* and *Quad-JRDO*. Experimental results and bitrate savings of the proposed **JRDO** approaches are presented and discussed in Section 3.2.2 for both **AVC** and **HEVC**.

### 3.2.1 Proposed JRDO models

Once the inter-block dependencies are defined, we consider  $J_i(\vec{p}_i)$ , the **R-D** cost of  $CU_i$  knowing  $\vec{p}_i$ , if no dependencies interfere, either in terms of distortion or **CABAC**. We define  $\Delta J_i(\vec{p}_{ref}, \vec{p}_i)$  as the intra propagation cost, with  $\vec{p}_{ref}$  representing the coding decisions of previous **CUs** that may affect  $CU_i$ . The generic **JRDO** equation for  $CU_i$  **R-D** cost is then formalized as (3.12).

$$J_i(\vec{p}_{ref}, \vec{p}_i) = J_i(\vec{p}_i) + \Delta J_i(\vec{p}_{ref}, \vec{p}_i) \quad (3.12)$$

Studies cited in Section 3.1.2 have tried to model  $\Delta J_i(\vec{p}_{ref}, \vec{p}_i)$  by simply considering the spatial distortion propagation. In order to simplify implementations and keep the computational complexity reasonably low for the study, the joint optimization is limited to intra prediction modes. Hence, we focus on **PUs** and the vector of coding parameters  $\vec{p}_i$  is a 1-D vector that designates the spatial predictor.

For a **PU** to encode with a **JRDO** approach, the difficulty consists in how to consider neighboring **PUs**: either in terms of spatial distance, coding order, or both. Fig. 3.4 shows an example of how the dependency may affect a given **PU**. If we focus on the dependencies that may affect the  $PU_7$ , we see that the prediction may refer to samples within  $PU_1$ ,  $PU_4$

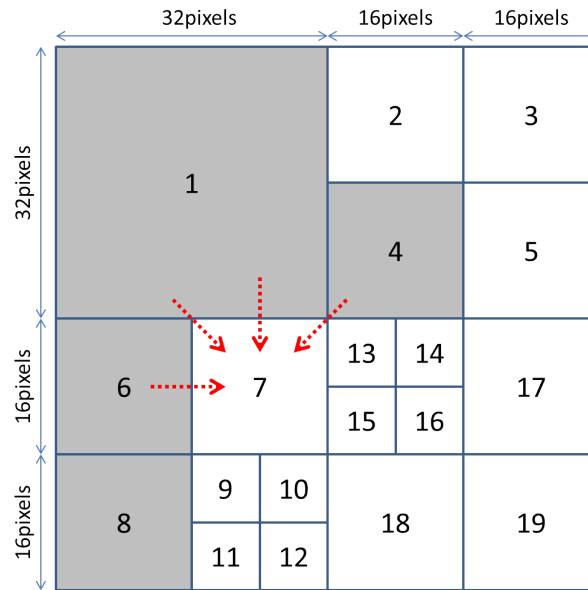


Fig. 3.4 QuadTree possible partitioning and related spatial dependencies illustration

and  $PU_6$ . Theoretically, some spatial predictors available may refer to samples from  $PU_8$ , that is not encoded at the stage of estimating coding decision for  $PU_7$ . In such case, the reconstructed pixels from the closest PU (here  $PU_6$ ) are projected to replace the missing ones. Moreover, the MPM coding solution used in HEVC introduces additional dependencies with  $PU_1$  and  $PU_6$  to code the spatial predictor of  $PU_7$ . Jointly optimizing the considered PUs should significantly improve the coding efficiency of the spatial area. Nevertheless, compression is applied sequentially and  $PU_2$ ,  $PU_3$  and  $PU_5$  have to be compressed first, in order to have the correct coding context when deciding and coding  $PU_7$ .

This observation highlights the difficulty of implementing exhaustive JRDO approaches in intra coding: on the one hand jointly optimizing many PUs is highly complex in terms of computation resources, and on the other hand ignoring three PUs leads to strong approximations on coding context. Models proposed in the following avoid these difficulties and apply an exhaustive joint optimization on the current PU and neighboring PUs taking into account both spatial and coding order distances.

### Dual-JRDO

Z-scan is the coding order used to sequentially encode the four sub-PUs that form the splitting decision. Based on Z-scan and local dependencies, one predicts that  $PU_{i+1}$  is highly dependent of  $PU_i$  if  $PU_{i+1}$  is the spatial right neighbor of  $PU_i$ .

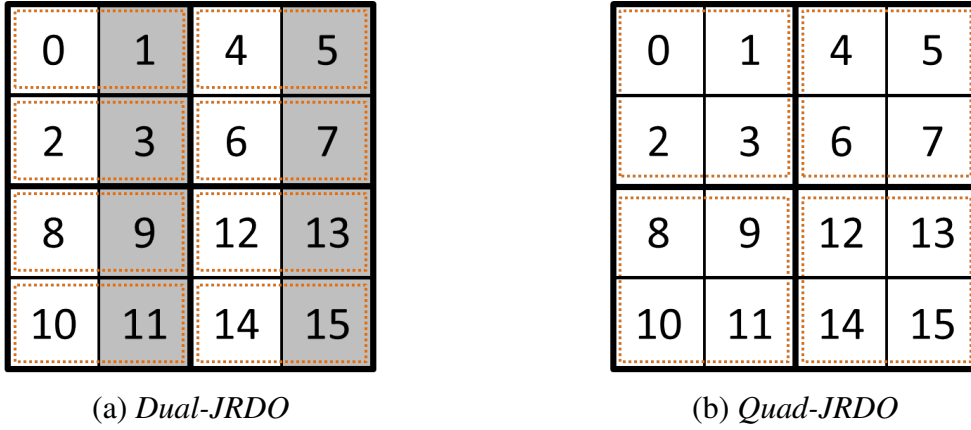


Fig. 3.5 Example of *Dual-JRDO* and *Quad-JRDO*

To confirm this, we experimentally verified that the right neighbor **PU** is often the most dependent on the current **PU** to encode. This experimental verification is described after the description of the current method. Therefore, we propose the *Dual-JRDO* model that jointly optimizes the intra prediction mode of each **PU** with the prediction mode of its right neighbor. In order to avoid wrong syntax context states, *Dual-JRDO* handles two cases:

- If  $PU_i$  right neighbor is  $PU_{i+1}$ , apply the *Dual-JRDO*.
- Otherwise, *Independent-RDO* is applied on the current  $PU_i$ .

With  $\vec{p}_i^*$  is the selected predictor to encode  $PU_i$  and  $\vec{p}'_{i+1}$  is the estimated optimal coding mode for  $PU_{i+1}$ , *Dual-JRDO* solution is expressed by (3.13).

$$\{\vec{p}_i^*, \vec{p}'_{i+1}\} = \arg \min_{\{\vec{p}_i, \vec{p}_{i+1}\}} \{J_i(\vec{p}_i) + J_{i+1}(\vec{p}_i, \vec{p}_{i+1})\} \quad (3.13)$$

In **HEVC**, the neighboring **PU**s can be further split, leading to  $\vec{p}'_{i+1} \neq \vec{p}_{i+1}^*$ . To overcome this problem, one considers that two **PU**s coming from the same split process have a high probability to have the same final partition size. Statistically, we can notice that the probability of this assumption to be true increases as **PU** size decreases. An example of *Dual-JRDO* is shown in Figure 3.5 (a), with dotted lines delimiting the optimization area and dark gray area refers to block coded independently using *Independent-RDO*.

In *Dual-JRDO*,  $K^2$  possibilities are explored for half of the **PU**s and  $K$  for the other half, with  $K$  the number of Intra prediction modes equal to 35 in **HEVC**. Concerning **AVC**,  $K = 4$  for 16x16 blocks and  $K = 9$  for others. In the case of *Independent-RDO*,  $K$  possibilities are explored for all **PU**s. We deduce that in this particular case *Dual-JRDO* multiplies the complexity of *Independent-RDO* by  $(K + 1)/2$ .

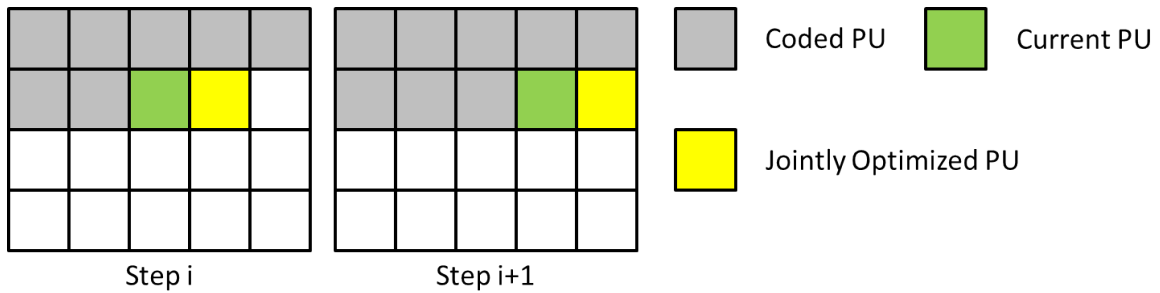


Fig. 3.6 Proposed coding scheme for optimizing jointly a PU with its right neighbor.

In *AVC*, block partitioning necessarily splits blocks into partitions of the same size, thus we have  $\vec{p}'_{i+1} = \vec{p}^*_{i+1}$  leading to complexity reduction. Results of *Dual-JRDO* for both *AVC* and *HEVC* standards are presented in Section 3.2.2.

#### Experiments on joint optimization: use the right or the bottom neighbor?

Due to the coding order of CUs and PUs in *HEVC*, spatially neighboring units may not be the next to encode. However, these units may have strong dependencies with the current one, especially in terms of spatial distortion propagation. One assumption made when we design the *Dual-JRDO* was that dependency with right neighbor is more important than dependency with the bottom neighbor. A simplified coding scheme is first proposed in order to estimate the correctness of such assumption:

- Disable the QuadTree partitioning
- Fix CTU size and constraint the PU to be coded in 2Nx2N mode
- Fix TU size

In this simplified coding scheme, PUs are coded in raster scan order and we propose the two coding schemes described in Fig. 3.6 and Fig. 3.7. The current PU is optimized jointly using the *Dual-JRDO* with either its right neighbor (Fig. 3.6) or its bottom neighbor (Fig. 3.7). In the case of joint optimization with the bottom neighbor, several PUs are required to be coded during the exhaustive analysis of *Dual-JRDO*. In order to avoid optimizing a large number of PUs jointly, these PUs are considered as bypassed and are coded with *Independent-RDO* method. The procedure is summarized as follow:

1. Test the next prediction mode for the current PU
2. Apply *Independent-RDO* on all bypassed PUs

3. Apply *Independent-RDO* on the bottom PU, save the total R-D cost for all analyzed PUs and go to step 1

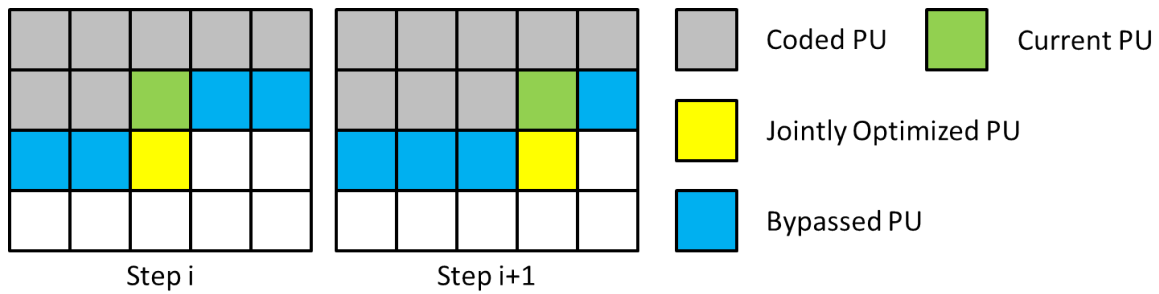


Fig. 3.7 Proposed coding scheme for optimizing jointly a PU with its bottom neighbor.

The prediction mode of current PU that leads to the minimal total R-D cost is set and the next PU is analyzed. The coding efficiency of each method are reported in Table. 3.2 for various size of CU/PU and TU. The first observation is that optimizing jointly the right neighbor is always more beneficial than the bottom one. The second observation is that when both neighboring PUs are jointly optimized, the resulting gains are comparable to the one obtained with only optimizing jointly the current and right PU. However, we acknowledge that the conclusion may be debatable, since the impact from all bypassed PUs is not considered.

Similar experiment have been processed in order to choose not applying the *Dual-JRDO* when the two neighboring PUs do not follow each other in coding order. The results shows the coding efficiency improvement to be negligible, when always applying *Dual-JRDO*, compared to the proposal.

Table 3.2 Average Y-BD-BR (PSNR) depending of which neighboring PU is jointly optimized in *HM16.12 Dual-JRDO*.

CU Size	TU Size	Right PU Optimized	Bottom PU Optimized	Both PUs Optimized
16x16	16x16	-0.56%	-0.04%	-0.56%
	8x8	-0.79%	-0.01%	-0.79%
	4x4	-1.84%	-0.65%	-1.84%
8x8	8x8	-1.80%	-0.21%	-1.99%
	4x4	-2.90%	-1.00%	-3.23%

### Quad-JRDO

We can see from Figure 2.7 that numerous spatial predictors exploit vertical spatial correlations. By definition, *Dual-JRDO* does not consider distortion propagated vertically, because the distortion of  $PU_i$  bottom samples is not propagated to its right neighbor. *Quad-JRDO* proposes to include vertical neighbors of  $PU_i$  in the joint optimization process.

*Quad-JRDO* optimizes all sub-**PU**s coming from the same split operation. At **CTU** level, raster scan order imposes to code the whole line of **PU**s before reaching bottom neighbors of  $PU_i$ . This results in unachievable computational complexity or wrong syntax context states, reason why *Quad-JRDO* is not applied at **CTU** nor **MB** level. The optimization formulation of *Quad-JRDO* is given in (3.14).

$$\{\bar{p}_k^*\}_{k=i}^{i+3} = \arg \min_{\{\bar{p}_k\}_{k=i}^{i+3}} \sum_{k=i}^{i+3} \left\{ J_k \left( \{\bar{p}_l\}_{l=i}^k \right) \right\} \quad (3.14)$$

*Quad-JRDO* supposes that all sub-**PU**s are not further split, which is not matching the *QuadTree* structure in **HEVC**. To overcome this limitation, (3.14) is only applied to the special case of  $N \times N$  mode, other cases use the *Independent-RDO*. An example of *Quad-JRDO* is shown in Figure 3.5 (b). The use of *Quad-JRDO* for  $N \times N$  analysis multiplies the complexity of *Independent-RDO* by  $K^3/4$ .

In **AVC**, **MB**s are always split in sub-partitions of same size, either 8x8 or 4x4 partitions. In the case of 8x8 partition mode, coding parameters of the four 8x8 blocks are optimized jointly. In the case of 4x4 partition mode, each 8x8 block jointly optimizes the four 4x4 sub-blocks. Experimental configurations and **R-D** results for both standards are presented and discussed in Section 3.2.2.

## 3.2.2 Experiments

The two methods exposed in Section 3.2.1 have been implemented into **HEVC** and **AVC** reference test models, *HM16.6* [MRB<sup>+</sup>14] and *JM19.0* [TLSS09] respectively. The set of sequences utilized is picked among **JCT-VC** test set presented by *Bossen* in [Bos13]. Furthermore, because of the computational complexity of our methods, the encoding is restricted to the first frame of each sequence, in All-Intra configuration. One remembers that neither tested nor referenced schemes are restricted, especially in terms of *QuadTree* partitioning. *Dual-JRDO* and *Quad-JRDO* schemes assumes neighboring **PU**s to be of same size than current one for optimization, not constraining their final size.

Tables 3.3 and 3.4 present coding efficiency improvements of *Dual-JRDO* and *Quad-JRDO* against *Independent-RDO* (Anchors). Results use *Bjontegaard* metric [Bjo01a] and



are expressed in **BD-BR**, i.e. the percentage of bitrate savings to achieve similar distortion, measured as frame **PSNR**. Even if the initial metric is expressed using 4 different **QP** values, we use it with 5 **QP** values ( $QP \in \{22, 27, 32, 37, 42\}$ ) to cover a larger range of bitrates. Besides, since the proposed solutions are mainly used to optimize luminance (Y) encoding, we focus on Y **BD-BR**, nevertheless, similar gains have been obtained in YUV 4:2:0.

### Results on *Dual-JRDO*

Results of *Dual-JRDO* are depicted in Table. 3.3. We observe constant gains against *Independent-RDO*. Average bitrate savings are of  $-0.77\%$  and of  $-0.71\%$  in *JM19.0* and *HM16.6*, respectively. *Dual-JRDO* outperforms *Independent-RDO* up to more than  $-1.3\%$  in both reference softwares. However, one observes that the *BasketballPass* sequence in *JM19.0* is the only one to present negligible losses. *Dual-JRDO* slightly favors horizontal predictions. In few cases where vertical prediction is better than horizontal, *Dual-JRDO* can slightly penalize coding efficiency.

The results presented in [QXB<sup>+</sup>14] come from two separate solutions, the first contribution is related to a **JRDO** approach and the second contribution to a **MLM** framework. In their **JRDO** approach, similar dependencies as *Dual-JRDO* are considered and gains announced for video of 1920x1080 resolution are about  $-0.13\%$ . Our study on identical test set shows that achievable gains are on average  $-0.80\%$  for **AVC**. They estimate the distortion dependency with an off-line linear distortion propagation model, and analytically deduct the related optimal  $\lambda$ . Our exhaustive joint prediction optimization demonstrates that there is room for improvement by modeling dependencies.

### Results on *Quad-JRDO*

In the case of the *Quad-JRDO* model presented in Section 3.2.1, the optimization is applied only to 4x4 and 8x8 blocks in *JM19.0*, and NxN case in *HM16.6*. The remaining decisions are based on *Independent-RDO*. Results for both implementations are presented in Table 3.4.

In [YCJ08], You *et al.* reports bitrate savings of  $1.70\%$  in the *JM*, that are comparable with the results of *Quad-JRDO*. Their optimization proposal is coarser than the proposed exhaustive search, but indirectly optimize an area of 16 blocks, or **PU**. We believe that, as shown by comparing *Dual-JRDO* and *Quad-JRDO* performance, larger area of optimization allow larger gains. But the computational complexity prevent an exhaustive verification of this statement.

As expected, much higher gains are observed with this second model, which is also much more complex. In average, bitrate savings over *Independent-RDO* are  $-1.78\%$  in *JM19.0* and

Table 3.3 Y BD-Rate of *Dual-JRDO* in *JM19.0* and *HM16.6*

Test sequences		<i>JM19.0</i>	<i>HM16.6</i>
1920x1080	Kimono	-1.01%	-0.21%
	ParkScene	-0.68%	-0.48%
	Cactus	-0.80%	-0.62%
	BQTerrace	-0.58%	-0.69%
	BasketballDrive	-0.93%	-0.47%
	<b>Average</b>	<b>-0.80%</b>	<b>-0.49%</b>
1280x720	FourPeople	-0.77%	-0.68%
	Johnny	-0.94%	-0.41%
	KristenAndSara	-0.96%	-0.47%
	<b>Average</b>	<b>-0.89%</b>	<b>-0.52%</b>
832x480	RaceHorses	-0.57%	-0.50%
	BQMall	-0.75%	-0.89%
	PartyScene	-0.46%	-0.88%
	BasketballDrill	-1.37%	-1.31%
	<b>Average</b>	<b>-0.89%</b>	<b>-0.90%</b>
416x240	RaceHorses	-0.67%	-0.98%
	BQSquare	-0.73%	-1.10%
	BlowingBubbles	-0.66%	-0.61%
	BasketballPass	0.08%	-1.02%
	<b>Average</b>	<b>-0.50%</b>	<b>-0.93%</b>
All	<b>Average</b>	<b>-0.77%</b>	<b>-0.71%</b>
	<b>Maximum</b>	<b>-1.37%</b>	<b>-1.31%</b>
	<b>Minimum</b>	<b>0.08%</b>	<b>-0.21%</b>

–1.47% in *HM16.6*. *BasketballPass* sequence, for which negligible losses were observed in *Dual-JRDO*, now outperforms *Independent-RDO* from –1.09% in *Quad-JRDO*. Besides, we must note that the *Quad JRDO* is less efficient than *Dual JRDO* for some high resolution sequences such as *Kimono* and *BasketballDrive*. One explanation is that some HD sequences may have more homogeneous areas, where larger partitions are preferred for the prediction; the joint optimization of  $N \times N$  (i.e.  $4 \times 4$ ) partition mode is then of limited interest for these particular cases.

The significant coding efficiency improvement between *Dual-JRDO* and *Quad-JRDO* mostly comes from the consideration of vertical predictions and 2-D spatial dependency. Based on these results, it seems relevant to assume that adding more PUs in the proposed joint optimization process, would bring much more gain. One could expect to tend toward *global-*

Table 3.4 Y BD-Rate of *Quad-JRDO* in *JM19.0* and *HM16.6*

Test sequences		<i>JM19.0</i>	<i>HM16.6</i> (NxN)
1920x1080	Kimono	-2.53%	-0.04%
	ParkScene	-1.60%	-1.00%
	Cactus	-1.91%	-1.33%
	BQTerrace	-1.37%	-1.34%
	BasketballDrive	-1.79%	-0.24%
	<b>Average</b>	<b>-1.84%</b>	<b>-0.79%</b>
1280x720	FourPeople	-2.04%	-1.40%
	Johnny	-1.86%	-1.07%
	KristenAndSara	-1.76%	-1.53%
	<b>Average</b>	<b>-1.89%</b>	<b>-1.33%</b>
832x480	RaceHorses	-1.39%	-1.38%
	BQMall	-1.69%	-1.95%
	PartyScene	-1.38%	-1.94%
	BasketballDrill	-3.10%	-2.31%
	<b>Average</b>	<b>-1.89%</b>	<b>-1.90%</b>
416x240	RaceHorses	-1.67%	-2.04%
	BQSquare	-1.57%	-2.19%
	BlowingBubbles	-1.69%	-1.60%
	BasketballPass	-1.09%	-2.08%
	<b>Average</b>	<b>-1.51%</b>	<b>-1.98%</b>
All	<b>Average</b>	<b>-1.78%</b>	<b>-1.47%</b>
	<b>Maximum</b>	<b>-3.10%</b>	<b>-2.31%</b>
	<b>Minimum</b>	<b>-1.09%</b>	<b>-0.04%</b>

*RDO* efficiency. In practice, the complexity of such process would lead to computationally intractable simulations.

In the next section, we focus on the *Dual-JRDO* implementation in the scope of *HEVC*. Using the *Dual-JRDO* presented in Section 3.2.1 as a starting point, we evaluate the opportunities for reducing the *Dual-JRDO* computational complexity.

### 3.3 Low Complexity JRDO of Prediction Units Couples for HEVC Intra Coding

In the previous section, we made explicit the dependencies related to the *HEVC* Intra coding and proposed to achieve *JRDO* on group of 2 *PU*s (*Dual-JRDO*) or 4 *PU*s (*Quad-*

Table 3.5 The *Dual-JRDO* gain against *Independent-RDO* for each depth

PU size	64x64	32x32	16x16	8x8	4x4	All
Average	0.00%	0.00%	-0.06%	-0.11%	-0.46%	<b>-0.63%</b>
Maximum	0.00%	-0.08%	-0.28%	-0.19%	-1.02%	<b>-1.12%</b>
Minimum	0.00%	+0.23%	+0.04%	+0.05%	-0.01%	<b>-0.19%</b>

*JRDO*). Original models bring systematic bitrate savings for similar quality, but suffer from a significant computational complexity increase. In the following, we focus on the *Dual-JRDO* case and propose to optimize it in order to achieve coding gains with a more acceptable computational complexity. We introduce the three acceleration solutions for *Dual-JRDO* in Section 3.3.1. Experimental results of the proposed *Fast Dual-JRDO* model are presented and discussed in Section 3.3.2.

### 3.3.1 Acceleration Methods

In this section, three methods are introduced in order to reduce the computational complexity  $C_{px}$  of the *Dual-JRDO*.  $C_{px}$  increase is directly related to  $N_b$ , the number of analyzed PUs (of all size) in a frame during *RDO*. In the case of a single CTU of size 64x64,  $N_b = 341$ : 1 PU 64x64, 4 PUs 32x32, 16 PUs 16x16, 64 PUs 8x8 and 256 PUs 4x4. We note that  $N_b$  is composed of 93.84% of PUs with size of 4x4 and 8x8.

Additional experiments, illustrated in Table 3.5, assess the *BD-BR* gain of *Dual-JRDO* independently brought by each PU size (or specific depth). Test conditions are the same as described in Section 3.3.2. These results show that the *R-D* gains introduced by *Dual-JRDO* are mostly brought by coding efficiency improvement of 4x4 and 8x8 PUs. We also point out that applying *Dual-JRDO* on these PU sizes carries a large portion of the computational complexity increase.

In the following, one focuses on accelerating *Dual-JRDO* applied to 4x4 and 8x8 PUs because of their high complexity. In practice, 64x64 PU size rarely appear to be optimal, even in the *Independent-RDO* case, which explains that no gains are observed here.

#### Adapting to Spatial Activity

QuadTree is responsible for the large  $C_{px}$  endured by HEVC [MAH<sup>+</sup>17]. Since Intra coding favors large PU size for smooth areas and small PU size for textured areas, many fast algorithms estimate the spatial activity of the source to adaptively skip *RDO* for some PUs. In [TG12], Tian and Goto propose to apply thresholds to down-sampled blocks' variance in

order to eliminate large PU sizes for textured content and small PU sizes for homogeneous content. *Shi et al.* reuse the same measure in [SAZ<sup>+</sup>13] in order to estimate with confidence if the neighboring PUs size are suitable for the current PU. Finally, such local activity-oriented metric can also feed a more complex learning tool such as Support Vector Machine (SVM), as proposed by *Liu et al.* in [LCFC15].

Due to its proven efficiency, we propose to rely on a spatial activity measure similar to the one defined by *Tian et al.* in [TG12], for adaptive use of *Dual-JRDO*. In the case of high activity, the distortion is likely to be important on block boundaries. The spatial predictors may also fall into local optimums, leading to an heterogeneous field of Intra prediction vectors. In such situations, *Independent-RDO* is supposed to be sub-optimal and thus we activate *Dual-JRDO*.

As a good trade-off between metric computational overhead and estimator accuracy, the spatial activity is computed over 16x16 pixel area. Furthermore, in order to be more robust to random noise, the computation is done on a 16x16 block down-sampled to 4x4. Note that sub-blocks (16x8 or 8x4) share the same spatial activity value which is computed from the corresponding upper 16x16 size bloc, as described below:

1. Down-sample all 16x16 blocks into 4x4 blocks, then compute spatial activity  $g_i$  as defined in (3.15)
2. Each PU of 4x4 and 8x8 size is assigned with  $g_i$  value of the corresponding 16x16 PU they belong to
3. If  $g_i \geq Th$ ,  $PU_i$  is processed with *Dual-JRDO*

$$g_i = \frac{1}{16} \sum_{x=0}^3 \sum_{y=0}^3 \min \begin{cases} |I_i(x,y) - I_i(x-1,y)| \\ |I_i(x,y) - I_i(x,y-1)| \end{cases} \quad (3.15)$$

$I_i(x,y)$  is the pixel luminance value at relative position  $(x,y)$  of down-sampled 16x16  $PU_i$ . In order to exclude neighboring PUs energy, we set  $I_i(-1,y) = I_i(x,-1) = 0$ .

$Th$  is a predefined threshold computed off-line with a supervised learning using logistic regression as proposed by *le Cessie and van Houwelingen* in [ICvH92]. *Dual-JRDO* estimation encloses *Independent-RDO* estimation by design, hence we can a posteriori observe if *Dual-JRDO* was of interest.

Supervised learning is used to estimate the optimal threshold for five QP values:  $QP \in \{22, 27, 32, 37, 42\}$ . The relationship between QP and  $Th$  is then obtained through logistic

least square method applied on the previously obtained  $(QP, Th)$  couples as expressed in (3.16).

$$Th(QP) = \alpha \times e^{\beta \times QP} \quad (3.16)$$

$(\alpha, \beta)$  values are equal to  $(0.0963, 0.107)$  in our case.

During threshold learning, we observe that the classification is more efficient for high  $QP$  values. One possible explanation is that for low rates, the distortion  $D$  tends to be equal to the prediction error. Consequently, since the spatial activity is a coarse estimation of the difficulty to predict a couple of PUs, it becomes at low rates a better predictor of the need to activate *Dual-JRDO*.

### Short Listing on $PU_{i+1}$

In the HEVC reference Model HM16.12 [MRB<sup>+</sup>14] used in our experiments, *Independent-RDO* for Intra coding is only applied on a shortlist of modes created by *Rough Mode Decision (RMD)* algorithm, that is described by *Lainema et al.* in [LBH<sup>+</sup>12]. RMD consists in short-listing prediction modes with the lowest residual *Hadamard Transformed Sum of Absolute Difference (SATD)* values plus the approximated predictor syntax cost. Only this short-list is then estimated through RDO. The minimal number of modes to be considered in RDO is respectively set to 8 for 4x4 and 8x8 PUs, and 3 for larger PU sizes. If MPMs or some of them are missing from the short list they are added to it. Hence, the maximum number of modes estimated through RDO is 11 for small PUs and 6 for large PUs.

The set of coding modes to consider (i.e. the possible values of  $\vec{p}_i$ ) is denoted  $M_i$ . In *Dual-JRDO*,  $M_{i+1}$  set consists of 35 intra prediction modes defined in HEVC. Since  $\vec{p}_{i+1}^*$  is necessarily subject to the RMD process because of *Independent-RDO* implementation in the test model, it is relevant to also construct the  $M_{i+1}$  list based on the RMD optimization. Residue, syntax mode cost and MPMs being all dependent of  $\vec{p}_i$ , we denote as  $M_{i+1}(\vec{p}_i)$  the set of modes to be considered for  $PU_{i+1}$  while optimizing  $PU_i$ . Therefore, (3.13) becomes (3.17).

$$\vec{p}_i^* = \arg \min_{\vec{p}_i \in M_i} \left\{ J_i(\vec{p}_i) + \min_{\vec{p}_{i+1} \in M_{i+1}(\vec{p}_i)} \{ J_{i+1}(\vec{p}_i, \vec{p}_{i+1}) \} \right\} \quad (3.17)$$

Note that RMD is inherited from the RDO implementation in the HM16.12. However, any conceivable short-listing approach efficient for HEVC Intra coding with the *Independent-RDO* model, could also be beneficial for the *Dual-JRDO* model.

### Prediction Modes Clustering based on Residual Analysis

Several dependencies between PUs have been exhibited in Section 2.8: the spatial distortion propagation, the MPM dependency and the entropy coding dependency (CABAC). The CABAC dependency is considered negligible in *Dual-JRDO* coding scheme since both  $PU_i$  and  $PU_{i+1}$  are subject to very similar contexts. It is equivalent to assume that bin probabilities are unlikely to vary with an important degree between two successive PU codings. Basically, we assume the most influential dependencies affecting  $PU_{i+1}$  are the spatial distortion propagation and the MPM dependency from  $PU_i$ .

Two prediction modes which result into identical residual signal should also result in identical reconstructed signal. This assertion is true if no divergent process impacts the coding of residual. It implies identical transformation and quantization steps for HEVC Intra coding.

MDCS is a technique implemented in HEVC Intra coding and presented by *Sole et al.* in [SJM<sup>+</sup>12b] that does not fulfill the requirement of no mode-dependent process on residuals. However, we ignore the minor difference of process attributed to MDCS since it has a slight impact on the proposed solution efficiency.

By considering only distortion dependency and the correlation between prediction residual and reconstructed data, we suppose that two modes of  $PU_i$  resulting in the same residual data share the exact same impact on  $PU_{i+1}$ . We define as a *cluster* a set of prediction modes which result into identical residual signal. Let  $\vec{p}_{i1}$  and  $\vec{p}_{i2}$ , two coding parameters of  $PU_i$  which result into the same prediction residual. Under the previous statement, equality (3.18) holds.

$$\min_{\vec{p}_{i+1}} \{J_{i+1}(\vec{p}_{i1}, \vec{p}_{i+1})\} = \min_{\vec{p}_{i+1}} \{J_{i+1}(\vec{p}_{i2}, \vec{p}_{i+1})\} \quad (3.18)$$

From (3.18) we can write (3.19) and (3.20).

$$\vec{p}_i^* = \arg \min_{\vec{p}_i} \{J_i(\vec{p}_i) + J_{i+1}(\vec{p}'_{i+1})\} \quad (3.19)$$

$$\vec{p}'_{i+1} = \arg \min_{\vec{p}_{i+1}} \{J_{i+1}(\vec{p}_i, \vec{p}_{i+1})\} \quad (3.20)$$

That is correct with all possible  $\vec{p}_i$  remaining in the same *cluster*.  $\vec{p}'_{i+1}$  is defined as the optimal  $PU_{i+1}$  coding mode for all  $\vec{p}_i$  in the same cluster.  $\vec{p}'_{i+1}$  estimation becomes similar to all  $\vec{p}_i$  in the same cluster.

This method is summarized in three steps:



Table 3.6 Configurations

Configurations	$C_0$	$C_1$	$C_2$	$C_3$	$C_4$
Spatial Activity Adaptation		x			x
Short-List $M_{i+1}$			x		x
Residual Based Clustering				x	x

1. Construct the different clusters by analyzing mode residuals during the **RMD** process applied to  $PU_i$
2. If  $\vec{p}_i$  is the first of its cluster, optimize  $\vec{p}_{i+1}$  among all possible modes
3. Otherwise, optimize  $\vec{p}_{i+1}$  among previous  $\vec{p}_{i+1}^*$  of the same cluster and new **MPMs**

Many bits are saved if the optimal mode belongs to **MPMs**. Consequently, the third step ensures that **MPMs** of  $\vec{p}_{i+1}$  are always tested if they differ from the **MPMs** previously considered within the cluster, i.e. if different from  $\vec{p}_i$ . This technique is an effective shortcut as long as the number of final clusters is low, which is often verified for small **PU** sizes.

### 3.3.2 Experiments and Results

Acceleration methods presented in Section 3.3.1 have been implemented in **HM16.12** with the *Dual-JRDO* algorithm. Results are presented with five configurations  $\{C_k\}_{k=0}^4$  summarized in Table 3.6. The anchor is **HM16.12** with *Independent-RDO*. Impacts of each solution on both *Cpx* and **R-D** efficiency are individually evaluated. For comparison purpose, we include results of *Dual-JRDO* in **HM16.12** without acceleration ( $C_0$ ).

Test conditions follow the recommendations of the Joint Collaborative Team on Video Coding (JCT-VC) [Bos13] in **AI** configuration. Coding efficiency is measured using *Bjontegaard* **BD-BR** [Bjo01a] with **PSNR**. Since **BD-BR** is the difference of areas under two **R-D** functions, we choose to add a fifth **R-D** point at  $QP = 42$  in order to cover a larger bitrate range with the same metric. We use the configuration files provided with **HM16.12**.

For this experiment, **YUV** **BD-BR** results of each configuration against *Independent-RDO* are presented in Table 3.7. *Cpx* savings over initial *Dual-JRDO* ( $C_0$ ) are presented in Table 3.8. *Cpx* savings are estimated according to (3.21), with  $Time_{ref}$  and  $Time_{current}$  being the encoding times of **HM16.12** with *Dual-JRDO* without modification and *Dual-JRDO* with the proposed optimizations, respectively.

$$Cpx(\%) = \frac{Time_{current} - Time_{ref}}{Time_{ref}} * 100 \quad (3.21)$$



We observe systematic bitrate savings against *Independent-RDO* for all considered coding configurations. However, the more aggressive is the algorithm, in terms of  $Cpx$  reduction, the less efficient *Dual-JRDO* is.

Table 3.7 *Dual-JRDO* coding efficiency over *Independent-RDO* in *HM16.12*.

Test sequences		$C_0$	$C_1$	$C_2$	$C_3$	$C_4$
Class B	Kimono	-0.19%	-0.21%	-0.20%	-0.20%	-0.20%
	ParkScene	-0.37%	-0.35%	-0.28%	-0.39%	-0.26%
	Cactus	-0.58%	-0.53%	-0.45%	-0.59%	-0.40%
	BQTerrace	-0.53%	-0.52%	-0.42%	-0.52%	-0.39%
	BasketballDrive	-0.57%	-0.49%	-0.53%	-0.58%	-0.48%
	<b>Average</b>	<b>-0.45%</b>	<b>-0.42%</b>	<b>-0.38%</b>	<b>-0.46%</b>	<b>-0.35%</b>
Class C	RaceHorses	-0.39%	-0.35%	-0.28%	-0.39%	-0.25%
	BQMall	-0.62%	-0.56%	-0.47%	-0.62%	-0.41%
	PartyScene	-0.70%	-0.64%	-0.50%	-0.70%	-0.48%
	BasketballDrill	-1.72%	-0.59%	-0.63%	-0.71%	-0.55%
	<b>Average</b>	<b>-0.61%</b>	<b>-0.54%</b>	<b>-0.47%</b>	<b>-0.61%</b>	<b>-0.42%</b>
Class D	RaceHorses	-0.56%	-0.57%	-0.37%	-0.57%	-0.38%
	BQSquare	-0.82%	-0.80%	-0.65%	-0.83%	-0.61%
	BlowingBubbles	-0.52%	-0.48%	-0.35%	-0.54%	-0.36%
	BasketballPass	-0.62%	-0.49%	-0.49%	-0.62%	-0.42%
	<b>Average</b>	<b>-0.63%</b>	<b>-0.59%</b>	<b>-0.46%</b>	<b>-0.64%</b>	<b>-0.44%</b>
Class E	FourPeople	-0.64%	-0.56%	-0.46%	-0.64%	-0.43%
	Johnny	-0.64%	-0.61%	-0.60%	-0.66%	-0.49%
	KristenAndSara	-0.64%	-0.57%	-0.51%	-0.62%	-0.48%
	<b>Average</b>	<b>-0.64%</b>	<b>-0.58%</b>	<b>-0.52%</b>	<b>-0.64%</b>	<b>-0.47%</b>
Class F	BasketballDrillText	-0.84%	-0.69%	-0.68%	-0.84%	-0.59%
	chinaspeed	-1.12%	-1.01%	-0.87%	-1.11%	-0.82%
	slideediting	-0.96%	-0.92%	-0.63%	-0.94%	-0.55%
	slideshow	-0.55%	-0.42%	-0.68%	-0.61%	-0.59%
	<b>Average</b>	<b>-0.87%</b>	<b>-0.76%</b>	<b>-0.67%</b>	<b>-0.88%</b>	<b>-0.60%</b>
All	<b>Average</b>	<b>-0.63%</b>	<b>-0.57%</b>	<b>-0.49%</b>	<b>-0.63%</b>	<b>-0.45%</b>
	<b>Maximum</b>	<b>-1.12%</b>	<b>-1.01%</b>	<b>-0.87%</b>	<b>-1.11%</b>	<b>-0.82%</b>
	<b>Minimum</b>	<b>-0.19%</b>	<b>-0.21%</b>	<b>-0.20%</b>	<b>-0.20%</b>	<b>-0.20%</b>

Table 3.8 *Dual-JRDO* complexity increase over *Independent-RDO* in *HM16.12*.

Test sequences		$C_0$	$C_1$	$C_2$	$C_3$	$C_4$
Class B	Kimono	878%	400%	211%	722%	139%
	ParkScene	905%	424%	203%	703%	137%
	Cactus	796%	442%	203%	729%	144%
	BQTerrace	865%	495%	198%	728%	145%
	BasketballDrive	869%	281%	198%	670%	102%
	<b>Average</b>	<b>863%</b>	<b>408%</b>	<b>202%</b>	<b>711%</b>	<b>133%</b>
Class C	RaceHorses	843%	529%	199%	700%	156%
	BQMall	808%	508%	196%	700%	150%
	PartyScene	833%	645%	196%	733%	178%
	BasketballDrill	817%	453%	195%	691%	139%
	<b>Average</b>	<b>825%</b>	<b>534%</b>	<b>196%</b>	<b>706%</b>	<b>156%</b>
Class D	RaceHorses	693%	615%	200%	719%	176%
	BQSquare	697%	619%	197%	683%	168%
	BlowingBubbles	731%	644%	197%	741%	176%
	BasketballPass	698%	385%	197%	688%	120%
	<b>Average</b>	<b>705%</b>	<b>566%</b>	<b>198%</b>	<b>708%</b>	<b>160%</b>
Class E	FourPeople	798%	423%	203%	646%	135%
	Johnny	780%	280%	202%	565%	102%
	KristenAndSara	1009%	316%	200%	589%	110%
	<b>Average</b>	<b>863%</b>	<b>340%</b>	<b>202%</b>	<b>600%</b>	<b>116%</b>
Class F	BasketballDrillText	960%	444%	197%	695%	140%
	chinaspeed	778%	418%	199%	642%	128%
	slideediting	795%	570%	191%	638%	152%
	slideshow	957%	203%	199%	467%	74%
	<b>Average</b>	<b>873%</b>	<b>409%</b>	<b>196%</b>	<b>610%</b>	<b>123%</b>
All	<b>Average</b>	<b>826%</b>	<b>455%</b>	<b>199%</b>	<b>672%</b>	<b>138%</b>
	<b>Best</b>	<b>693%</b>	<b>203%</b>	<b>185%</b>	<b>467%</b>	<b>74%</b>
	<b>Worst</b>	<b>1009%</b>	<b>645%</b>	<b>212%</b>	<b>741%</b>	<b>178%</b>

Adaptive activation of the model based on spatial activity corresponds to configuration  $C_1$ . In average  $Cpx$  is reduced from 826% to 455% for 0.06% **BD-BR** loss. The slight observed loss for  $C_1$  can be explained by the off-line learning to approximate the threshold  $Th$  used in the decision.

The  $C_2$  configuration uses **RMD** during the  $\bar{p}'_{i+1}$  estimation. It is one of the most efficient in terms of  $Cpx$  reduction. We observe the average  $Cpx$  decrease to 199% against *Independent-RDO* configuration, at the cost of an average **BD-BR** increase of 0.12%. The results of this solution confirm that any short-listing approach efficient into *Independent-RDO* can be easily transposed into *Dual-JRDO* framework.

The  $C_3$  configuration uses prediction mode clustering based on residual analysis. **BD-BR** gains are better preserved by suppressing redundant coding process without any approximation. Experimental observations show that cases of identical residual for different predictors occur rarely in textured content. The computational cost of comparing all residuals is also a non-negligible overhead. These two facts explain why  $Cpx$  does not significantly decrease (from 826% to 672% in average).

The  $C_4$  configuration represents the combination of the three solutions from Section 3.3.1. For each **PU**, the algorithm equivalent to  $C_1$  decides whether *Dual-JRDO* is to be used or not. Next, the algorithm corresponding to the  $C_3$  configuration builds the mode clusters based on the **RMD** process. Finally, for the first tested mode of each cluster, **RMD** is enabled while analyzing  $PU_{i+1}$ . For any new mode that belongs to the same cluster, the solution described in Section 3.3.1 is applied. The final Fast *Dual-JRDO* combination limits the  $Cpx$  increase to 138% against *Independent-RDO*, with an average **BD-BR** gain of  $-0.45\%$  and up to  $-0.82\%$ .

In this Section, we have proposed three acceleration methods to benefit from Inter-Block dependencies and improve **HEVC** Intra coding efficiency with limited computational complexity overhead.

### 3.4 Dual-JRDO for quantizer parameter estimation

In Section 3.2, we proposed an encoding optimization method that jointly optimize a set of spatial predictor for global **RDO** purpose. We have shown that relatively small bitrate savings are achieved by the method compared to the introduced computational complexity increase. However, we demonstrated that the spatial distortion propagation dependency shall be considered for optimizing overall coding efficiency. To better exploit this dependency, we aim to use the *Dual-JRDO* framework in order to optimize the **QP**, considering the distortion is more correlated to the **QP** than prediction modes. Note that for optimizing predictor, the

basic unit to consider in HEVC is the PU while for optimizing quantizer it is the CU. Hence, in the following we only consider CUs.

QP optimization at CU-level requires the use of AQ method, that affects a QP offset, positive or negative, for each CU. This strategy introduces an additional dependency that have been named the quantizer signalization overhead in Section 2.8. Thanks to the design of *Dual-JRDO*, other dependencies affecting the optimization process (spatial distortion propagation, MPM dependency, entropy coding dependency) are indirectly addressed. However, the quantizer signalization overhead may be specifically addressed if necessary. Other works, such as the dynamic-programming-based optimization proposed by *Ortega and Ramchandran* in [OR95], address the optimization of quantizer signalization.

The HEVC reference model, that is used as reference in our experiments, implements an independent AQ method. In order to achieve a fair comparison with our proposal, named *Dual-JRDO AQ*, we compare the achieved coding efficiency against the performance of *Independent-RDO* with or without the initial AQ method of the reference model. Results are presented in Section 3.4.2.

### 3.4.1 Reference AQ method

The exhaustive adaptive quantization method implemented in the HM is part of the JCT-VC proposal of *McCan et al.* in [MHK<sup>+</sup>10]. Two parameters are used in order to define the search space for optimal QP: *MaxDeltaQP* and *MaxCuDQPDepth*. The optimal QP offset is estimated in the range  $[-MaxDeltaQP; +MaxDeltaQP]$  for each CU that lies belongs to a QuadTree depth lower or equal to *MaxCuDQPDepth*, compared to the CTU level. It is an exhaustive search so the computational complexity of analyzing a given CU, that satisfies the depth condition, suffers a factor of  $2 \times MaxDeltaQP + 1$ . Each QP offset produces a different R-D cost and the minimal is taken as the optimal. We keep the same notations as earlier for RDO and the QP of the  $i^{th}$  CU is written  $\vec{p}_i$  hereafter.  $\vec{p}_i$  is a 1-D vector that corresponds to an offset added to the frame's QP and applied to the  $i^{th}$  CU. Consequently,  $\vec{p}_i$  is operated as a scalar.

R-D cost computation, using the Lagrangian-based RDO, induces the issue of choosing the correct  $\lambda$  value as Lagrangian multiplier, as discussed in Section 1.1.5. This issue is usually solved by considering  $\lambda$  as a function of QP. The  $\lambda(QP)$  function may take several forms, but is usually a monotonic increasing function. In order to allow fair comparison of two QP offsets, the implementation of the AQ method in the reference model use the same  $\lambda$  value for estimating all R-D costs, that is estimated from the frame QP.

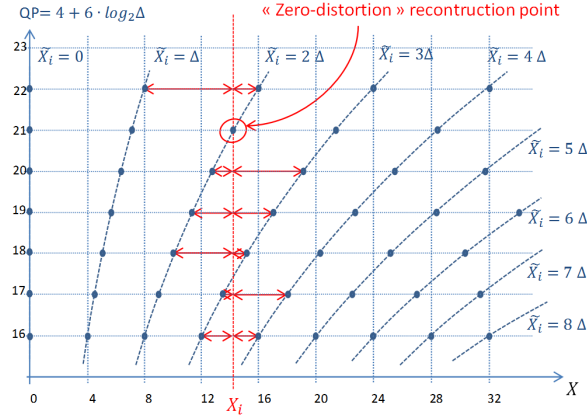


Fig. 3.8 Distortions according to possible reconstruction levels when varying the QP

Hereafter, we discuss the problem of comparing R-D costs with different QP and  $\lambda$  values that leads to an undesired result. For two offset  $p_1$  and  $p_2$ , with  $p_1 < p_2$ , we can compare the two R-D cost as in (3.22).

$$\min_{\vec{p}_i} \{D(\vec{p}_i) + \lambda(\vec{p}_i)R(\vec{p}_i)\}, i \in \{1, 2\} \quad (3.22)$$

We assume such comparison leads to favor the smallest QP, which is not desired. Moreover, the Lagrangian value is intuitively a solution to localize the search space in a given range of the R-D curve. Modifying the Lagrangian values is equivalent to search for an optimal operational point in two independent search spaces, that is also not desired.

Basically, the quantization process in HEVC is a rounding of the transformed residual coefficient, parametrized by the interval between two reconstruction levels. The consideration of AQ in our case aims to determinate if small variation of the quantization granularity allow to better represent coefficients, i.e. diminish the distortion for the same bitrate. An illustration of this case is given in Fig. 3.8.

The quantization step, i.e. the distance between two reconstruction levels, is multiplied (or divided) by 2 when the quantization is increased (or decreased) by a value of 6. It means that a QP offset of 6 is equivalent to increase or decrease the number of reconstruction levels, as shown in Fig. 3.8. Considering the objective is to better match coefficients distribution without increasing the bitrate, we set  $MaxDeltaQP = 6$ . We set  $MaxCuDQPDepth = 3$  in order to apply the AQ method to all CUs, based on a CTU size of 64x64. Results of this configuration, in YUV BD-BR for the first frame of each sequence, against no-AQ are presented in the first column (*With overhead*, i.e. quantizer signaling overhead) of Table. 3.9.

Table 3.9 AQ method coding efficiency with or without QP offset overhead over no AQ in HM16.12.

Test sequences		With overhead	Without overhead
Class B	Kimono	0.61%	-1.26%
	ParkScene	0.74%	-1.61%
	Cactus	0.38%	-2.55%
	BQTerrace	-0.04%	-2.66%
	BasketballDrive	0.73%	-2.37%
	<b>Average</b>	<b>0.48%</b>	<b>-2.09%</b>
Class C	RaceHorses	0.29%	-2.11%
	BQMall	0.33%	-2.69%
	PartyScene	0.01%	-2.44%
	BasketballDrill	0.66%	-2.46%
	<b>Average</b>	<b>0.32%</b>	<b>-2.42%</b>
Class D	RaceHorses	0.50%	-2.58%
	BQSquare	0.21%	-2.41%
	BlowingBubbles	0.61%	-2.08%
	BasketballPass	0.55%	-2.54%
	<b>Average</b>	<b>0.47%</b>	<b>-2.40%</b>
Class E	FourPeople	0.92%	-2.75%
	Johnny	0.21%	-2.95%
	KristenAndSara	0.86%	-2.62%
	<b>Average</b>	<b>0.66%</b>	<b>-2.77%</b>
Class F	BasketballDrillText	0.34%	-2.91%
	chinaspeed	-2.62%	-5.92%
	slideediting	-2.32%	-5.21%
	slideshow	-4.26%	-7.83%
	<b>Average</b>	<b>-2.22%</b>	<b>-5.47%</b>
All	<b>Average</b>	<b>-0.06%</b>	<b>-3.00%</b>
	<b>Best</b>	<b>-4.26%</b>	<b>-7.83%</b>
	<b>Worst</b>	<b>0.92%</b>	<b>-2.08%</b>

Results show an average bitrate decrease near zero ( $-0.06\%$ ), because significant coding gains are achieved on few sequences, while a small bitrate increase is observed on a large majority of sequences. We made the assumption that the quantization process is improved in general, but rarely appears to compensate the bitrate increase related to the quantizer signalization overhead.

In order to estimate this overhead we propose to estimate the bitrate consumed for the offset signalization during the coding procedure, and subtract it to the total bitrate. The resulting bitrate is used for the **BD-BR** computations that is presented in the second column (*Without overhead*) Table. 3.9. We observe systematic bitrate savings from  $-2.08\%$  to  $-7.83\%$  with an average value of  $-3.00\%$ . These results demonstrates that the signalization of quantization offsets induces significant bitrate increase and should be improved. However, the **AQ** method, apart from the signalization overhead, efficiently improves the coding.

### 3.4.2 *Dual-JRDO* **AQ** results

The *Dual-JRDO* is configured similarly to the reference **AQ** method. **QP** offsets belong to the range  $[-6; +6]$  and all **CUs** from  $64 \times 64$  to  $8 \times 8$  are optimized with the *Dual-JRDO* as presented in Section 3.2.1. The result of *Dual-JRDO* **AQ** is presented in Table. 3.10 against no **AQ** method and against the independent **AQ** approach.

We observe that *Dual-JRDO* is able to provide some gains compared to no **AQ** configuration, but several sequences still suffer from **R-D** losses, which makes it impractical. The limited improvements of the proposed method are caused by the signalization overhead of **QP** offsets. When compared to the independent **AQ** we show that an average bitrate savings of  $-0.64\%$  is achieved, with almost systematic gains. *Dual-JRDO* exploits correlation between horizontally neighboring **CUs**, that may sometime decrease the correlation between vertically neighboring **PUs**. In such situations, the global coding efficiency may be negatively affected.

## 3.5 Conclusion

To conclude, we demonstrated that systematic bitrate savings can be achieved by jointly optimizing coding parameters over dependent units. By considering the simplified case of spatial predictor optimization over 2 **PUs** (*Dual-JRDO*), average **BD-BR** gains are of  $-0.77\%$  in the **AVC** reference model and  $-0.71\%$  in the **HEVC** reference model. When the joint optimization is applied on 4 units (*Quad-JRDO*), these average gains increase to  $-1.78\%$  and  $-1.47\%$ , for **AVC** and **HEVC** respectively. However, such a restricted joint



Table 3.10 *Dual-JRDO AQ* method coding efficiency in *HM16.12*.

Test sequences		Vs No-AQ	Vs Independent-AQ
Class B	Kimono	0.37%	-0.24%
	ParkScene	0.36%	-0.38%
	Cactus	-0.28%	-0.64%
	BQTerrace	-0.66%	-0.60%
	BasketballDrive	0.16%	-0.58%
	<b>Average</b>	<b>-0.01%</b>	<b>-0.49%</b>
Class C	RaceHorses	-0.12%	-0.41%
	BQMall	0.03%	-0.30%
	PartyScene	-0.46%	-0.47%
	BasketballDrill	-0.56%	-1.20%
	<b>Average</b>	<b>-0.28%</b>	<b>-0.59%</b>
Class D	RaceHorses	-0.15%	-0.63%
	BQSquare	-0.93%	-1.12%
	BlowingBubbles	-0.20%	-0.80%
	BasketballPass	0.47%	-0.10%
	<b>Average</b>	<b>-0.20%</b>	<b>-0.59%</b>
Class E	FourPeople	0.14%	-0.77%
	Johnny	0.44%	0.21%
	KristenAndSara	-0.04%	-0.87%
	<b>Average</b>	<b>0.18%</b>	<b>-0.47%</b>
Class F	BasketballDrillText	-0.80%	-1.13%
	chinaspeed	-3.40%	-0.82%
	slideediting	-3.29%	-1.00%
	slideshow	-5.16%	-0.92%
	<b>Average</b>	<b>3.16%</b>	<b>-0.97%</b>
All	<b>Average</b>	<b>-0.70%</b>	<b>-0.64%</b>
	<b>Best</b>	<b>-5.16%</b>	<b>-1.20%</b>
	<b>Worst</b>	<b>0.47%</b>	<b>0.21%</b>

optimization remains close to the local optimization used in the *Independent-RDO* and the computational complexity overhead is already tremendous.

In order to tackle this complexity issue, a low-complexity *Dual-JRDO* scheme for HEVC is proposed in Section 3.3. A robust source spatial activity is first used to efficiently activate or not the *Dual-JRDO* model for each PU. The RMD short-listing algorithm is successfully integrated to all possible steps of the process, i.e. local optimizations included within the joint optimization. Finally, a prediction mode clustering approach is introduced that removes redundant computations for modes resulting in identical residue. This last acceleration method is based on the fact that similar residues on a given PU should lead to the same impact on the optimal coding of PUs that use it as reference for prediction. Using these different techniques of complexity reduction, the computational overhead can be reduced from 826% to 134% while ensuring systematic bitrate savings of  $-0.45\%$  in average.

We also proposed to use the *Dual-JRDO* framework for jointly optimizing QP offsets in the context of AQ. AQ is shown to perform very well for Intra coding, in terms of coding efficiency, but suffers important coding overhead for the signalization of QP offsets.

Computational complexities of proposed JRDO approaches make them unusable for real-time applications and the relatively small coding efficiency improvement does not justify such a complexity. However, numerous low-complexity approaches of Intra coding are based on a supervised learning of parameters that uses the *Independent-RDO* as the ground truth. Such off-line learning required the ground truth to be optimal, i.e. it must provide the highest coding efficiency regardless of related computational complexity. An extensive optimization process such as the *Quad-JRDO* method proposed in this Chapter is therefore a preferable ground truth alternative compared to the *Independent-RDO*. Consequently, we believe that JRDO models must at least be considered in learning phases, as long as the complexity remains acceptable for the model training phase.

The initial motivation for these experiments was to evaluate the highest coding efficiency achieved by considering dependencies in the RDO process. We saw that an exhaustive approach rapidly reach its limits due to the exponential increase of computational complexity. Several ideas of JRDO models remains untested, such as an estimation of the propagated distortion based on a prediction process that overlaps on future blocks. Considering the complexity and modest coding efficiency improvements of our proposed solutions, we assume the JRDO in Intra-only context to be of limited interest. Consequently, despite several ideas on the topic, we chose to not go any further with potential models in Intra-only coding scheme. The next chapter focuses on temporal dependencies and applying the JRDO for Inter coding.



# Chapter 4

## Coding dependencies in Inter coding

Fewer dependencies may affect the Inter coding efficiency, compared to the Intra coding case, in the considered video coding scheme. The two dependencies related to Inter coding are defined as the temporal distortion propagation and the **MV** prediction-related dependency. We assume that in usual codecs, larger part of the bitrate is used to transmit transformed residue compared to the bitrate required for motion information. It is especially correct for high bitrate, as discussed by *Stankowski et al.* in [SKG<sup>+</sup>14]. Consequently, we assume in this chapter the opportunities in terms of coding efficiency improvement to be significantly more important by properly handling the temporal distortion propagation than the **MV** prediction-related dependency. Consequently, the following of this chapter puts the emphasis on techniques that consider the temporal distortion propagation for global optimization.

The Intra prediction consists in a spatial projection of pixels, i.e.  $N$  pixels are translated towards  $N \times N$  pixels. Hence, it is difficult to linearly model the relationship between distortions of reference and predicted units. The problem is simplified for motion compensation which is a temporal block to block matching, i.e. all reference pixels are translated in the same direction. Consequently, the distortion introduced on reference samples will linearly affect the prediction error. Sub-pel interpolation does not affect this statement. This linear process may either be modeled in order to achieve global optimization, or addressed through exhaustive joint optimization such as trellis implementation. After presenting state-of-the-art methods that cover these two approaches, we develop our dependency model for the temporal distortion propagation solution named **Rate Distortion Spatio-Temporal Quantization (RDSTQ)** for efficient adaptive quantization. The proposed technique and related works have been published in [RLTBB17] and [BLTR<sup>+</sup>18b].

The remainder of this chapter is organized as follows. Section 4.1 gives an overview of the context and works which consider the temporal distortion propagation for achieving global optimization. The temporal distortion propagation model considered in our solution is

presented in Section 4.2. Section 4.3 investigates the proposed HEVC video optimization solution exploiting the formalized dependencies through adaptive quantization. Insights on the proposed solution and implementation details under two HEVC software encoders are provided in Section 4.4. Section 4.5 gives the experimental results showing the benefits of the proposed model within the two considered codecs. Finally, Section 4.7 concludes this Chapter.

## 4.1 Previous Methods

A large number of methods have been proposed in order to take into account the temporal distortion propagation in video coding. These methods may be organized into categories based on the following elements:

- the considered theoretical model, based on assumptions or empirical observations
- the granularity of the dependency (GOP, frame, block, pixel)
- the computational complexity, that is mainly introduced by a pre-analysis or a multi-pass coding procedure
- the parameter to be optimized, e.g.  $\lambda$  value or quantization parameter

The pre-analysis, generally called look-ahead, consists in a video source analysis without any encoding decision. The look-ahead provides useful data to the encoder and introduces a manageable complexity increase. Indeed, it usually runs in parallel with the encode thanks to an efficient use of multi-threading. Apart from some delay, it lowers significantly the impact of look-ahead processing with respect to the encoding. Multi-pass consists into multiple encodings of the video, refining the coding parameters at each pass. It generates a significant computational complexity. Look-ahead is often assimilate to a 2-pass encoding, but the difference is that no actual encoding is processed during a look-ahead analysis.

In the following, we develop the usual concept modeling the temporal distortion propagation. As developed in Section 1.4, the prediction error variance  $\sigma_i^2$  may be expressed as a function of the source difference  $\mathbb{E} (X_i - X_{ref})^2$  and the reference distortion  $D_{ref}$  described by (4.1).

$$\sigma_i^2 = \mathbb{E} \left[ (X_i - X_{ref})^2 \right] + D_{ref} \quad (4.1)$$

Note that, compared to the function in (1.42), we remove the last term by assuming the high-rate use case. When introducing this expression into the Shannon R-D function (1.8), we obtain the development in (4.2).

$$D_i = c \sigma_i^2 2^{-2R_i} = \underbrace{c \mathbb{E} \left[ (X_i - X_{ref})^2 \right]}_{=d_i} 2^{-2R_i} + \underbrace{c D_{ref} 2^{-2R_i}}_{=p_i D_{ref}}, \quad (4.2)$$

where  $c$  is the constant modeling source distribution and  $R_i$  is the rate allocated to the  $i^{th}$  coding unit. Note that the coding unit is not necessarily an HEVC CU but may be a frame for example. The distortion is hereafter expressed with the form proposed in (4.3).

$$D_i = d_i + p_i D_{ref}, \quad \text{with } p_i = 2^{-2R_i} \quad (4.3)$$

The first term of the sum  $d_i$  is only related to the allocated rate and the innovation of the signal, or unpredictable part, that is independent from the reference samples. We name this term the intrinsic distortion, or local distortion, i.e. the distortion that is only caused by local coding parameters. The product  $p_i D_{ref}$  refers to the temporal distortion propagation.  $p_i$  represents the amount of distortion propagated from the reference to the current unit.

Two categories of dependency model can be distinguished based on (4.3). The first category simplifies the model by omitting the impact of local coding decisions on the  $p_i$  parameter. Here the local coding parameters, written  $\vec{p}_i$  in previous chapters, are symbolized by  $R_i$ . This assumption implies  $p_i$  to be treated as a constant when optimizing  $\vec{p}_i$ . Models falling in this category are named hereafter as *Simplified propagation methods* and are depicted in Section 4.1.1.

A second category depicts the solutions that does not use simplification and, basically, handle the addition of reference distortion at the prediction error level. We name these solutions as *Dependent propagation methods*, since a dependency is assumed between reference distortion impact  $p_i$  and local coding parameters  $\vec{p}_i$ . Some of these methods are presented in Section 4.1.2. Finally, some solutions that do not model the temporal distortion propagation but focus on exhaustive joint optimization are presented in Section 4.1.3. Models based on empirical observations are also included in Section 4.1.3.

### 4.1.1 Simplified propagation methods

#### Quantization and $\lambda$ parameter cascading

In the literature, the temporal distortion propagation has been considered between temporal layers for optimizing hierarchical coding structures. In the following we focus on the RA

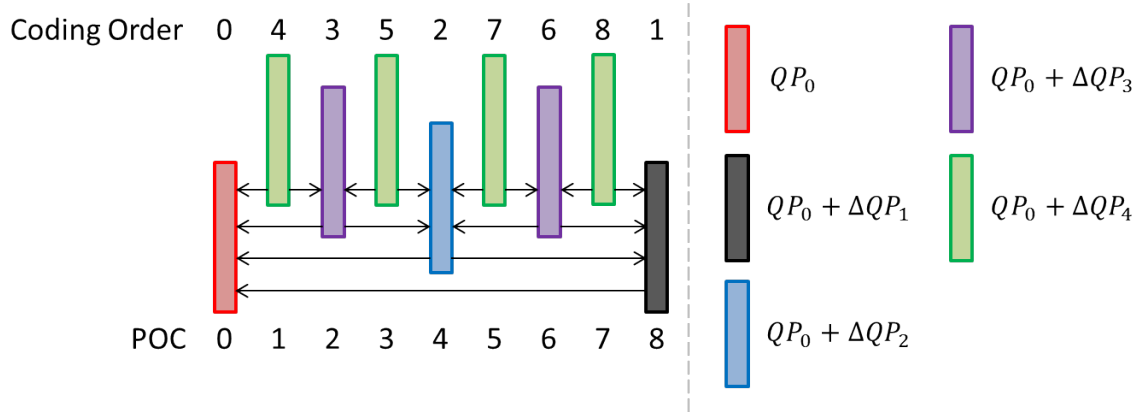


Fig. 4.1 Illustration of the Quantization Parameter Cascading (QPC) for hierarchical layers.

coding structure, as defined in Section 2.1. In the context of RA coding structure, coding efficiency is closely related to the Quantization Parameter Cascading (QPC) technique that sets the QP offset between layers, as illustrated on Fig. 4.1. Several simplified distortion propagation-based methods take into account the distortion propagated from lower layers toward the higher layers for optimized QPC. In [GWY<sup>+</sup>16], Gong *et al.* have observed that the distortions of the lowest layer and of any higher layer frame follow the linear relationship in (4.4).

$$D_l = a_{l0}D_0 + b_{l0}, \quad (4.4)$$

where  $D_l$  is the distortion of any frame that belongs to the  $l^{\text{th}}$  temporal layer,  $a_{l0}$  and  $b_{l0}$  are parameters that model the relationship between the  $l^{\text{th}}$  layer and the lowest one, i.e.  $l = 0$ . Authors also claim that rates of the lowest temporal layer and of any higher layer frame meet the power relationship. We do not further discuss it in this Chapter, since the emphasis is put on the relationship between distortions. Developing these models, an optimization problem for QPC is proposed and simplified based on following observations:

- Near optimal R-D performance is achieved if the layer QP offset  $dQP_l = 1$  for all layers  $l > 1$ . Consequently, only  $dQP_1$  matters, that is the first offset.
- The optimal value of  $dQP_1$  is unaffected by the GOP size.

Two models are further proposed by Gong *et al.*, a first one setting adaptively  $dQP_1$  based on a pre-encoding step, and a second one that empirically estimates  $dQP_1 = 5$  as the optimal offset. When compared to the usual  $dQP_l = 1, \forall l > 1$  in the HM14.0 codec, the first adaptive method enables  $-4.87\%$  BD-BR while the second empiric method enables  $-4.32\%$  BD-BR. We observe that adaptive QPC is of limited interest compared to the empirical offset, for a

known and fixed coding structure. Note that coding structures may be adaptive, unevenly inserting I-frame or dynamically modifying the number of B-frames. When using an adaptive coding structure, a fixed offset may not be optimal and adaptive QPC is necessary.

*Yang et al.* improve the previous approach in [YWGF17] by considering temporal and spatial source complexity models and fitting the optimal  $dQP_1$  function through off-line learning. The temporal complexity model is based on the average inter-frame luminance difference that includes a low-pass filter in order to ignore the global luminance variations. The spatial complexity model is based on the pixels variance within pictures. An interesting observation is the following: if the GOP is temporally complex,  $dQP_1$  tends to decrease, while if the pictures are spatially complex,  $dQP_1$  tends to increase. The comparison of this method with the one presented by *Gong et al.* in [GWY<sup>+</sup>16] shows an average improvement from  $-5.33\%$  to  $-5.60\%$  BD-BR, while no pre-encoding step is required.

In [YWGW16], *Yang et al.* experimentally confirm that varying the QP offset between the two first layers has no influence on the distortion relationship between frames of higher temporal layers and their reference frames. This useful observation allows to express the distortion of the entire GOP as a function of the first layer distortion and the QP offset of the next layer. However, authors did not provide information on the estimation process to determine the distortion propagation function between the various layers. When optimizing the first layer QP offset, authors achieve  $-3.3\%$  average bitrate savings in the HEVC reference model compared to the classical offset of 1 on each layer.

In classical RDO the  $\lambda$  parameter, used to compute the R-D cost, is a function of the QP value, as discussed in Section 1.1.5. When optimizing the QP for different frames or blocks, papers cited above often considered the  $\lambda$ -QP relationship to remain unchanged, which may be questionable. In the following, we introduce two techniques taking into account the  $\lambda$  value for optimizing coding layers in hierarchical coding structures.

*Li et al.* propose an empirical and off-line based solution in [LXL13] to take into account the distortion propagation from reference frames. The coding scheme considered is *IBBPBBP* in the context of AVS. Assuming the distortion introduced on reference frames, i.e. I-frames and P-frames, affects the lower temporal layers, i.e. B-frames, the R-D cost minimization of reference frames is expressed as (4.5) which is equivalent to scale the  $\lambda$  as in (4.6).

$$\min_{\vec{p}} J(\vec{p}) = \min \{ (1 + \omega) D(\vec{p}) + \lambda R(\vec{p}) \} \quad (4.5)$$

$$\min_{\vec{p}} J(\vec{p}) = \min \left\{ D(\vec{p}) + \frac{\lambda}{(1 + \omega)} R(\vec{p}) \right\} \quad (4.6)$$



The value  $\omega$  represents the portion of distortion that affects B-frames and is learned through extensive off-line computation. The model is efficient, with an average bitrate reduction of  $-3.7\%$ , and does not suffer from significant computational complexity increase. However, it does not adapt to various coding structures or content characteristics, despite the large variation of optimal  $\omega$  value for each sequence in the learning database. An interesting observation is that optimal  $\omega$  value tends to increase linearly with the QP value.

*Yang et al.* also consider the affine distortion propagation function between layers in [YWG<sup>+</sup>17]. After some mathematical developments, they end up with the conclusion that the optimal  $\lambda$  value for each frame is a scaled version of its original QP, as presented in (4.7). An important observation made by authors is that the optimal scaling factor expression is related to the layer and the motion characteristics measured at the GOP-level.

$$\lambda_l = \begin{cases} \lambda_{org} & (l = 0) \\ \beta_l \times \lambda_{org} & (l > 0) \end{cases} \quad (4.7)$$

$\lambda_l$  is the Lagrangian multiplier of the  $l^{th}$  temporal layer,  $\lambda_{org}$  the original Lagrangian multiplier considered and  $\beta_l$  the scaling factor. The GOP motion characteristic used to compute the  $\beta_l$  value is an average frame luminance difference between only a few frames of the entire GOP, which maintains the complexity overhead of the method reasonable. Their methods enable  $-4.59\%$  average bitrate savings in the reference model.

We report in Table. 4.1 the results claimed by the authors of the different methods cited above. Note that results reported for *Gong* [GWY<sup>+</sup>16] (Constant) stands for the fixed offset of 5 between the two first layers as presented in their paper. Results for the two methods of *Gong et al.*, have been extracted from [YWGF17] in order to have similar test dataset. Despite the different implementations or considered standards, one observes that all presented solutions claim similar gains. This means that, the solution of *Gong et al.* that is based on a fixed offset whatever the sequence is able to compete with each other solution and does not require any computational complexity overhead or implementation process. Based on methods presented hereafter, we will observe that significantly higher coding efficiency can be achieved if the dependency is considered at the frame-level or block-level.

### Adaptive Quantization

As shown by the methods presented above, the QP value is an important parameter when considering distortion propagation. However, quantization is applied on blocks and thus, the distortion propagation should be more efficiently modeled when introduced at the block level. Macroblock-Tree is a well known algorithm proposed by *Garett* for x264 open-source AVC encoder in [GG11]. The main interest of this approach is to model the temporal distortion

Table 4.1 Results of methods considering the Inter-layer frames distortion propagation

QPC optimization	
<i>Gong</i> [GWY <sup>+</sup> 16]	-5.33%
<i>Gong</i> [GWY <sup>+</sup> 16] (Constant)	-4.89%
<i>Yang</i> [YWGF17]	-5.60%
<i>Yang</i> [YWGW16]	-3.30%
$\lambda$ optimization	
<i>Li</i> [LXL13]	-3.70%
<i>Yang</i> [YWG <sup>+</sup> 17]	-4.59%

propagation between CUs and adaptively set the optimal QP value for each MB as described by (4.8).

$$dQP_i = str \times \log_2(1 + p_i) \quad (4.8)$$

where  $str$  is a user parameter with an empirical default value set to 2.0.  $dQP_i$  is the QP offset of the  $i^{th}$  CU and  $p_i$  is the proportion of distortion propagated from the reference to the current unit, defined in (4.9).

$$p_i = \gamma_i^{prop} / SATD_i^{intra} \quad (4.9)$$

$SATD_i^{intra}$  is an SATD-based estimation of the intra mode complexity.  $\gamma_i^{prop}$  is the amount of dependence from the  $i^{th}$  CU toward all future ones, i.e. the importance of the CU based on an estimation on how its own distortion may affect future units. The initial algorithm for AVC has also been adapted into the CU-tree algorithm for the x265 open-source HEVC encoder, with an evaluation of its behavior proposed by *Grozman* in [Gro15]. In [RLTBB17], we reported an average bitrate savings of  $-10.1\%$  against no adaptive quantization for the CU-tree in the x265.

*Yin et al.* improved the Macroblock-Tree implementation for x264 in [YWXS15], by focusing on the function in (4.8). They suppose that the optimal mapping from  $p_i$  to  $dQP_i$  should adapt to video sequence characteristics. By using the Competition Decision Algorithm (CDA) presented in [FN13], authors succeed to outperform the initial algorithm from 0.54 average BD-SSIM improvement to 0.68 average BD-SSIM when compared to no adaptive quantization. However, we note that no data is provided on the computational overhead, that should be substantial considering the complexity of the new mapping process. Indeed, this new mapping includes a competition of various  $dQP_i$  based on their R-D costs.

## Rate Control

**Rate Control (RC)** plays an important role in video coding for global **RDO**. The **RC** component aims to maintain uniform picture quality for the given coding constraints. Usual constraints are the **Constant Bitrate (CBR)** and the **Variable Bitrate (VBR)**. **CBR** consists in keeping the bitrate constant over time, that is useful when it comes to broadcast television. **VBR** dynamically adapts the bitrate over time, in a given range, to adapt to the source complexity and ensure a better overall **R-D** performance. Obviously, **RC** cannot perform well if it ignores dependencies between coding elements. We present in the following some techniques that introduced the distortion propagation between frames in order to optimize bit allocation within a **GOP**. A comprehensive overview of the **RC** topic is proposed by *Chen and Ngan* in [CN07]. The **RC** algorithm implemented in the **HEVC** reference software is named hereafter the **R- $\lambda$**  model and was introduced by *Li et al.* in [LLLZ14].

*Wang et al.* observed in [WMW<sup>+</sup>13] that the distortion of the current frame has an approximate linear correlation with the distortion of reference samples used for prediction, as expressed in (4.10).

$$D(QP_0, QP_1) = m_1 D_{ref}(QP_1) + (m_0 - m_1) D_{ref}(QP_0) \quad (4.10)$$

$D$  is the distortion of the current frame to be estimated and  $D_{ref}$  is the distortion of the reference frame. The reference frame is constrained to be the nearest available reference in temporal distance. This constraint is supposed relevant, since the nearest reference frame is often the optimal one, as discussed by *Zhao et al.* in [ZWC16].  $QP_0$  and  $QP_1$  are the quantization parameters of the reference frame and the current one, respectively.  $m_0$  and  $m_1$  are model parameters. Based on this dependency model, a frame-level **RC** is proposed which overcomes the **R- $\lambda$**  model by up to  $-3.33\%$  **BD-BR** in the **HM8.0**.

$$D(QP_0, QP_1) = \begin{cases} m_1 D_{ref}(QP_0) + r_1, & \text{if } QP_0 \neq QP_1 \\ m_0 D_{ref}(QP_0), & \text{if } QP_0 = QP_1 \end{cases} \quad (4.11)$$

In [LZZR18], *Li et al.* use similar dependency models that *Wang et al.* in [WMW<sup>+</sup>13] and presented in (4.11).  $r_1$  is a model parameter. They also validate the affine relationship between current and reference distortions through mathematical developments. An average bitrate savings of  $-3.41\%$  is shown against the native frame-level **RC** method implemented in the **HM12.0**. An important conclusion that is drawn from these studies is the simplification of the relationship between current and reference distortions when quantization parameters are identical.

*Zhou et al.* propose in [ZWTZ17] to approximate the motion compensated error  $MSE_i$  of the  $i^{th}$  CTU by the additive model expressed in (4.12).  $MSE_i^{src}$  is the source motion compensated error,  $D_{ref}$  is the quantization error introduced on prediction reference samples and  $\alpha$  is an estimation parameter. After relating the lagrangian multiplier  $\lambda$  to  $D_{ref}$ , using the high-rate approximation and the usual  $\lambda(QP)$  function, they proposed the linear distortion model exposed in (4.13) with  $a$ ,  $b$  and  $c$  being model parameters. The values  $MSE_i$  of all CTUs are averaged for multiple frames within a window in order to estimate a temporal redundancy ratio. This ratio is further used to adapt the QP offset of each frame. *Zhou et al.* enable an average BD-BR of  $-1.54\%$  in RA configuration within the HEVC reference model.

$$MSE_i = \alpha (MSE_i^{src} + D_{ref}) \quad (4.12)$$

$$MSE_i = aMSE_i^{src} + b\lambda + c \quad (4.13)$$

### 4.1.2 Dependent propagation methods

In [PAZD11] and [PAZ<sup>+</sup>13a], *Pang et al.* express the variance of prediction error  $\sigma_n^2$  as a weighted sum of the reference frame distortion  $D_{n-1}$ , variance of the prediction error with the source signal of the reference frame  $\tilde{\sigma}_n^2$  and the quantization step size  $\Delta_n$ . This relationship is exposed in (4.14) with  $\alpha$ ,  $\beta$  and  $\gamma$  the model parameters.

$$\sigma_n^2 = \alpha D_{n-1} + \beta \tilde{\sigma}_n^2 + \gamma \Delta_n \quad (4.14)$$

By introducing this formulation into the R-D Shannon bound, authors are able to formulate a new constrained minimization problem expressed in (4.15).

$$\begin{aligned} \min_{\{R_i\}_{i=1}^N} & \sum_{i=1}^N \{D_i(R_i)\} \\ \text{s.t.} & \sum_{i=1}^N R_i \leq R_T \\ & R_1 = G\left(\frac{a_0}{D_1+b_0} + c_0\right) \\ & R_j = a_1 \log \frac{\sigma_j^2}{D_j}, \text{ with } j = 2, 3, \dots, N \\ & \sigma_j^2 = \alpha D_{j-1} + \beta \tilde{\sigma}_j^2 + \gamma \Delta_j \end{aligned} \quad (4.15)$$

$G$  is a measure of frame complexity used to set the rate of the first frame and  $a_0$ ,  $b_0$ ,  $c_0$  and  $a_1$  are model parameters. Using successive convex optimization to solve the proposed bit

allocation problem, authors are able to achieve  $-13.29\%$  average bitrate savings compared to the classical **RC** method implemented in the **AVC** reference model.

More complex methods are also considered with a propagation model dependent of local decisions. *Fiengo et al.* express distortion as a convex function of all frames bitrate in [**FCCPP17**]. Primal-Dual Proximal Algorithm is further used to solve the convex optimization problem and achieve near optimal **RC**. In [**WRSW15**], *Wiken et al.* measure the dependencies between coefficients levels after **DCT/DST**, leading to an optimization problem solved by an iterative approach. However, both solutions are based on multi-pass processing, which is unusable for real-time applications.

On the contrary to methods presented in Section 4.1.3, exhaustive computations in the last two methods are used to estimate parameters that describe dependencies relationships. In Section 4.1.3, the exhaustive computations are used to directly optimize the overall coding efficiency, without any understanding of the dependencies.

*Yang et al.* present a **Source Distortion Temporal Propagation (SDTP)** model in [**YZFP12**] that increases the coding efficiency by adaptively scaling the  $\lambda$  value for each **CU**. In this model,  $p_i$  is a function of the **CU** rate  $R_i$ , as defined in (4.3).  $d_i$  is a function of  $R_i$  and the innovation  $\sigma_i^2$  of **CU**  $i$ .  $\sigma_i^2$  is defined here as the part of the signal which is unpredictable, i.e. the residue of prediction before quantization.

$$\sigma_i^2 = \mathbb{E} \left[ (X_i - X_{ref})^2 \right] + D_{ref} \quad (4.16)$$

Using equation (4.16), discussed in the beginning of this Chapter, authors describe dependencies between **CUs** and they adaptively scale the  $\lambda$  value used for **R-D** cost computation. It leads to substantial coding efficiency improvement, i.e.  $-14.05\%$  **BD-BR** in average for 352×288 resolution in the **JM15.1 AVC** reference software. The more the distortion of a **CU** impacts other **CUs**, the more the  $\lambda$  value decreases.

The model proposed in [**YZFP12**] has been further extended by *Xie et al.* in [**XSX<sup>+</sup>15**] to the bit allocation strategy in the context of **RC**. Specific hierarchical coding schemes have also been investigated by *Gao et al.* for **Low-Delay (LD)** [**GZL16**] and in a most recent study for **RA** [**GZLY17**] coding configurations. In the specific case of **HM** and **RA** configuration, coding efficiency increases by 2.2% and can be further improved to 5.2% when the method is coupled with the high-complexity **Multi Quantization Parameter (MQP)** optimization proposed by *Sullivan and Wiegand* [**SW98**]. The **MQP** is an exhaustive search of the optimal delta quantizer within a given range, from -3 to 3 in [**GZLY17**], which increases the encoding run-time by nearly a factor of 6.

### 4.1.3 Exhaustive joint search methods and empirical models

This category consists in explicitly considering dependencies into the coding optimization, which results in exponential exhaustiveness and intractable computational complexity for real-time applications. Exhaustive computations aiming to achieve the best overall R-D efficiency have been explored for diverse issues in video coding during the last decades. We also added a few empirical models in the end of this Section.

*Ramchandran et al.* [ROV94] consider the frame bit allocation in video coding as a trellis problem solved with the Viterbi algorithm. Each trellis node represents a quantizer choice for a given frame, with associated frame R-D cost. Each path represents a unique set of quantization steps. Using a simple coding scheme and pruning rules, they succeed to achieve significant coding gain. Global optimization can also be opposed to local optimization when searching for optimal transformed coefficient levels. In [WLV00], *Wen et al.* use the same trellis approach to jointly optimize all the transformed coefficient levels after quantization.

If low computational complexity is not a stringent requirement, dependencies may also be exploited through brute force search without designing any theoretical model. In [IC15], *Im and Chan* point out that the usual HEVC relationship between  $QP$  and  $\lambda$ , presented in (4.17), is only statistically correct and that  $\lambda$  shall be adapted to the frame.

$$\lambda(QP) = K \times 2^{(QP-12)/3} \quad (4.17)$$

$K$  is a constant which may be tuned based on picture type. They propose to evaluate the quality of multiple reference frames, using different values of  $\lambda$ , and only consider the reference frames with the lowest R-D costs. An illustration of their solution with multiple  $\lambda$  values and related reference results for the inter mode decision is given in Fig. 4.2. Such a solution necessarily suffers from complexity increase as admitted by the authors. For the case depicted in Fig. 4.2, authors announce an encoding runtime to 170% in average compared to the reference and  $-4.53\%$  bitrate savings for RA configuration and 176×144 resolution.

Despite the obvious efficiency of exhaustive modeling approaches, i.e. an exhaustive search cannot possibly be worst than *Independent-RDO*, the HEVC standard offers a tremendous number of coding parameters value combinations. More generally, coding schemes tend to become more complex with an increasing number of available prediction reference samples and a larger number of block-partitioning combinations. Thus, such approaches are not suited for real-time encoding and are not suitable for proof of concept purpose either.

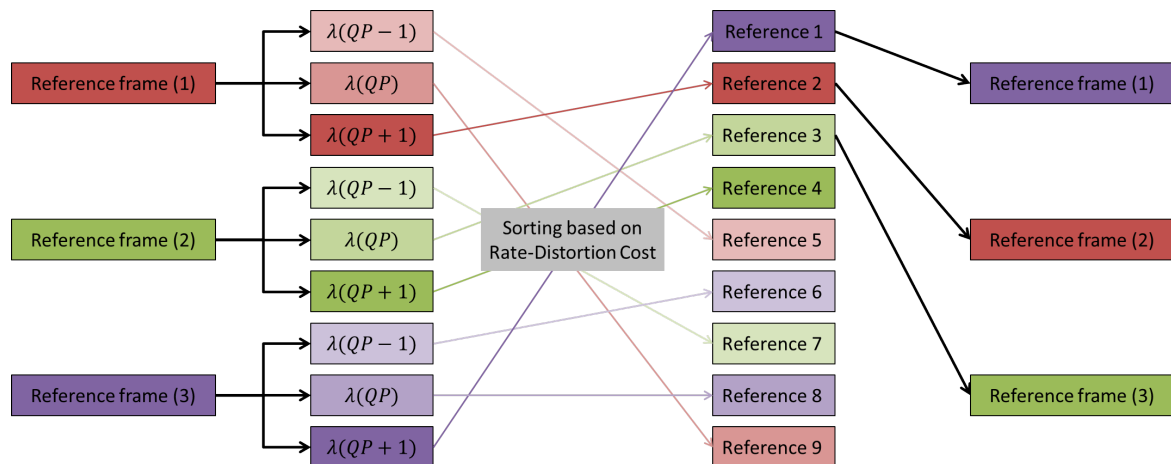


Fig. 4.2 An example of 3 reference frames search with 3 different  $\lambda$  values for Inter-coded frames as presented by *Im and Chan* in [IC15].

## Empirical models

In [XZZ<sup>+</sup>16], *Xu et al.* state that numerous coding techniques are developed by taking into account the dependencies within a **GOP**, a frame or a **CU**. They assume that a dependency also exists between **GOPs** that is currently not introduced into the coding optimization. By counting the number of direct and indirect temporal referencing between **GOPs**, they proposed an adaptive quantization scheme at **GOP**-level. They arbitrarily assume that a frame impact is divided by 2 after each referencing. Proposed offsets enable  $-0.96\%$  average bitrate savings on the luminance component and more than  $-4\%$  on each chrominance component when compared to the reference **HEVC** implementation.

Another empirical observation of the Inter-frame dependency, measured by the percentage of Skip blocks, is given by *Li et al.* in [LOHK07]. Authors proposed a Laplace distribution model for the Lagrangian optimal estimation in **AVC**, that they further improve in [LOHK09]. Authors point out that efficient modeling of the coefficients distribution is not sufficient to estimate the total rate. Indeed, significant amount of rate is consumed by side informations. They overcome this issue by defining a mismatch ratio between the estimated rate and the actual one, and correlating it with the percentage of **MB** coded in Skip mode. Their solution allows a gain of up to  $1.79dB$  in **PSNR** when compared to the initial solution implemented into the **JM**. The important conclusion of this study is that the percentage of Skip blocks, or by extension the probability of Skip, is an informative metric to model the rate.

#### 4.1.4 State-of-the-art conclusion

Based on all studied methods, some conclusions can be drawn. The number of studies focusing on temporal dependencies, without modifying the standard, is significantly higher than for spatial dependencies. Moreover, the claimed results are also much higher in Inter coding case than Intra coding case. It is coherent with the observation we made in Chapter 3, that opportunities of improving coding efficiency, by considering only Intra coding-related dependencies, are limited.

Numerous studies use the model proposed in (4.2). It may be difficult to handle because of the dependencies highlighted in the formula, i.e. the propagation of reference samples distortion depends on current coding decisions. Several studies, described in Section 4.1.1, use simplifications in order to obtain a good mathematical tractability. It allows to partially rely on proven theoretical fundamentals, while often resulting into simple encoder implementations.

One constraint we chose for designing the proposed model was to be close to real-time implementation. Moreover, we desired the model to be proved valid and efficient, without relying exclusively on the coding efficiency results. Thus, we did not consider exhaustive or empirical approaches. Our study falls into the category discussed in Section 4.1.1.

## 4.2 Temporal Distortion Propagation Model

In this section we present the proposed model that introduces the temporal distortion propagation at CU level. Propagation is introduced by temporal predictions between frames within the GOP structure, as proven by the various methods presented in Section 4.1. The distortion propagation model further enables to build an analytical solution deriving optimal local quantizers within a GOP at the CU granularity.

The subscript  $i_t$  is used when referring to the CU with spatial index  $i$  in the frame with temporal index  $t$ .  $N$  denotes the number of CUs in a frame, and  $T$  denotes the GOP size. The video encoding process aims to find the optimal coding parameters  $\vec{p}$  that minimize the total distortion  $D_{Tot}$  under the target rate  $R_{Tot}$  constraint, as expressed in (4.18).

$$\begin{aligned} \min_{\vec{p}} \quad & D_{Tot}(\vec{p}) \\ \text{s.t.} \quad & \sum_{t=1}^T \sum_{i=1}^N R_{i_t}(\vec{p}) = R_{Tot} \end{aligned} \tag{4.18}$$



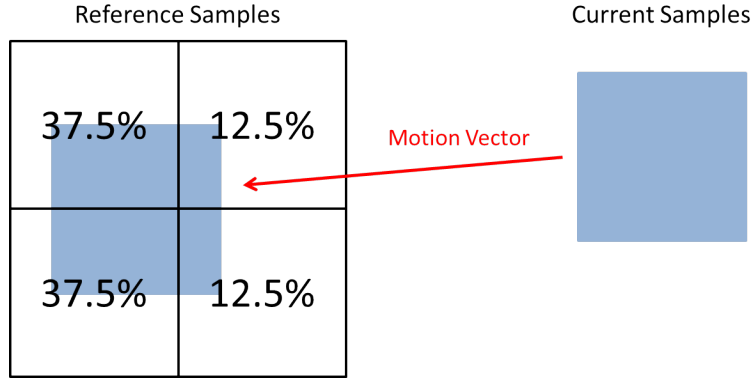


Fig. 4.3 Illustration of the surface ratio resulting from the motion compensation.

By definition we have:

$$D_{Tot}(\vec{p}) = \sum_{t=1}^T \sum_{i=1}^N D_{i_t}(\vec{p}). \quad (4.19)$$

Video encoders aim to maximize the video quality perceived by the **HVS**. To consider the **HVS** in the distortion model, a spatial psycho-visual weighting factor  $\Psi$  is introduced. This factor is applied on each **CU** to better reflect the quality perceived by the **HVS** and is discussed later in the chapter. The total distortion  $D_{Tot}$  to be minimized is expressed by (4.20). In the particular case of  $\Psi_{i_t} = 1, \forall i_t$ , the minimized distortion is chosen to be the classical **MSE**.

$$D_{Tot}(\vec{p}) = \sum_{t=1}^T \sum_{i=1}^N \Psi_{i_t} D_{i_t}(\vec{p}) \quad (4.20)$$

The temporal distortion propagation model used hereafter defines the distortion  $D_{i_t}$  of a **CU**  $i_t$  as the weighted sum of its local distortion  $d_{i_t}$  and the distortion  $D_{j_{tref}}$  propagated from its reference **CU**  $j_{tref}$ . The chosen propagation formula is given by (4.21).

$$D_{i_t}(\vec{p}) = d_{i_t}(\vec{p}_{i_t}) + p_{i_t} \underbrace{\sum_{j_{tref} \in Ref(i_t)} r_{j_{tref}, i_t} D_{j_{tref}}(\vec{p})}_{\eta_{i_t}}. \quad (4.21)$$

$Ref(i_t)$  is the set of reference **CUs** used for motion compensation,  $p_{i_t}$  is the probability of a **CU** to be Inter coded and  $r_{j_{tref}, i_t}$  the pixel surface ratio involved in the motion compensation to go from spatial position of  $j_{tref}$  to spatial position of  $i_t$ . An illustration of this ratio is given in Fig. 4.3.

$d_i(\vec{p}_i)$  is the local distortion, i.e. the distortion that only depends on  $\vec{p}_i$ , the coding parameters applied to encode the CU  $i_t$ .  $\eta_i$  is the amount of distortion from reference samples propagated into CU  $i_t$  after motion compensation. For writing simplification, distortion functions are expressed in the following without parameters, i.e.  $d_i(\vec{p}_i) = d_i$ , unless a particular coding parameter is necessary for understanding.

The main drawback of this model is to only consider Inter/Intra coding, i.e. modes involving the transmission of a residue, and to ignore the Skip coding mode where no residue is transmitted. To consider the Skip mode, we introduce  $c_i$  as the probability of the CU  $i_t$  to be coded in Inter/Intra mode and  $(1 - c_i)$  as the probability of the CU to be coded in Skip mode.

A large residue should lead to a high probability for Intra/Inter mode, while a large quantization step  $\Delta$  should lead to a high probability for Skip mode. Hence,  $c_i$  is proposed to be defined as:

$$c_i = \frac{12\sigma_{src_i}^2}{12\sigma_{src_i}^2 + \Delta_i^2} \quad (4.22)$$

$$c_i \rightarrow \begin{cases} 1 & \text{if } 12\sigma_{src_i}^2 \gg \Delta_i^2 \\ \frac{12\sigma_{src_i}^2}{\Delta_i^2} & \text{if } 12\sigma_{src_i}^2 \ll \Delta_i^2 \end{cases} \quad (4.23)$$

where  $\sigma_{src_i}^2$  is the variance of predicted residue obtained by motion compensation between source samples, and  $\Delta_i$  is the quantization step used to code the CU  $i_t$ . The behavior analysis of (4.22) is given in (4.23). If  $12\sigma_{src_i}^2 \ll \Delta_i^2$ ,  $c_i$  tends toward  $12\sigma_{src_i}^2/\Delta_i^2$ , which can be approximated as 0. It is intuitively adequate that a large residue leads to a high probability for Intra/Inter mode, while a large quantization step leads to a high probability for Skip mode. The choice of Skip probability model is also a mathematical workaround to draw a more robust solution, as described in Section 4.4.

In order to include  $c_i$  in the propagation model in (4.21), we first define  $D_i^C$  and  $D_i^S$  in equation (4.24) as the distortion of a CU  $i_t$  to be coded in Inter/Intra and Skip, respectively.

$$D_i^C = d_i + p_i \eta_i, \quad D_i^S = \sigma_{src_i}^2 + p_i \eta_i \quad (4.24)$$

According to (4.24), the propagation model in (4.21) is turned into (4.25).

$$D_i = c_i D_i^C + (1 - c_i) D_i^S = c_i d_i + (1 - c_i) \sigma_{src_i}^2 + p_i \eta_i \quad (4.25)$$

By developing the total distortion  $D_{Tot}$  from (4.20) and using the temporal propagation defined in (4.25), we can express the total distortion as a weighted sum of local distortions (4.26). The details of the calculation are explained in Appendix B.1.

$$D_{Tot} = \sum_{t=1}^T \sum_{i=1}^N \left( c_{i_t} d_{i_t} + (1 - c_{i_t}) \sigma_{src_{i_t}}^2 \right) U_{i_t}, \quad (4.26)$$

where  $U_{i_t}$  is the accumulation factor defined by

$$U_{i_t} = \Psi_{i_t} + \sum_{\tau=t}^T \sum_{i_\tau} \sum_{i_{\tau-1} \in Ref(i_\tau)} \dots \sum_{i_{t+1} \in Ref(i_{t+2})} \left( \Psi_{i_\tau} p_{i_\tau} r_{i_{\tau-1}, i_\tau} p_{i_{\tau-1}} \dots p_{i_{t+1}} r_{i_t, i_{t+1}} \right). \quad (4.27)$$

$U_{i_t}$  can be semantically interpreted as the proportion of the local distortion  $d_{i_t}$  that impacts the total distortion  $D_{Tot}$ . It also has the property of being recursively computable as shown in Appendix B.2. Its final expression is

$$\begin{cases} U_{i_T} &= \Psi_{i_T} \\ U_{i_t} &= \Psi_{i_t} + \sum_{i_{t+1}} p_{i_{t+1}} r_{i_t, i_{t+1}} U_{i_{t+1}}. \end{cases} \quad (4.28)$$

The main interest of the formulation in (4.26) is to isolate local distortions  $d_{i_t}$  that depend only on local coding parameters  $\vec{p}_{i_t}$ . Consequently, the problem stated in (4.20) can be solved by locally setting coding parameters that optimize the overall R-D efficiency of a GOP. The application case of adaptive local quantization and its related analytical solution are both described in the next section.

### 4.3 RDSTQ Algorithm

In the following, we investigate the proposed HEVC video optimization solution aiming to estimate optimal delta quantizers at CU-level.

### 4.3.1 Local Quantization Problem

The coding parameters of interest in this section are the local quantization parameters, noted  $q_i$  for CU  $i_t$ . For ease of reading, the set of local quantizers for all CUs in a GOP is noted  $\{q\}$ , with  $\{q\} = \{q_{i_t}\}_{i=1..N, t=1..T}$ . The overall constrained minimization problem is then:

$$\begin{aligned} \{q^*\} &= \arg \min_{\{q\}} D_{Tot}(\{q\}) \\ &= \arg \min_{\{q\}} \sum_{t=1}^T \sum_{i=1}^N \Psi_{i_t} D_{i_t}(\{q\}) \\ \text{s.t. } &\sum_{t=1}^T \sum_{i=1}^N R_{i_t}(\{q\}) = R_{Tot}. \end{aligned} \quad (4.29)$$

A simplification is made for solving the problem and achieve an analytical solution. This simplification is to consider that the Inter probability  $p_i$  and the references distortion  $\eta_i$  that affects  $i_t$  are independent of  $q_i$ . According to (4.28),  $U_{i_t}$  is then also independent of  $q_i$ . Intuitively, the local quantizer should affect the Inter probability. However, its influence is negligible in most cases, i.e. Intra (or Inter) cost is much smaller than Inter (or Intra) cost and  $p_{i_t} \rightarrow 0$  (or  $p_{i_t} \rightarrow 1$ ).

The non-Skip probability  $c_{i_t}$  and the local distortion  $d_{i_t}$  both depend on the local quantization parameter  $q_{i_t}$ . The necessary condition to find the minimum of  $D_{Tot}$  is determined by the condition of all the derivatives equal to zero  $\forall i \in \{1, \dots, N\}, \forall t \in \{1, \dots, T\}$ :

$$\frac{\partial D_{Tot}}{\partial \Delta_{i_t}} = \left( \frac{\partial d_{i_t}}{\partial \Delta_{i_t}} c_{i_t} + \frac{\partial c_{i_t}}{\partial \Delta_{i_t}} d_{i_t} - \frac{\partial c_{i_t}}{\partial \Delta_{i_t}} \sigma_{src_{i_t}}^2 \right) U_{i_t} \quad (4.30)$$

(4.30) can be simplified into (4.32) as described below:

$$\frac{\partial c_{i_t}}{\partial \Delta_{i_t}} = \frac{-24\sigma_{src_{i_t}}^2 \Delta_{i_t}}{144\sigma_{src_{i_t}}^4 + \Delta_{i_t}^4 + 24\sigma_{src_{i_t}}^2 \Delta_{i_t}^2} \approx 0 \quad (4.31)$$

We justify this approximation by observing that whatever the values of  $\sigma_{src_{i_t}}$  and  $\Delta_{i_t}$ , denominator is always much larger than numerator. This simplification leads to

$$\frac{\partial D_{Tot}}{\partial \Delta_{i_t}} = \frac{\partial d_{i_t}}{\partial \Delta_{i_t}} c_{i_t} U_{i_t} \quad (4.32)$$

### 4.3.2 Analytical Solution

We depict hereafter the analytical solution which makes use of (4.32) to solve the constrained problem described in (4.29). The analytical solution results in obtaining optimal delta quantizers  $dQP$  for all CUs, while maintaining the GOP total rate identical to a configuration without AQ.

Problem (4.29) is modeled thanks to the Lagrangian multiplier method with  $\lambda$  the Lagrangian multiplier. The new function to be minimized is the total R-D cost  $J_{Tot}$  defined in (4.33).

$$J_{Tot} = D_{Tot} + \lambda \left( \sum_{t=1}^T \sum_{i=1}^N R_{i_t} - R_{Tot} \right) \quad (4.33)$$

The necessary condition to find the minimum of  $J_{Tot}$  is that all the partial derivatives with respect to quantization parameters are equal to zero  $\forall i \in \{1, \dots, N\}, \forall t \in \{1, \dots, T\}$ :

$$\frac{\partial J_{Tot}}{\partial \Delta_{i_t}} = \frac{\partial D_{Tot}}{\partial \Delta_{i_t}} + \lambda \frac{\partial}{\partial \Delta_{i_t}} \sum_{t=1}^T \sum_{i=1}^N R_{i_t} = 0 \quad (4.34)$$

We express the rate  $R_{i_t}$  of a CU  $i_t$  as a function of  $R_{i_t}^C$  and  $R_{i_t}^S$  as the rates of a CU  $i_t$  to be coded in Inter/Intra and Skip, respectively. However, the rate of skipped CUs is theoretically equal to zero. Thus, we have

$$R_{i_t} = c_{i_t} R_{i_t}^C + (1 - c_{i_t}) \underbrace{R_{i_t}^S}_{\approx 0} = c_{i_t} R_{i_t}^C. \quad (4.35)$$

In order to keep formula easy to read, in the following we simply write  $c_{i_t} R_{i_t}^C = c_{i_t} R_{i_t}$ . If we suppose the *independence of rates*, which is discussed and validated in Section 4.4, (4.34) is simplified into (4.36).

$$\frac{\partial J_{Tot}}{\partial \Delta_{i_t}} = \frac{\partial d_{i_t}}{\partial \Delta_{i_t}} c_{i_t} U_{i_t} + \lambda \frac{\partial R_{i_t}}{\partial \Delta_{i_t}} c_{i_t} = 0 \quad (4.36)$$

The R-D Shannon bound is injected into (4.36) to obtain the optimal  $\lambda$  as (4.37). Developments are detailed in Appendix B.3.

$$\lambda = 2 \ln(2) U_{i_t} D_{i_t} \quad (4.37)$$

To simplify writing, we define  $\lambda'$  as

$$\lambda' = \frac{\lambda}{2 \ln(2)} \quad (4.38)$$

We then have

$$\log_2(\lambda') = \log_2(U_{i_t} D_{i_t}). \quad (4.39)$$

The RDSTQ aims to keep unchanged the average GOP bitrate  $R_{Tot}$ . It is achieved if the total rate obtained through RDSTQ is equal to the total rate obtained with a unique

quantization step applied to all CUs in the GOP. In next developments from (4.40) to (4.48), we exhibit the total GOP rate and further apply the rate constraint.

By summing the log values weighted according to non-Skip probability  $c_i$  on both sides of (4.39) over all CUs of the GOP, we have

$$\log_2(\lambda') \underbrace{\sum_{t=1}^T \sum_{i=1}^N c_i}_{=N_{Tot}} = \sum_{t=1}^T \sum_{i=1}^N c_i \log_2(U_i D_i), \quad (4.40)$$

$$\log_2(\lambda') = \frac{1}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N c_i \log_2(U_i D_i). \quad (4.41)$$

We consider a given CU  $k_\tau$  and mix (4.39) with (4.41):

$$\frac{1}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N c_i \log_2(U_i D_i) = \log_2(U_{k_\tau} D_{k_\tau}) \quad (4.42)$$

In order to remove the cumbersome sum of all local distortion logarithms, we compute the  $\frac{2R_{Tot}}{N_{Tot}}$  using the R-D Shannon bound.

$$\frac{2R_{Tot}}{N_{Tot}} = \frac{2}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N c_i R_i \quad (4.43)$$

$$\frac{2R_{Tot}}{N_{Tot}} = \frac{\sum_{t=1}^T \sum_{i=1}^N c_i (\log_2(\sigma_i^2) - \log_2(D_i))}{N_{Tot}}. \quad (4.44)$$

The term depending on all local distortions can be eliminated by using (4.42) and (4.44) in order to obtain (4.45).

$$\frac{2R_{Tot}}{N_{Tot}} = -\log_2(U_{k_\tau}) - \log_2(D_{k_\tau}) + \frac{\sum_{t=1}^T \sum_{i=1}^N c_i \log_2(\sigma_i^2 U_i)}{N_{Tot}} \quad (4.45)$$

This result is necessary for applying the rate constraint. The high bitrate approximation (4.46) is injected into (4.45) in order to bring up the quantization parameter  $QP_{k_\tau}$  as follows:

$$D_{k_\tau} = \frac{\Delta_{k_\tau}^2}{12} = \frac{2^{\frac{QP_{k_\tau}-4}{3}}}{12} \quad (4.46)$$

$$\log_2(D_{k_\tau}) = \frac{QP_{k_\tau}-4}{3} - \log_2(12) \quad (4.47)$$

$$\frac{2R_{Tot}}{N_{Tot}} = -\left(\frac{QP_{k_\tau} - 4}{3} - \log_2(12)\right) - \log_2(U_{k_\tau}) + \frac{\sum_{t=1}^T \sum_{i=1}^N c_i \log_2(\sigma_i^2 U_i)}{N_{Tot}} \quad (4.48)$$

To estimate delta quantizers, we consider the case of a **GOP** encoded with a unique quantization parameter, named  $QP$ , and develop another expression of  $\frac{2R_{Tot}}{N_{Tot}}$  as follows:

$$\frac{2R_{Tot}}{N_{Tot}} = \frac{2}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N c_i R_i \quad (4.49)$$

$$\frac{2R_{Tot}}{N_{Tot}} = \frac{-1}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N c_i (\log_2(D_i) - \log_2(\sigma_i^2)) \quad (4.50)$$

We inject (4.47) and simplify  $QP_{k_\tau} = QP, \forall k, \tau$  to obtain

$$\frac{2R_{Tot}}{N_{Tot}} = \frac{4 - QP}{3} + \log_2(12) + \frac{\sum_{t=1}^T \sum_{i=1}^N c_i \log_2(\sigma_i^2)}{N_{Tot}} \quad (4.51)$$

Since the **AQ** is designed to be neutral with regards to the average **GOP** rate, and assuming residue variances are kept unchanged, we can mix (4.48) and (4.51) to exhibit the optimal delta quantizer  $dQP_{k_\tau} = (QP_{k_\tau} - QP)$  of the **CU**  $k_\tau$ .

$$dQP_{k_\tau} = -str \left( \log_2(U_{k_\tau}) - \frac{1}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N c_i \log_2(U_i) \right) \quad (4.52)$$

We note that  $str$  is called the *strength*. Its theoretical optimal value is  $str = 3$ , coming from the relationship between  $QP_{k_\tau}$  and  $\Delta_{k_\tau}$ . Increasing or decreasing this value may stretch the quantizers dynamic range and thus modify the **R-D** efficiency and the **Target Bitrate Deviation (TBD)**. Setting the strength to a large value may drastically increase the **TBD** and we observed empirically that setting  $str = 2$  for all experiments is a good trade-off between **R-D** gains and **TBD**.

The **RDSTQ** algorithm is based on the temporal propagation model presented in Section 4.2. In considering such a model into the local quantization problem presented in (4.29), we are able to efficiently improve the overall **R-D** coding efficiency. Thanks to the analytical solution, optimal delta quantizers are easily estimated based on a look-ahead process and do not require extensive multi-pass analysis. Moreover, as shown in the next section, the range of delta quantizers is bounded and controllable.

## 4.4 Hypotheses Validation

This section aims to provide justification and validation for some of the simplifications or assumptions made in the previous section. It is divided into five subsections. First, the *independence of rates* hypothesis considered during the analytical solution development is validated through experiments. Second, the estimation of inter probability is discussed with the support of ground truth data extracted from off-line encodings. Third, the Skip mode probability is discussed. Then, the look-ahead which provides necessary input parameters for the RDSTQ to compute delta quantizer is discussed, with details of its implementations into x265 and HM. Most notably, we demonstrate that the range of delta quantizers is bounded and can be controlled beforehand.

### Proof of independence of rates

We give in this section some insights for the independence of rates assumption, along with experimental validation. From 4.2, we can write 4.53.

$$\begin{aligned}
 D_i &= \underbrace{c \mathbb{E} \left[ (X_i - X_{ref})^2 \right]}_{=d_i} 2^{-2R_i} + c D_{ref} 2^{-2R_i} \\
 &= d_i + \frac{c \mathbb{E} \left[ (X_i - X_{ref})^2 \right] 2^{-2R_i}}{\mathbb{E} \left[ (X_i - X_{ref})^2 \right]} D_{ref} \\
 &= d_i + \frac{d_i}{\mathbb{E} \left[ (X_i - X_{ref})^2 \right]} D_{ref}
 \end{aligned} \tag{4.53}$$

If we assume no dependencies in intra coding, i.e.  $D_i = d_i$ , and that the probability of a CU to be coded in Inter is equal to  $p_i$ , the distortion can be expressed as

$$D_i = d_i \left( 1 + \frac{p_i}{\mathbb{E} \left[ (X_i - X_{ref})^2 \right]} D_{ref} \right). \tag{4.54}$$

According to the Shannon R-D function, the rate  $R_i$  of the CU  $i$  is expressed as in (4.55).

$$R_i = -\frac{1}{2} \log_2 \left( \frac{D_i}{\sigma_i^2} \right) \tag{4.55}$$



We consider equations (4.1) and (4.54) to express the rate  $R_i$  as (4.56).

$$\begin{aligned}
R_i &= -\frac{1}{2} \log_2 \left( \frac{d_i \left( 1 + \frac{p_i D_{ref}}{\mathbb{E}[(X_i - X_{ref})^2]} \right)}{\mathbb{E}[(X_i - X_{ref})^2] + p_i D_{ref}} \right) \\
&= -\frac{1}{2} \log_2 \left( \frac{\frac{d_i}{\mathbb{E}[(X_i - X_{ref})^2]} (\mathbb{E}[(X_i - X_{ref})^2] + p_i D_{ref})}{\mathbb{E}[(X_i - X_{ref})^2] + p_i D_{ref}} \right) \\
&= -\frac{1}{2} \log_2 \left( \frac{d_i}{\mathbb{E}[(X_i - X_{ref})^2]} \right)
\end{aligned} \tag{4.56}$$

Consequently, since  $d_i$  and  $\mathbb{E}[(X_i - X_{ref})^2]$  does not depend on  $D_{ref}$ , we can assume the rate  $R_i$  is independent from  $D_{ref}$ .

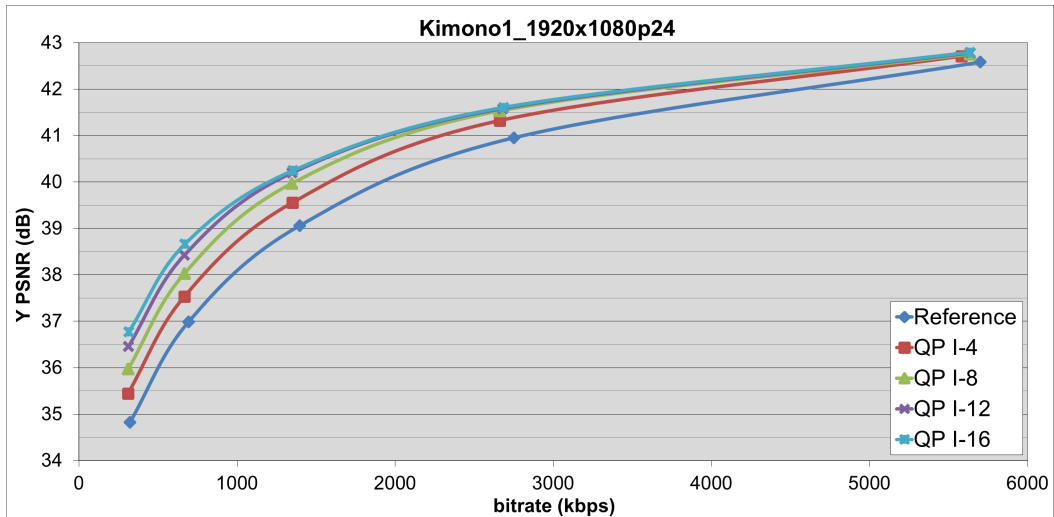
An experiment was conducted in order to evaluate the correctness of *independence of rates* assumption. To do so, **R-D** curves of non-Intra frames in a **GOP** are generated with a fixed **QP** configuration while different **QP** offsets are set on the Intra frame, in the set  $[0; -4; -8; -12; -16]$ . The *Reference R-D* curve corresponds to the 0 offset case. The experiment was conducted into **HM** encoder with Intra coding disabled in non-Intra frames. The objective of this experiment is to confirm that increasing quality on Intra frame shifts the **R-D** curves of depending frames toward less distortion without rate deviation.

Fig. 4.4 shows experimental results for *Kimono* (a) and *Cactus* (b) video sequences in **RA** configuration with hierarchical 3-B. These curves show that **R-D** points are aligned along the rate axis whatever the **QP** offset on the Intra frames. Consequently, temporal dependency between **CUs** only impacts distortions and not rates. This validates the *independence of rates* assumption applied in Section 4.3.2.

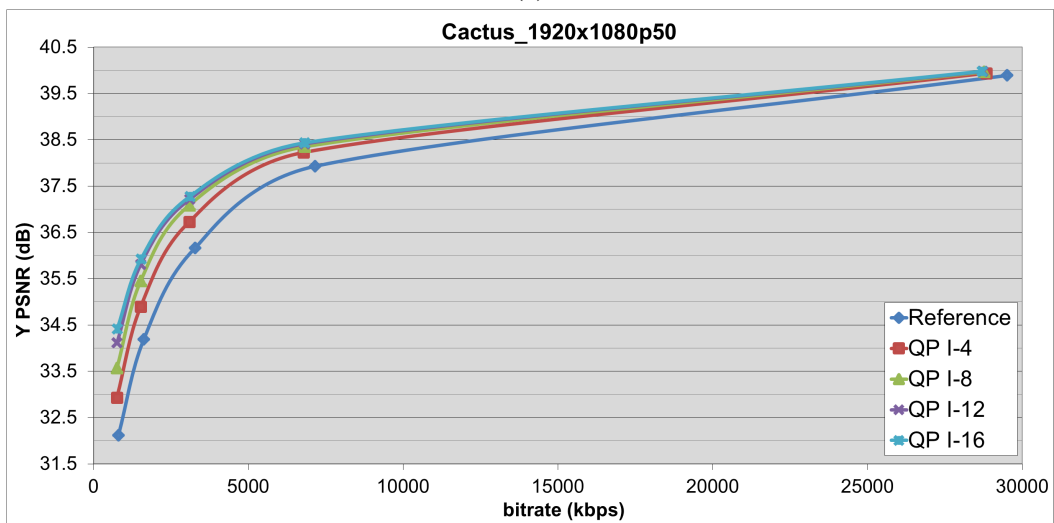
### Inter Probability Estimation

In this section, we present the considered Inter probability estimators. The Inter probability have been introduced in Section 4.2 in order to model the distortion according to temporal dependency.  $\omega_i^{Intra} > 0$  and  $\omega_i^{Inter} > 0$  are defined as the **SATD** prediction costs of Intra and Inter modes, respectively. The **SATD** costs are estimated in the look-ahead analysis. The probability of the Inter prediction mode is defined as a function of the ratio  $r_i = \omega_i^{Intra} / \omega_i^{Inter}$ . The Inter probability estimator used in previous sections is given by

$$p_i = 1 - \min\left(1; \frac{1}{r_i}\right). \tag{4.57}$$



(a)



(b)

Fig. 4.4 R-D curves of non-Intra frames according to Intra QP offsets with (a) *Kimono* and (b) *Cactus* sequences

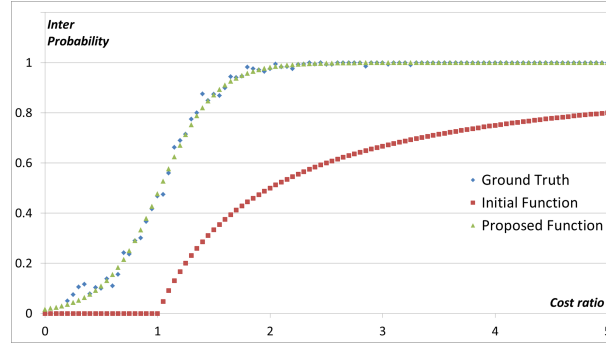


Fig. 4.5 Inter Probability  $p_{i_t}$  according to cost ratio  $r_{i_t}$  estimated by (4.57) for the Initial function and (4.59) for the proposed function.

This formula implies that if **SATD** costs are equivalent, i.e.  $r_{i_t} = 1$ , Inter probability should be null and there is no propagation, i.e.  $p_{i_t} = 0$ . However, close Intra/Inter prediction costs should intuitively lead to equiprobable Intra and Inter modes. Moreover, neither theoretical nor experimental proof of the correctness of (4.57) has been given. We propose hereafter to improve the Inter probability estimation.

Based on statistical inference, the Inter probability  $p$  is estimated, from an off-line **RDO** analysis, as the Likelihood function defined in (4.58).

$$p = L(r|mode) \propto P(mode = Inter|r) \quad (4.58)$$

Fig. 4.5 compares both functions  $p_{i_t}$  (4.57) and  $p$ .  $r_{i_t}$  is the prior information known beforehand while the event for a **CU**  $i_t$  to be Inter coded is the evidence. We observe in Fig. 4.5 that (4.57) is quite far from the ground truth. Consequently, another function defined by (4.59) is proposed, which is a sigmoid distribution fitting the ground truth curve.

$$p_{i_t} = \frac{1}{1 + a \exp^{-br_{i_t}}} \quad (4.59)$$

The (4.59) function is plotted on Fig. 4.5. Data shown on the Fig. are extracted from the first **GOPs** of each sequence that belongs to *classD*. The performance of this function is discussed in Section 4.5.  $a$  and  $b$  are model parameters estimated from our experiments to  $a = 0.5651$  and  $b = 3.6064$ .

### Skip probability justification

In order to achieve an analytical solution, the high-rate assumption is used to estimate the quantizer from the distortion model. Despite its mathematical tractability, such assumption

is debatable and does not stand for low bitrates. In order to be more robust to different use cases, we consider the distortion formula of *Xu et al.* proposed in [XJGZ07]. This formula has been discussed in Section 1.2.3 and is written in (4.60).

$$D_{i_t} = \frac{\sigma_{i_t}^2 \Delta_{i_t}^2}{12\sigma_{i_t}^2 + \Delta_{i_t}^2} \quad (4.60)$$

This model makes it difficult to extract the delta quantizer based on the distortion. However, (4.60) is strictly equal to the formula in (4.61).

$$D_{i_t} = \frac{\Delta_{i_t}^2}{12} \times \underbrace{\frac{12\sigma_{i_t}^2}{12\sigma_{i_t}^2 + \Delta_{i_t}^2}}_{c_{i_t}} \quad (4.61)$$

We observe that the chosen non-Skip probability  $c_{i_t}$  scales the distortion, using the high-rate assumption, into the desired distortion. Finally, using developments described in (4.30), (4.31) and (4.32) allow to keep the analytical solution simple while using a more robust distortion model.

### Look-Ahead Design

In this section we give more insights about the look-ahead implementation in x265 and the look-ahead we developed in *HM*.

The analytical solution explained above provides the optimal set of local quantizers to the encoder from an R-D standpoint. However, several input parameters, depending on source characteristics, are required prior to compute these quantizers. Look-ahead is a common sub-process designed to estimate such parameters, based on a pre-analysis which mimics the encoder behavior. Due to algorithm requirements, a look-ahead was used in both x265 and *HM* implementations.

The x265 encoder already encloses an efficient look-ahead. Videos are first down-sampled in order to divide the height and width of original pictures by 2. Low-resolution frames are partitioned into 8x8 blocks and each block is analyzed in Intra and Inter modes. Intra and Inter modes are compared based on SATD costs. For both Intra mode and Inter motion estimation, fast analysis is used and based on dichotomous approaches.

In the *HM* encoder, no look-ahead is currently available. Taking advantage of available tools in the *HM*, we successfully emulated a look-ahead to extract the necessary information. Our look-ahead is configured as follow:

- No QuadTree: only 16x16 CUs are used

- All modes are analyzed in [SATD](#) and use source signal for reference prediction
- No bitstream is actually written since no reconstructed data are required
- All necessary values are stored in a look-ahead file

The [HM](#) look-ahead is finally achieved by parsing this look-ahead file. We assume the proposed look-ahead in [HM](#) is more accurate, in terms of correlation with the actual encoder decisions, compared to the x265 one. This assumption comes from the x265 look-ahead working on down-sampled source pictures. Consequently, better [R-D](#) efficiency is observed for the [HM](#), as shown in Section 4.5. The computational overhead of this pre-analysis is around 30% of the [HM](#) encoding complexity. This complexity increase is usually very manageable for real industrial implementations, first thanks to the efficient use of multi-threading, and second by leveraging on look-ahead information to speed up the main encoding decisions.

### Quantizer dynamic range

In this section, the dynamic range of delta quantizers is analyzed. We prove that the dynamic range of delta quantizers obtained through the model are bounded. The output dynamic range of the delta quantizers is predictable before the encoding process. This property helps to prevent from any conformance issue or boundary defect.

Let assume a sequence is temporally stable, i.e. probability of Inter mode is equal for all [CUs](#) with identical spatial positions in different frames. We have seen in Section 4.2 that

$$\begin{cases} U_{j_T} &= \Psi_{j_T} \\ U_{j_{t-1}} &= \sum_{i_t} p_{i_t} r_{j_{t-1}, i_t} U_{i_t} + \Psi_{j_{t-1}}. \end{cases} \quad (4.62)$$

For the sake of simplicity, let  $\Psi_{i_t} = 1 \forall i, t$ . If we assume all Inter mode probabilities in the same spatial area to be equal to  $p$ , i.e. within the temporally stable part of a picture, then we obtain

$$U_{i_t} = \sum_{k=t}^T p^{T-k}. \quad (4.63)$$

Under the assumption that  $p_{i_t} = p, \forall i, t$ ,  $U_{i_t}$  is a geometrical series. Knowing that  $p \in [0...1]$ , we finally obtain

$$U_{max} = \lim_{(t,T) \rightarrow (0,\infty)} U_{i_t} = \frac{1}{1-p}. \quad (4.64)$$

Table 4.2 Theoretical number of frames required for  $U$  convergence based on  $p$  values and related  $U_{max}$

$p$ value	$N_{conv}$	$U_{max}$
0.1	3	1.11
0.2	4	1.25
0.3	5	1.43
0.4	8	1.67
0.5	9	2.00
0.6	13	2.50
0.7	17	3.33
0.8	31	5.00
0.9	73	10.0
1.0	NaN	$T$

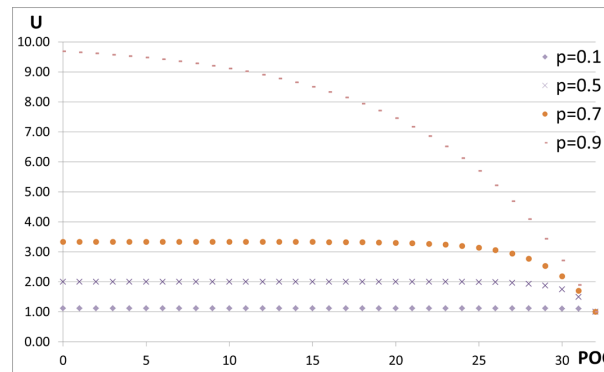


Fig. 4.6  $U$  value evolution within a **GOP** of 32 frames for different values of  $p$ , with  $\Psi_{i_t} = 1 \forall i, t$ .

This equation suggests that, by design,  $U_{i_t}$  converges toward a maximum value  $U_{max}$ , that depends on source characteristics, under the assumption that  $T$  is large enough. We also notice that the lower the value of  $p$ , the faster the  $U_{i_t}$  convergence. We report in Table 4.2, for a given value of  $p$ , the maximum achievable weight  $U_{max}$  reached once the number of frames in the **GOP** equals or exceeds  $N_{conv}$ . The convergence is assumed with a two decimal places precision.

We can see the consequence of such convergence on Figure 4.6. If one increases  $T$  value, as long as the sequence is temporally stable, reference frames ultimately have an equal level of importance within the **GOP**. We note that such convergence is most likely to occur for small  $p$  value, i.e. sequences difficult to predict temporally.

In the special case of  $p = 1$ , the  $U_{max}$  value only depends on the **GOP** length:  $U_{max} = T$ . Once  $U_{max}$  is estimated, the dynamic range of delta quantizer  $rng_{dQP}$  is given by

$$rng_{dQP} = -str(\log_2(U_{max})) \quad (4.65)$$

with  $str$  being the strength mentioned in Section 4.3.2. Based on this formula, one may choose to control the dynamic of the delta quantizers by directly modifying the strength value. In our experiments  $str = 2$ .

## 4.5 Performance Evaluation

This section aims to validate the coding efficiency of the proposed solution, assess the **TBD** reduction and confirm the expected behavior of the model. First, the proposed probability described in Section 4.4, and the Skip probability model are evaluated. Second, the rate distribution between frames of the **GOP** is observed. Third, the positive impact of  $\Psi$  function on overspent rate situation is confirmed. Finally, the method is compared to state-of-the-art methods, thanks to the proposed **HM** implementation.

The x265 software **HEVC** encoder [x26] is used in the experiments. The **HM** encoder [MRB<sup>+</sup>14] is also used to confirm results in a different encoder. The **CTC** defined by the **JCT-VC** [Bos13] have been followed. The videos are encoded in **RA** coding configuration, with hierarchical 3-B, for five **QP** values  $\in \{22, 27, 32, 37, 42\}$ . The **QP** value of 42 was added to highlight the Skip mode influence since it is statistically more used at low bitrate.

When no psycho-visual function is considered, i.e.  $\Psi_i = 1, \forall i, t$ , the model is simply called **Rate Distortion Temporal Quantization (RDTQ)**, since spatial criteria is ignored, and we focus on the **BD-BR** computed using the **PSNR** metric. Otherwise, the model is called **RDSTQ** and we focus on the **BD-BR** computed using the **SSIM** metric, that is better correlated with **HVS** perception of quality. In the case of **RDSTQ** we set  $\Psi_i = 1/\sigma_i^2$ , with  $\sigma_i^2$  being the local variance of source luminance pixels of the block  $i_t$ . *Yeo et al.* [YTT13a] proved that weighting a **MSE** distortion by the inverse of local pixel block variance specifically optimizes the **SSIM** metric, which explains our choice of  $\Psi$  function.

### Coding Efficiency

Coding performance is measured using the **BD-BR** metric [Bjo01a]. A negative **BD-BR** value reflects the percentage of bitrate savings achieved at equivalent YUV distortion, between the anchor and the proposed solution. The **BD-BR** results and the corresponding target bitrate

Table 4.3 Coding efficiency over no local quantization in *HM* for *RDTQ*.

BD-BR PSNR	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-10.35%	-10.16%	-14.15%	-14.02%
	Class B	-7.55%	-7.57%	-12.02%	-12.97%
	Class C	-15.24%	-15.25%	-19.01%	-19.20%
	Class D	-13.95%	-13.48%	-16.52%	-16.08%
	Class E	-14.08%	-13.30%	-22.67%	-21.03%
	<b>Average</b>	<b>-12.08%</b>	<b>-11.83%</b>	<b>-16.58%</b>	<b>-16.51%</b>
	<b>Best</b>	<b>-21.04%</b>	<b>-21.51%</b>	<b>-26.57%</b>	<b>-26.68%</b>
	<b>Worst</b>	<b>-3.36%</b>	<b>-7.65%</b>	<b>-7.59%</b>	<b>-7.86%</b>
BD-BR SSIM	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-13.63%	-12.98%	-24.18%	-22.94%
	Class B	-7.30%	-7.33%	-18.36%	-19.57%
	Class C	-18.43%	-18.43%	-27.84%	-28.11%
	Class D	-19.62%	-18.82%	-28.02%	-27.05%
	Class E	-8.92%	-8.09%	-21.39%	-19.95%
	<b>Average</b>	<b>-13.48%</b>	<b>-13.10%</b>	<b>-23.76%</b>	<b>-23.57%</b>
	<b>Best</b>	<b>-25.92%</b>	<b>-25.93%</b>	<b>-36.88%</b>	<b>-36.20%</b>
	<b>Worst</b>	<b>-1.48%</b>	<b>-2.16%</b>	<b>-12.48%</b>	<b>-16.33%</b>
TBD	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	20.32%	5.86%	43.82%	12.40%
	Class B	22.12%	12.92%	64.98%	31.98%
	Class C	16.69%	5.36%	41.46%	12.93%
	Class D	23.53%	7.65%	50.56%	15.09%
	Class E	44.65%	23.84%	113.77%	42.87%
	<b>Average</b>	<b>24.78%</b>	<b>11.10%</b>	<b>62.33%</b>	<b>23.63%</b>

deviations, *TBD*, averaged on the considered *QP* values are presented in Table. 4.3 and Table. 4.4 for *HM* and in Table. 4.5 and Table. 4.6 for x265.

The anchors are respectively the x265 and *HM* encoders without *AQ* algorithm. The two Inter probability models, defined in (4.57) and (4.59) are compared and respectively named *Initial probability* and *Proposed probability*.

From Table. 4.3 and Table. 4.4, we can observe higher bitrate savings for the Proposed probability (4.59) over the Initial probability (4.57), whether the Skip mode consideration is enabled or not. The Proposed probability (4.59) saves in average  $-4.5\%$  PSNR-based *BD-BR* compared to the Initial probability with *RDTQ* and  $-8.61\%$  SSIM-based *BD-BR* compared to the Initial probability with *RDSTQ*. When Skip is considered, performance suffer from an average bitrate increase between  $0.07\%$  and  $0.25\%$  for *RDTQ* and between  $0.54\%$  and  $0.63\%$  for *RDSTQ*.



Table 4.4 Coding efficiency over no local quantization in *HM* for *RDSTQ*.

BD-BR PSNR	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-11.47%	-11.38%	-13.72%	-13.56%
	Class B	-9.03%	-8.81%	-11.50%	-11.61%
	Class C	-15.54%	-15.44%	-18.48%	-18.46%
	Class D	-14.17%	-13.97%	-15.79%	-15.64%
	Class E	-16.82%	-16.52%	-22.03%	-20.82%
	<b>Average</b>	<b>-13.19%</b>	<b>-13.00%</b>	<b>-16.01%</b>	<b>-15.78%</b>
	<b>Best</b>	<b>-22.58%</b>	<b>-22.52%</b>	<b>-26.41%</b>	<b>-26.41%</b>
	<b>Worst</b>	<b>-5.29%</b>	<b>-5.40%</b>	<b>-6.64%</b>	<b>-6.42%</b>
BD-BR SSIM	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-19.26%	-18.76%	-28.02%	-26.45%
	Class B	-13.42%	-13.26%	-22.30%	-23.24%
	Class C	-21.97%	-21.62%	-30.37%	-30.26%
	Class D	-23.65%	-22.67%	-30.93%	-29.28%
	Class E	-14.28%	-13.37%	-23.78%	-21.81%
	<b>Average</b>	<b>-18.38%</b>	<b>-17.84%</b>	<b>-26.89%</b>	<b>-26.26%</b>
	<b>Best</b>	<b>-30.38%</b>	<b>-29.89%</b>	<b>-40.90%</b>	<b>-39.65%</b>
	<b>Worst</b>	<b>-10.71%</b>	<b>-9.61%</b>	<b>-18.65%</b>	<b>-17.77%</b>
TBD	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	10.28%	4.39%	32.65%	9.57%
	Class B	9.89%	5.38%	49.57%	19.43%
	Class C	9.52%	3.39%	33.93%	8.26%
	Class D	18.96%	4.09%	46.51%	7.28%
	Class E	13.11%	4.10%	75.56%	16.54%
	<b>Average</b>	<b>12.40%</b>	<b>4.33%</b>	<b>47.87%</b>	<b>12.67%</b>

The *TBD* shows 2 to 4 times higher deviation when using the Proposed probability compared to Initial probability. Indeed, the Proposed probability induces larger propagation and consequently smaller delta quantizers, i.e. more rates, on reference frames. The consideration of Skip probability efficiently reduces the *TBD*, and then the Proposed probability provides similar *TBD* as the Initial probability while maintaining *BD-BR* gains. Finally, the average *TBD* for *RDTQ* and *RDSTQ* is in average equal to 23.63% and 12.67%, respectively.

Observations from the x265 experiments depicted in Table. 4.5 and Table. 4.6 tend toward similar conclusions. The average bitrate savings for *RDTQ* and *RDSTQ* are respectively  $-1.43\%$  *PSNR*-based *BD-BR* and  $-3.46\%$  *SSIM*-based *BD-BR*. The *TBD* is however reduced compared to the Initial probability without Skip consideration. The *TBD* is reduced from 37.37% to 17.43% with *RDTQ* and from 15.16% to 10.77% with *RDSTQ*.

As desired, the Proposed probability improves the coding efficiency while the Skip mode consideration efficiently reduces the *TBD*. We demonstrate in these experiments that the

Table 4.5 Coding efficiency over no local quantization in x265 for RDTQ.

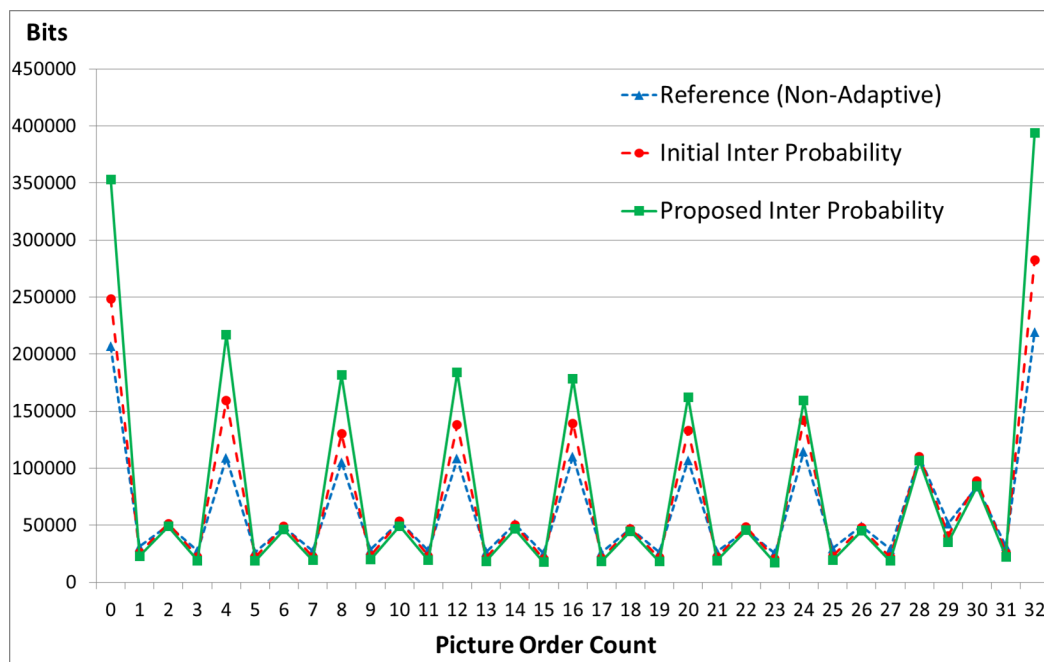
BD-BR PSNR	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-7.96%	-7.90%	-10.13%	-9.92%
	Class B	-6.90%	-6.93%	-7.36%	-8.48%
	Class C	-13.97%	-13.69%	-15.08%	-15.12%
	Class D	-11.42%	-10.91%	-12.24%	-11.77%
	Class E	-11.63%	-10.89%	-15.96%	-14.28%
	<b>Average</b>	<b>-10.38%</b>	<b>-10.08%</b>	<b>-11.90%</b>	<b>-11.81%</b>
	<b>Best</b>	<b>-19.00%</b>	<b>-19.01%</b>	<b>-22.38%</b>	<b>-22.08%</b>
	<b>Worst</b>	<b>+0.24%</b>	<b>-1.59%</b>	<b>+1.48%</b>	<b>-3.23%</b>
BD-BR SSIM	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-12.13%	-11.55%	-20.74%	-19.48%
	Class B	-8.00%	-8.01%	-14.08%	-15.51%
	Class C	-18.33%	-18.04%	-24.49%	-24.60%
	Class D	-16.97%	-15.91%	-24.22%	-22.54%
	Class E	-6.15%	-4.71%	-15.96%	-13.99%
	<b>Average</b>	<b>-12.44%</b>	<b>-11.84%</b>	<b>-19.70%</b>	<b>-19.28%</b>
	<b>Best</b>	<b>-24.39%</b>	<b>-24.22%</b>	<b>-32.28%</b>	<b>-31.42%</b>
	<b>Worst</b>	<b>+3.23%</b>	<b>+0.85%</b>	<b>-2.08%</b>	<b>-9.33%</b>
TBD	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	28.04%	7.16%	50.96%	11.75%
	Class B	37.17%	10.63%	80.63%	22.19%
	Class C	23.30%	4.72%	47.71%	8.08%
	Class D	31.09%	5.32%	60.65%	9.82%
	Class E	71.03%	25.82%	138.29%	35.90%
	<b>Average</b>	<b>37.37%</b>	<b>10.28%</b>	<b>75.19%</b>	<b>17.43%</b>

model stands whatever the codec implementation or with and without considering the  $\Psi$  scaling factor.

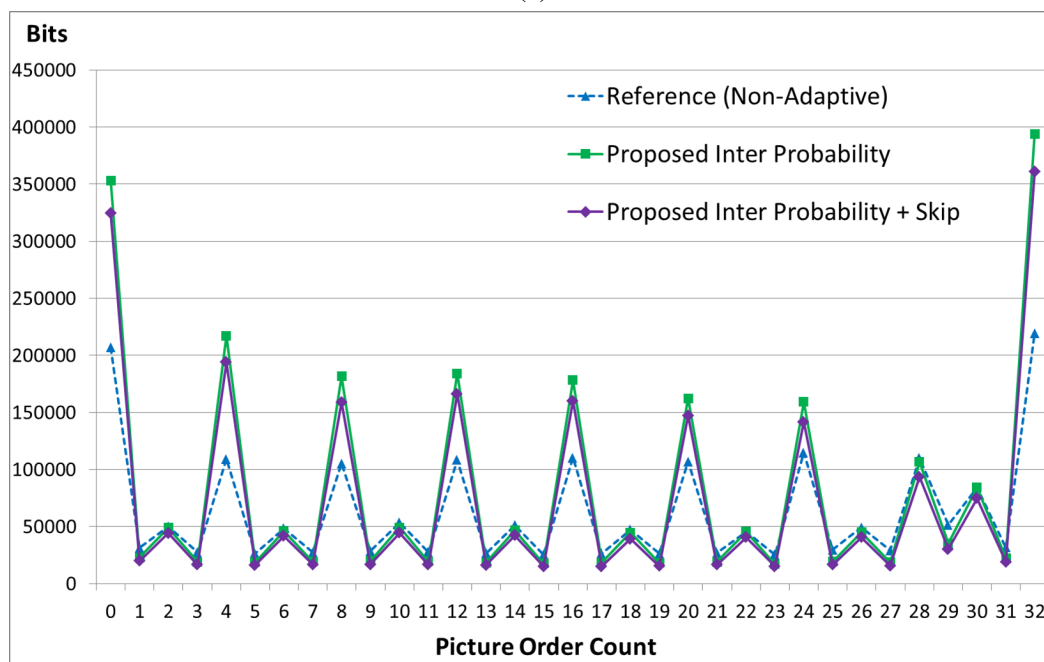
### Local Rate Distribution

In this section, we discuss the distribution of rates for several frames, typically a whole GOP, when some of the models presented above are enabled. The sequence *RaceHorses* with resolution of 832x480 is used for experiments in this section.

As expected, we observe in Fig. 4.7 (a) that the RDTQ model allocates more rate on the reference frames and lower temporal layer, while it decreases the rate allocated to frames in the highest temporal layers. The Proposed probability, propagating more *weight* on reference frames, tends to stretch even more the bitrate distribution across temporal layers.



(a)



(b)

Fig. 4.7 Rate distribution of first GOP frames with sequence *RaceHorses* at  $QP = 32$  for (a) RDTQ with Initial Probability and RDTQ with Proposed Probability; for (b) Proposed Probability with and without Skip consideration.

Table 4.6 Coding efficiency over no local quantization in x265 for RDSTQ.

BD-BR PSNR	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-5.72%	-5.46%	-4.42%	-4.39%
	Class B	-5.76%	-6.06%	-3.52%	-5.37%
	Class C	-12.02%	-11.65%	-11.61%	-11.55%
	Class D	-9.46%	-9.02%	-8.20%	-7.94%
	Class E	-12.05%	-11.13%	-12.20%	-10.59%
	<b>Average</b>	<b>-9.02%</b>	<b>-8.74%</b>	<b>-7.90%</b>	<b>-8.08%</b>
	<b>Best</b>	<b>-18.34%</b>	<b>-18.08%</b>	<b>-19.16%</b>	<b>-18.72%</b>
	<b>Worst</b>	<b>-0.35%</b>	<b>-0.62%</b>	<b>+1.80%</b>	<b>+1.39%</b>
BD-BR SSIM	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	-20.00%	-19.34%	-25.55%	-23.87%
	Class B	-16.79%	-16.38%	-20.43%	-20.96%
	Class C	-24.12%	-23.27%	-28.71%	-27.88%
	Class D	-25.49%	-23.82%	-30.39%	-27.57%
	Class E	-12.94%	-11.58%	-19.20%	-16.43%
	<b>Average</b>	<b>-20.07%</b>	<b>-19.09%</b>	<b>-24.85%</b>	<b>-23.53%</b>
	<b>Best</b>	<b>-30.87%</b>	<b>-30.11%</b>	<b>-39.01%</b>	<b>-36.60%</b>
	<b>Worst</b>	<b>-10.58%</b>	<b>-8.62%</b>	<b>-14.18%</b>	<b>-12.11%</b>
TBD	Probability Model	Initial	Initial + skip	Proposed	Proposed + skip
	Class A (8bits)	12.41%	5.13%	33.09%	7.55%
	Class B	14.90%	8.13%	53.24%	14.89%
	Class C	10.37%	8.87%	34.20%	10.89%
	Class D	21.85%	8.57%	51.47%	9.12%
	Class E	14.89%	5.51%	69.26%	8.09%
	<b>Average</b>	<b>15.16%</b>	<b>7.62%</b>	<b>49.05%</b>	<b>10.77%</b>

When considering the Skip probability, rates are equally decreased for each type of frame as observed in Fig. 4.7 (b), but it does not alter the delta rates between frames. This behavior is expected since taking into account the Skip mode consideration aims to limit the overspent rate on the entire GOP.

### $\Psi$ function and QP spatial distribution

In this section, more insights are given about the impact of the  $\Psi$  function on QPs spatial distribution. We observe the distribution of quantizers over an entire frame when the  $\Psi$  function is enabled. As earlier introduced, the psycho-visual factor chosen here is based on local spatial pixel variance, and is dedicated to optimize the SSIM score. It has the property to consider spatial masking effect, i.e. the fact that human eyes are less sensible to distortion made on high textured area (high local variance) than on area of low spatial complexity (low

local variance). Spatial masking significantly impacts compression artifact perception, as further analyzed by *Rimac-Drlje et al.* in [RDaM09].

The distribution of quantizers with and without psycho-visual function is shown on Fig. 4.8 for the frame 128 of *BQTerrace* sequence, with a target  $QP = 22$ . The darker blocks have the lowest quantizer (high rate) and the brighter ones have the highest quantizer (low rate). We point out that for this particular sequence encoded at  $QP = 22$ , almost no block is coded in Skip mode. Hence, we keep apart the influence of the Skip estimation model in this analysis.

We observe on Fig. 4.8 (a) that if no psycho-visual function is considered, the terrace is affected with high quality while the water and the roof are quantized more aggressively. The Inter probability is based on the relative difference between Intra mode and Inter mode estimated complexities. The more the estimated Intra complexity is relatively high compared to the Inter one, the more importance is put on reference frames. Given that, *BQTerrace* is highly uniform in terms of temporal complexity but not in terms of spatial complexity, it explains why more quality is affected to the most spatially complex areas, such as the terrace in this case.

However, the more textured is a block, the less small amounts of distortion are visible by the human eye. When the psycho-visual function is enabled (Fig. 4.8 (b)), we observe a better balanced distribution of the quantizers over the frame. Less rate is overspent on the terrace, while the water is subject to a quality improvement, in compliance with the spatial masking effect.

In our experiments we focus on the spatial masking effects based on the local pixel variance, that correlates well the *SSIM* quality metric. However, we point out that *RDSTQ* may be used to optimize any other perceptual criterion based on the selection of a  $\Psi$  factor that scales well the *MSE* to this criteria. For the interested readers, *Winkler* [Win05] provides a good overview of possible vision model and perceptual metrics to consider.

## Comparison to state of the art

This section compares our method with some state of the art solutions. To be fair with other methods found in the literature, the coding scheme was changed for the 7-B hierarchical and  $QP$  values  $\in \{22, 27, 32, 37\}$ . Other coding parameters remain the same and the reference is the *HM* encoder without *AQ* algorithm.

Three methods were chosen for comparison. The first one is proposed by *Gao et al.* in [GZLY17] and designed for optimizing the *PSNR*. Two other methods designed for optimizing the *SSIM* are proposed by *Yeo et al.* in [YTT13b] and *Xiang et al.* in [XJY<sup>+</sup>17]. We refer to these methods with their reference numbers in the following table. The proposed

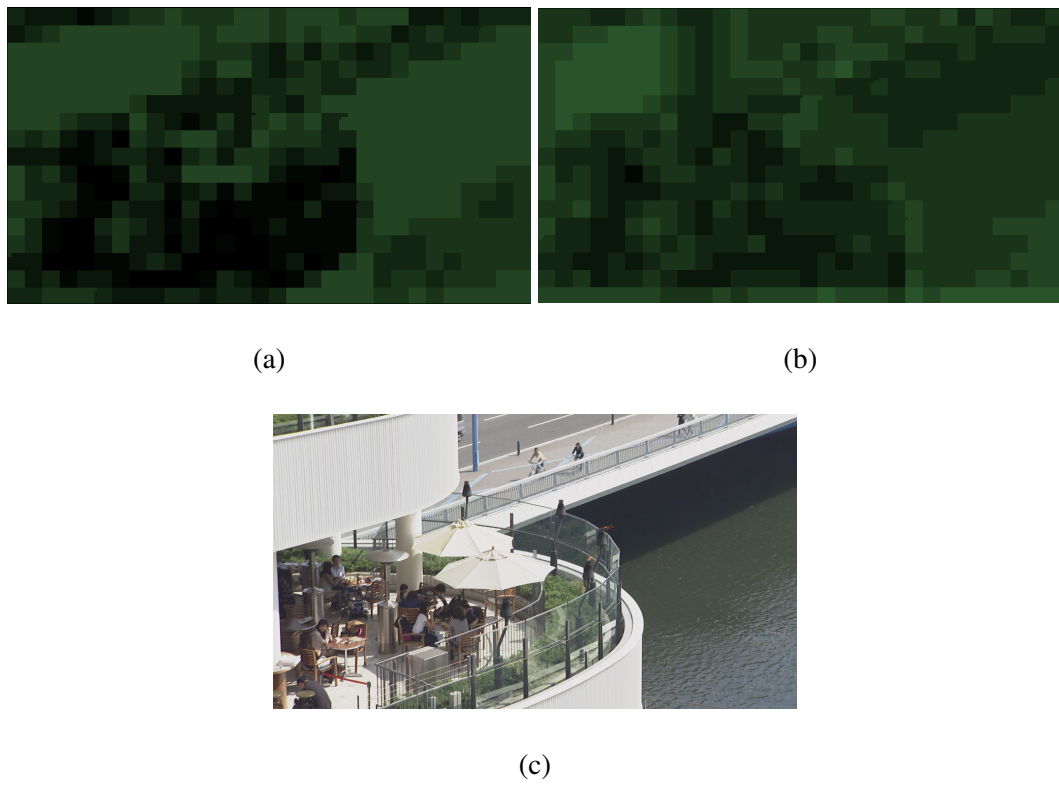


Fig. 4.8 QP distribution over the frame 128 of *BQTerrace* sequence (a) without psycho-visual function and (b) with psycho-visual enabled. (c) The source frame

solution denoted as *Ours* in results is the **RDSTQ** improved by the Proposed Inter and Skip probabilities. Simulation results are presented in Table. 4.7 for the *HM* encoder.

Table 4.7 Coding efficiency over no local quantization in *HM*.

Classes	BD-BR PSNR		BD-BR SSIM		
	[GZLY17]	Ours	[YTT13b]	[XJY <sup>+</sup> 17]	Ours
Class A	-4.25%	-12.53%	-5.78%	-5.42%	-27.98%
Class B	-4.10%	-9.35%	-4.18%	-3.12%	-20.82%
Class C	-5.60%	-17.56%	-3.90%	-5.13%	-29.78%
Class D	-4.10%	-15.82%	-4.47%	-4.53%	-30.86%
Class E	-8.40%	-25.09%	-2.86%	-0.25%	-27.37%
<b>Average</b>	<b>-5.18%</b>	<b>-15.59%</b>	<b>-4.14%</b>	<b>-3.69%</b>	<b>-26.93%</b>

It can be seen that our proposed solution substantially outperforms the **SDTP** optimized for **RA** coding configuration by  $-11.51\%$  in terms of **PSNR BD-BR** in average. The main reason is the simplified estimation of the dependencies made in [GZLY17] that extrapolates the dependency network instead of building it through a look-ahead as proposed. Consequently, *Gao et al.* solution saves some computational complexity by avoiding the use of a look-ahead but greatly limits the efficiency of the encoding optimization.

In terms of **SSIM**, the proposed solution outperforms the two methods by more than  $-22\%$  in average. However, an important drawback is that both methods consider rate constraints on a frame basis and not a **GOP** basis, which forbids any bit transfer between frames. These **AQ** methods are consequently more constrained than our proposal, even if *Xiang et al.* [XJY<sup>+</sup>17] implicitly try to consider the temporal dependencies through Inter mode **SATD** estimation. The large difference in coding efficiency confirms that **GOP** optimization is much more efficient than frame optimization. Moreover, we point out that even if **GOP** optimization requires a more complex look-ahead than frame optimization, such implementation are very acceptable in industrial applications.

### Encoding Complexity

We provide a rough comparisons of *HM* encoding runtime of **RDSTQ** with Proposed Inter probability and Skip probability for sequences in classes C, D and E in Table. 4.8. Different sequences are tested with  $QP \in \{22, 27, 32, 37, 42\}$ . We observe that encoding runtime is higher from  $53.28\%$  in average. This increase mostly comes from the look-ahead and is more noticeable for low bitrate. We point out that the look-ahead complexity comes from the data writing in a separated file, further read by the *HM* for **RDSTQ**. Thus, an embedded

look-ahead would not be as complex as the one we proposed. Moreover, in a multi-threaded implementation, such overhead would be neglected.

Table 4.8 Comparison of encoding runtime (in seconds) with no-AQ and RDSTQ in HM.

Sequences	No-AQ	RDSTQ		Runtime relative offset
	Encoding	Encoding	Lookahead	
RaceHorses_480	8092.67	7857.36	2081.604	22.81%
	6830.27	6546.99	2081.604	26.33%
	5846.35	5593.69	2081.604	31.28%
	5135.56	4849.16	2081.604	34.96%
	4498.13	4163.41	2081.604	38.84%
BasketballDrill	9725.36	9055.82	3839.134	32.59%
	8431.55	7784.01	3839.134	37.85%
	7418.90	6856.51	3839.134	44.17%
	6635.51	6085.56	3839.134	49.57%
	5973.31	5407.38	3839.134	54.80%
BlowingBubbles	2298.63	2079.30	1200.608	42.69%
	1837.58	1681.99	1200.608	56.87%
	1530.68	1403.72	1200.608	70.14%
	1313.54	1206.92	1200.608	83.29%
	1152.55	1071.54	1200.608	97.14%
BasketballPass	2456.23	2334.28	1396.372	51.89%
	2164.93	2048.26	1396.372	59.11%
	1918.05	1795.33	1396.372	66.40%
	1699.37	1577.95	1396.372	75.03%
	1510.91	1377.15	1396.372	83.57%
KristenAndSara	15247.84	14950.88	7617.788	48.01%
	13175.36	12584.20	7617.788	53.33%
	12380.74	11437.62	7617.788	53.91%
	11934.06	11155.10	7617.788	57.31%
	11605.41	10978.87	7617.788	60.24%
<b>Average</b>	-	-	-	<b>53.28%</b>

## 4.6 QuadTree Consideration for RDSTQ

We have proposed in this Chapter an efficient AQ method, named RDSTQ, that achieves global RDO over an entire GOP by taking into account the distortion propagation between frames. This method relies on accurate estimations of the Inter and Skip prediction modes probabilities and an analytical solution is driven from the proposed propagation model. A



potential limitation of the method comes from assuming an uniform partitioning within look-ahead process, i.e. a priori information on the sequence extracted based on an uniform picture partitioning. HEVC enables non-uniform adaptive partitioning using a QuadTree that is totally ignored in the model. In this Section, we discuss theoretical thoughts on how the QuadTree may be considered in the scope of the RDSTQ.

We first introduce the occurrence probability of a CU, that is the probability of a CU to be coded at a given CU depth. The introduction of occurrence probability allows to consider a non-uniform partitioning in the AQ method. Then we discuss the adaptation of the back propagation strategy of RDSTQ to compute the accumulation factor  $U_i$  of each possible CU (of variable size) at the  $i_t$  index. Finally, we provide a comparative analysis of this model with regards to the previously introduced RDSTQ in order to highlight its interest.

#### 4.6.1 Probability of occurrence for non-uniform partitioning

Hereafter, we discuss how we define the probability of occurrence  $s_{i_t,d}$  for a given CU  $i_t$  at depth  $d$  in the context of QuadTree partitioning. Considering samples belonging to the smallest CU size, i.e. samples at the highest QuadTree depth in a CTU, the occurrence probability of one CU at depth  $d$  is defined as the probability that the considered samples are coded with a CU of depth  $d$ . We note  $X_{i_t}$  the samples that belongs to CU  $i_t$  and  $\mathbb{P}(X_{i_t} \in d)$  the probability that these samples, or pixels, are coded in a CU of depth  $d$ . Thus, we define:

$$s_{i_t,d} = \mathbb{P}(X_{i_t} \in d) \quad (4.66)$$

The partitioning depends on successive split decisions. Thus,  $s_{i_t,d}$  can be expressed as (4.67), with  $\mathbb{P}(X_{i_t}, d, d+1)$  being the probability to split the CU at the designated depth  $d$  into four CUs of depth  $d+1$ .

$$s_{i_t,d} = (1 - \mathbb{P}(X_{i_t}, d, d+1)) \times \prod_{depth=0}^{d-1} \mathbb{P}(X_{i_t}, depth, depth+1) \quad (4.67)$$

In order to illustrate the occurrence probability, we choose to index the different CUs of a CTU using a Z-scan, as presented on Fig. 4.9. The index 1 refers to the 64x64 CU, the indexes 2,3,4,5 to the four 32x32 CUs, etc. The example is only given for one CTU. The maximum number of CUs within a CTU is equal to  $\sum_{d=0}^{d_{Max}} 4^d$ , with  $d_{Max}$  being the maximum depth.  $d_{Max} = 3$  is the maximum authorized by the HEVC standard, that results into 85 CUs associated to one CTU. We note  $N'$  the number of possible CUs in a frame. As an example  $N' = 85 \times N_{CTU_s}$  if  $d_{Max} = 3$  and the frame width and height are divisible by the CTU size.

Based on the Fig. 4.9, we propose an illustration of the concept of (4.67) in Fig. 4.10.

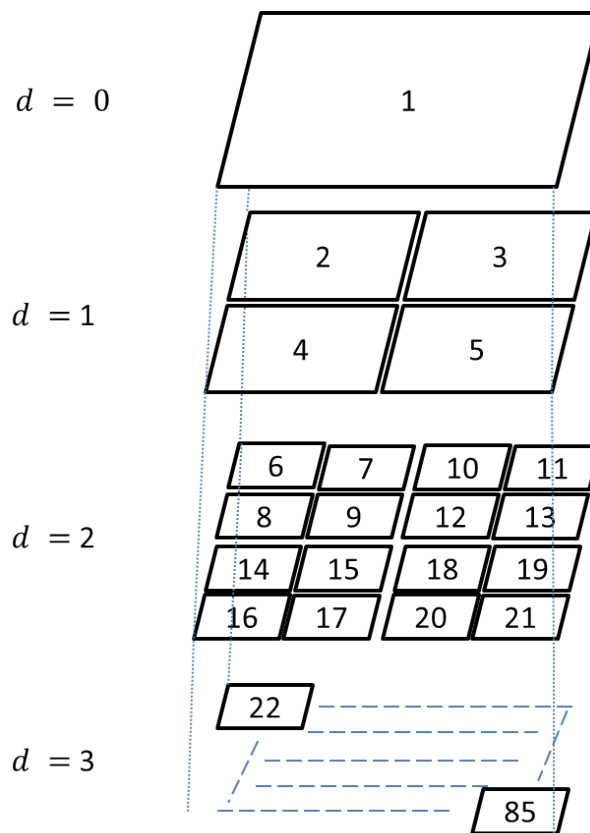


Fig. 4.9 Illustration of the chosen spatial CU indexing for the first CTU of a frame.

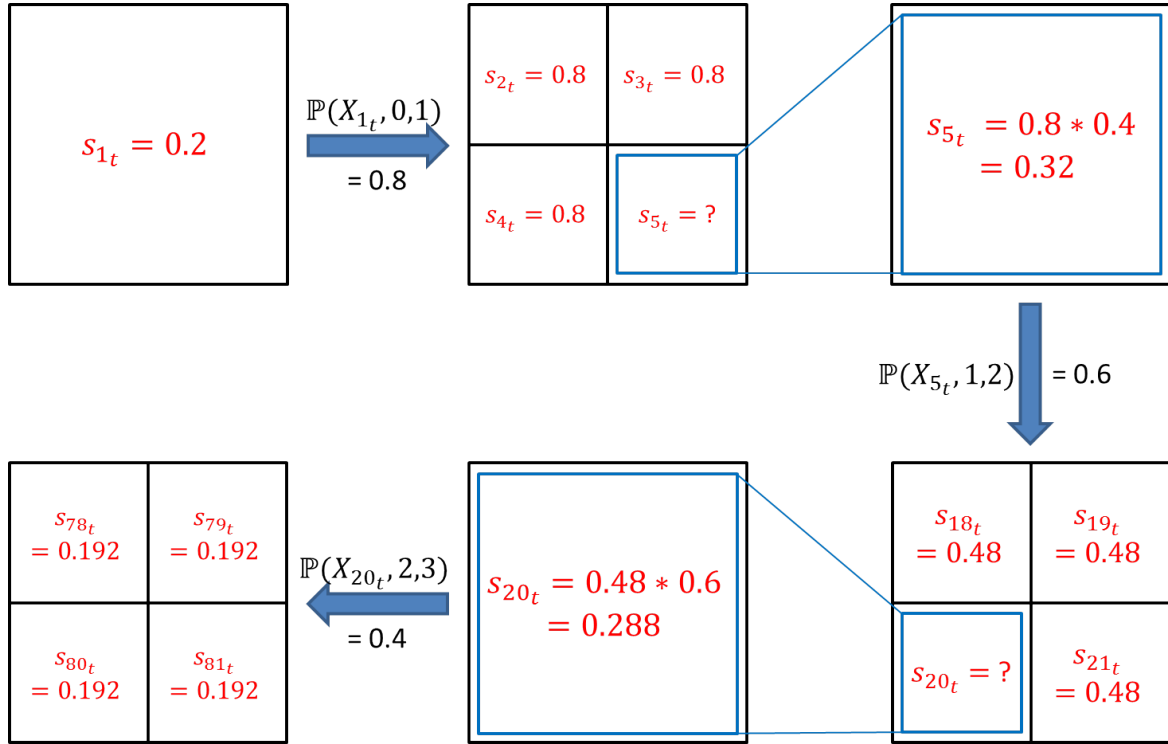


Fig. 4.10 Example of occurrence probability

We name  $CU_i$  a **CU** of index  $i$  in the frame  $t$  at the highest depth  $d_{Max}$ . The main interest of the proposed model is that, summing all occurrence probabilities of **CUs** that may contain  $CU_i$  equals to 1. The fact stated above is described by (4.68), with  $s_{i_t,d}$  being previously defined.

$$\sum_{d=0}^{d_{Max}} s_{i_t,d} = 1 \quad (4.68)$$

This property is necessary in the case of **RDSTQ**. As presented in Section 4.2, the back propagation strategy used in the method consists into estimating an importance factor for each **CU**. However, due to the motion compensation design, this factor may overlap on several blocks. Consequently, the property defined above ensures that the importance factor can be divided into all potential overlapped **CUs**, according to their occurrence probability.

In the following, we do not specify the depth of a **CU** in notations in order to keep mathematical formulations easy to read. Consequently, all possible **CUs** of a frame  $t$  are noted with the index  $i_t$  and their respective probability of occurrence  $s_{i_t}$ , since  $i_t$  necessarily refers to a unique depth.  $N'$  represents the total number of possible **CUs** in the frame and  $i \in \{1, \dots, N'\}$ .

## 4.6.2 Temporal distortion propagation model with non-uniform partitioning

According to the introduction of occurrence probability, the temporal distortion propagation model (4.69) is turned into (4.70).

$$D_{i_t} = c_{i_t} d_{i_t} + (1 - c_{i_t}) \sigma_{src_{i_t}}^2 + p_{i_t} \sum_{j_{t_{ref}} \in Ref(i_t)} r_{j_{t_{ref}}, i_t} D_{j_{t_{ref}}} \quad (4.69)$$

$$D_{i_t} = c_{i_t} d_{i_t} + (1 - c_{i_t}) \sigma_{src_{i_t}}^2 + p_{i_t} \sum_{j_{t_{ref}} \in Ref(i_t)} s_{j_{t_{ref}}} r_{j_{t_{ref}}, i_t} D_{j_{t_{ref}}} \quad (4.70)$$

We point out that the spatial index  $i$  corresponds to the index of one possible CU. Consequently, the scanning order corresponds to the Z-scan by depth and the total number of considered CUs in a frame is equal to  $N'$  as discussed before. The following notations are kept unchanged based on the initial proposed method:

- $D_{i_t}$  : the distortion of the CU  $i_t$ , i.e. the one observed between reconstructed and source samples.
- $d_{i_t}$  : the local distortion of the CU  $i_t$  that is only related to quantization error introduced on the CU  $i_t$ .
- $c_{i_t}$  : the coded probability of the CU  $i_t$ , the opposite of Skip mode probability.
- $p_{i_t}$  : the probability of a CU to use temporal prediction.
- $D_{j_{t_{ref}}}$  : the propagated distortion from a reference CU  $j_{t_{ref}}$  to the current CU  $i_t$ .

$r_{j_{t_{ref}}, i_t}$  is slightly adapted compared to the initial method.  $r_{j_{t_{ref}}, i_t}$  is the surface ratio of CU  $i_t$  that overlaps CU  $j_{t_{ref}}$ . However, in the context of non-uniform partitioning, these two CUs can be of different sizes. In such case, we need to scale the ratio depending on the CU sizes difference. Based on the QuadTree partitioning, the overlapping surface ratio is adapted as in (4.71), with  $depth(i_t)$  function returning the depth of the CU  $i_t$ .

$$r_{j_{t_{ref}}, i_t} = r_{j_{t_{ref}}, i_t} * 4^{depth_{j_{t_{ref}}} - depth(i_t)} \quad (4.71)$$

The reason for this scaling is that rates and distortions are averaged per sample/pixel. If a CU of index  $i_t$  is fully overlapping a larger reference CU  $j_{t_{ref}}$ , we must take into account that

the reference distortion  $D_{j_{ref}}$  is partially propagated to the distortion  $D_{i_t}$ . The expression of the total GOP distortion  $D_{Tot}$  is expressed in (4.72).

$$D_{Tot} = \sum_{t=1}^T \sum_{i=1}^{N'} s'_{i_t} \left( c_{i_t} d_{i_t} + (1 - c_{i_t}) \sigma_{src_{i_t}}^2 \right) U_{i_t}, \quad (4.72)$$

$s'_{i_t}$  represents the occurrence probability multiplied by a second scaling factor. Indeed, distortions and rates expressed in the RDSTQ model are averaged per sample, i.e. per pixel. Consequently, summing average distortions from blocks of different sizes requires some scaling. Thus  $s_{i_t}$  is turned into  $s'_{i_t}$  according to (4.73), with  $depth_{Max}$  the maximum CU depth and  $depth_{i_t}$  the depth of the CU of index  $i_t$ .

$$s'_{i_t} = s_{i_t} \times 4^{depth_{Max} - depth_{i_t}} \quad (4.73)$$

The accumulation factor  $U_{i_t}$  from (4.28), in that context, is redefined in (4.74).

$$\begin{cases} U_{i_T} &= \Psi_{i_T} \\ U_{i_t} &= \Psi_{i_t} + \sum_{i_{t+1}} s_{i_{t+1}} p_{i_{t+1}} r_{i_t, i_{t+1}} U_{i_{t+1}} \end{cases} \quad (4.74)$$

For simplification matters, we assume  $s_{i_t}$  to be independent of  $\Delta_{i_t}$ . The  $D_{Tot}$  derivative according to  $\Delta_{i_t}$  the quantization step of the CU  $i_t$  is written in (4.75).

$$\frac{\partial D_{Tot}}{\partial \Delta_{i_t}} = \frac{\partial d_{i_t}}{\partial \Delta_{i_t}} s'_{i_t} c_{i_t} U_{i_t} \quad (4.75)$$

Ultimately, we anticipate the optimal quantizer of a CU  $k_\tau$  to be obtained with the formula (4.76). Developments to obtain (4.76) are provided in Appendix C.

$$dQP_{k_\tau} = -str \left( \log_2(U_{k_\tau}) - \frac{\sum_{t=1}^T \sum_{i=1}^{N'} s'_{i_t} c_{i_t} \log_2(U_{i_t})}{\sum_{t=1}^T \sum_{i=1}^{N'} s'_{i_t} c_{i_t}} \right) \quad (4.76)$$

### 4.6.3 Comparative analysis with initial model

The initial RDSTQ model suffers from some limitations due to the theoretical assumptions used to obtain the analytical solution. Even if an uniform partitioning is assumed during the look-ahead, the encoding procedure uses the QuadTree partitioning. In the following, we focus on a  $2N \times 2N$  CU indexed  $i_t$  at a depth  $d$ , and the four corresponding  $N \times N$  CUs indexed by  $\{i_t, x\}_{x=1}^4$  at a depth  $d+1$ . We then compare two cases: considering uniform or non-uniform partitioning in the look-ahead.

**Considering uniform partitioning:**

In the initial model, CUs with larger size than the CU size considered for the quantizers estimation, i.e. 16x16 in the look-ahead, were applied the average delta quantizers of their related sub-CUs. We define the constant offset  $Cte_{QP}$  from the rate constraint as (4.77).

$$Cte_{QP} = \frac{\sum_{t=1}^T \sum_{i=1}^{N'} c_{i_t} \log_2(U_{i_t})}{\sum_{t=1}^T \sum_{i=1}^{N'} c_{i_t}} \quad (4.77)$$

In the initial solution we had:

$$\begin{cases} U_{j_T} &= \Psi_{j_T} \\ U_{j_{t-1}} &= \sum_{i_t} p_{i_t} r_{j_{t-1}, i_t} U_{i_t} + \Psi_{j_{t-1}}. \end{cases} \quad (4.78)$$

$$\begin{aligned} dQP_{i_t} &= \frac{1}{4} \sum_{x=1}^4 dQP_{i_t, x} \\ &= -str \frac{1}{4} \sum_{x=1}^4 (\log_2(U_{i_t, x}) - Cte_{QP}) \\ &= str \left( Cte_{QP} - \frac{1}{4} \sum_{x=1}^4 \log_2(U_{i_t, x}) \right) \\ &= str \left( Cte_{QP} - \log_2 \left( \prod_{x=1}^4 U_{i_t, x}^{\frac{1}{4}} \right) \right) \end{aligned} \quad (4.79)$$

**Considering non-uniform partitioning:**

In the proposed consideration of QuadTree, optimal quantizer are obtained by (4.76), that is a direct mapping from the accumulation factor  $U$  and a constant term coming from the total rate constraint. Consequently,  $dQP_{i_t}$  depends on the accumulation factor  $U_{i_t}$ , and the four delta quantizers of the sub-CUs depends on  $\{U_{i_t, x}\}_{x=1}^4$ .

The accumulation factor is recursively computed, and depends on the look-ahead motion estimation and the psycho-visual factor. Moreover, we show that the accumulation factor estimated for a given CU does not depend on its occurrence probability. Consequently, if we ignore the psycho-visual factor, i.e. in the context of RDTQ, we can state the equality in (4.80).

$$U_{i_t} = \frac{1}{4} \sum_{x=1}^4 U_{i_t, x} \quad (4.80)$$

This equality supposes that the accumulation factor of a 2Nx2N CU is equal to the average of accumulation factors for its corresponding NxN CUs. It is intuitively correct since

the two parts of the equality refer to the same spatial surface. Consequently, the optimal quantizer estimation can be written as (4.81).

$$dQP_i = str \left( Cte'_{QP} - \log_2 \left( \frac{1}{4} \sum_{x=1}^4 U_{i,x} \right) \right) \quad (4.81)$$

We point out that  $Cte'_{QP}$  is slightly different from  $Cte_{QP}$  and is defined as (4.82). However, it remains a constant offset that is added to all quantizers.

$$Cte_{QP} = \frac{\sum_{t=1}^T \sum_{i=1}^{N'} s'_i c_i \log_2 (U_i)}{\sum_{t=1}^T \sum_{i=1}^{N'} s'_i c_i} \quad (4.82)$$

If we compare (4.79) and (4.81), we observe a mismatch between using the geometrical mean or the arithmetic mean of the sub-CUs accumulation factor when estimating a larger CU delta quantizer. The QuadTree consideration, as proposed in this Section, should resolve this issue by computing the correct accumulation factor for each CU, whatever its size/depth. We point out that, based on an uniform partitioning look-ahead, it is pointless to only replace the geometrical mean of accumulation factor by the arithmetic mean. The reason is that the term  $Cte_{QP}$  resulting from the total rate constraint, is dependent of the occurrence probability as shown by (4.82).

At the time of this thesis writing, the QuadTree consideration in the RDSTQ framework is only based on theoretical thoughts. Further developments and experimentations are required in order to confirm the relevance and validity of this model.

## 4.7 Conclusion

In this Chapter, we demonstrate the benefits of considering temporal distortion propagation for adaptive quantization. We provide a new spatio-temporal algorithm to compute local quantizers, based on a theoretical framework able to describe the temporal distortion propagation from an R-D standpoint. In particular, we model the temporal distortion propagation making possible the (temporal) retro accumulations of any (spatial) psycho-visually weighted distortion onto reference images. Using the R(D) Shannon bound, its high bitrate approximation, and a Lagrange optimization, analytical solution is obtained and thoroughly demonstrated for delta quantizers. We also show through extensive experimentations the benefits of considering both Skip probability and accurate Inter probability estimators for AQ. It provides substantial bitrate savings whatever the HEVC codec implementations. Consider-

ing the **RDTQ**, i.e.  $\Psi_{i,t} = 1 \forall i, t$ , we report average **BD-BR** gains of  $-11.81\%$  and  $-16.51\%$  **PSNR**-based in the x265 and the **HM** encoders, respectively. We obtain these gains against no-**AQ** method and gains are systematic, i.e. all sequence codings are improved.

Thanks to the convenient consideration of a psycho-visual factor, the **RDSTQ** also allows to optimize more perceptually-oriented quality metrics, such as the **SSIM**. When using a psycho-visual factor based on the local pixel variance, that estimates the spatial masking, average **BD-BR** gains based on **SSIM** are then of  $-23.53\%$  and  $-26.26\%$  for the x265 and the **HM**, respectively. Careful comparison against state-of-the-art similar approaches is also reported. **RDSTQ** model outperforms previous techniques with  $-10.41\%$  **PSNR**-based and  $-22.79\%$  **SSIM**-based average bitrate savings, when reference is without **AQ**. The main conclusion coming out from these experiments is the higher efficiency of the **GOP** optimization compared to the frame optimization; **GOP** optimization being closer to the global optimization bound.

We point-out that most **AQ** methods, including ours, introduces a **TBD** when locally adapting the quantizers, compared to no **AQ** reference. We prove that the Skip probability consideration helps to reduce the average **TBD** of the **RDSTQ** with Initial probability from  $12.4\%$  to  $4.33\%$  in the **HM** and from  $15.16\%$  to  $7.62\%$  in x265. Finally, we demonstrate that computed delta quantizers based on the proposed model are bounded. Their output dynamic range is controllable, preventing from any worst case scenario.

However, the proposed method has some limitations: the model is simplified by assuming all blocks to be of the same size, notably during the look-ahead analysis. In **HEVC**, the QuadTree partitioning is known to be a key tool in terms of coding efficiency. We proposed in Section 4.6 an approach for considering the QuadTree partitioning in the temporal distortion propagation model. Thanks to the definition of an occurrence probabilities, some developments allow us to anticipate a simple optimal quantizer estimation model. This last proposal is based on unverified assumptions and requires accurate split decision probabilities. However, we believe that predicting the QuadTree based on non-uniform partitioning into the look-ahead should improve the model coding efficiency. The distortion propagation model should be more accurate and the quantizer computation would result in substantial coding gains. Consequently, future work will address this topic by validating the proposed model.





# Conclusion

In this thesis, we overview the dependencies related to an hybrid video coding scheme and investigate the solutions to consider them into the global **RDO** problem. After introducing some fundamentals about source coding in Chapter 1, we highlight in Section 1.4 that any source coding system including predictive coding and quantization is subject to coding decision dependencies.

Moreover, we prove in Section 1.5 that considering dependencies through exhaustive search of optimal decisions becomes rapidly intractable, in terms of computational complexity. The brief description of the **HEVC** standard that is proposed in Chapter 2 ends up with identification of multiple coding dependencies related to the coding scheme and listed hereafter:

- The distortion made on a coding unit may be propagated to other units because of the prediction mechanisms. Global distortion minimization must take it into account.
- Side informations are coded using differential/predictive coding and arithmetic coding that depends on other side informations values. Hence, regularizing side informations occurrence reduces the overall side information cost.
- Some coding processes are applied sequentially, such as the quantization after the transform. Joint optimization of the processes should lead to better performance.

After identifying the different dependencies related to the **HEVC** coding scheme, or more generally any modern hybrid video coding scheme, we focused our studies on solutions to improve the global **R-D** efficiency of an encoder within the scope of a standard.

In Chapter 3, an overview of the methods dealing with dependencies in Intra coding is proposed. These methods focus on the distortion that is spatially propagated from one block to another. One category of methods address the source problem, i.e. try to minimize the distortion on the block boundaries through normative or non-normative solutions. Other methods rather try to model the dependencies based on observations or mathematically in order to efficiently modify the **RDO**. However, none of these methods considers all Intra

coding-related dependencies at once, nor is able to provide a boundary of the maximum achievable coding efficiency. In response to this, the framework we proposed in Section 3.2 aims to

- Estimate the impact of dependencies through exhaustive minimization of a joint cost, i.e. minimizing the total cost of multiple coding units that are assumed dependent on each other.
- Evaluate the upper-bound of the rate-quality curve and the potential interest of developing JRDO models for the Intra-only coding context.

We start by formalizing the dependencies between PUs that are processed sequentially and have strong spatial dependencies. An exhaustive search for the joint minimum R-D cost is then processed in order to optimize group of spatial predictors of 2 or 4 PUs. The opportunities for reducing the computational complexity of the method are also studied in order to propose a more acceptable complexity with systematic coding gains. Finally, the exhaustive joint optimization is used for estimating QP offsets in an AQ context. The overall conclusion of this chapter is that joint optimization should be extended to a large number of coding decisions, instead of 2 or 4 as proposed, in order to achieve significant improvements. However, the complexity limits the possibility of an exhaustive verification of this statement. We also notice that adaptive quantization is an efficient tool to optimize video coding, but the bitrate overhead of the method in Intra coding annihilate its gains.

In Chapter 4 we focus on the dependencies consideration in the context of Inter coding. We assume the opportunities for encoding optimization to be more interesting than the one observed in the Intra coding case. Due to the extensive use of motion compensation, a dependency network connecting all blocks of the same GOP can be observed. The solution proposed in this Chapter models this dependency network through a temporal distortion propagation model and an efficient use of Inter and Skip modes probabilities. Optimal quantizers are then designed per coding unit in order to achieve the global optimization in terms of Rate-Distortion efficiency. By implementing the algorithm into the HEVC reference Model (HM), we prove it outperforms several related methods from state-of-the-art. Moreover, along with the demonstration of optimal quantizer solution, we propose an in-depth analysis of the algorithm behavior. This analysis includes, among others, the relative distribution of rates between frames and the output quantizers range control. Finally, we proposed in Section 4.6 an approach to improve the model by considering non-uniform partitioning.

To conclude this thesis, all proposed solutions focused on different dependencies affecting the encoding optimization in hybrid video coding scheme. Using non-normative solution, i.e.

compliant with a given standard, we successfully improve the overall coding efficiency of an encoder. In Chapter 3, we first estimate the opportunities in Intra-only coding for improving the optimization of coding parameters through joint exhaustive optimization. In Chapter 4, we use a simplified model of temporal distortion propagation. This model further allow to solve adaptive quantization optimization problem, from a global RDO standpoint.

## Perspectives

In the Chapter 3, we aimed to estimate an upper bound of the achievable coding efficiency when considering the dependencies. We reported coding improvements using exhaustive methods to jointly optimize different parameters (spatial predictor, quantization parameter) on multiple coding units (2 or 4), but the computational complexity was an intractable limitation. However, several state-of-the-art methods described in Section 3.1 present significant gains. We consequently believe, that an efficient modeling of the spatial distortion propagation should bring higher gains than the one we reported. For example, the RDSTQ model proposed in Chapter 4 should be efficient in the Intra-only case, as long as we are able to model the distortion propagation at pixel level. However, the desired upper bound does not seem estimable.

The RDSTQ model that is extensively described in Chapter 4 is proven to be highly efficient compared to state-of-the-art methods. However, some weaknesses should be taken into account:

- Necessary mathematical simplifications are made by ignoring the influence of quantizer on Inter probability and partially ignore it on the Skip probability, i.e. the derivative is assumed to be zero.
- The lookahead and related retro-propagation in the RDSTQ only consider a uniform partitioning of 16x16 blocks. However, HEVC allow a non-uniform partitioning using a QuadTree.

Ignoring the effect of quantizer on the estimated probabilities allows a simple analytical solution. Without simplifications, the consequence would be that local quantizers impacts the retro-propagation itself, resulting in a chicken and egg dilemma. However, it could improve the overall coding efficiency. As for the non-uniform partitioning, we believe it should also improve the coding efficiency. The QuadTree consideration have been explored in Section 4.6, but is not fully implemented nor validated at the moment.

Other dependencies have been identified but not necessarily treated during this thesis. An example would be the dependency related to motion vectors signalization. As discussed in

Section 2.8, heterogeneous motion vector fields would result into a larger signalization cost. Consequently, we believe that properly regularizing the motion vector field should improve the coding efficiency.

# Author Publications

## Articles

### Conference Paper

**Maxime Bichon**, Julien Le Tanou, Michael Ropert, Wassim Hamidouche, Luce Morin and Lu Zhang. "Temporal Adaptive Quantization using accurate estimations of Inter and Skip probabilities". In 2018 Picture Coding Symposium (PCS), June 2018.

**Maxime Bichon**, Julien Le Tanou, Michael Ropert, Wassim Hamidouche, Luce Morin and Lu Zhang. "Low Complexity Joint RDO of Prediction Units Couples for HEVC Intra Coding". In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), April 2018.

Michael Ropert, Julien Le Tanou, **Maxime Bichon**, Mederic Blestel. "R-D Spatio-Temporal Adaptive Quantization based on Temporal Distortion Backpropagation in HEVC". In 2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP), October 2017.

**Maxime Bichon**, Julien Le Tanou, Michael Ropert, Wassim Hamidouche, Luce Morin and Lu Zhang. "Inter-Block Dependencies Consideration for Intra Coding in H.264/AVC and HEVC Standards". In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2017.

### Journal Paper

**Maxime Bichon**, Julien Le Tanou, Michael Ropert, Wassim Hamidouche and Luce Morin. "Optimal Adaptive Quantization based on Temporal Distortion Propagation model for HEVC". Manuscript submitted for publication.



# Appendix A

## Derivation of the Rate-Quantization function

### A.1 Derivation of (1.23)

We start from the differential entropy  $H(X_i)$  of the random variable  $X_i$  is equal to (1.23).

$$H(X_i) = - \int f_{X_i}(x) \log_2(f_{X_i}(x)) dx \quad (\text{A.1})$$

The quantization process divides the probability density function into intervals  $\Delta_i = [x_i; x_{i+1}]$ . The quantization is considered scalar uniform and all intervals length are equals to  $\Delta = x_{i+1} - x_i$ . (A.1) can thus be turned into (A.2).

$$H(X_i) = - \sum_i \int_{\Delta_i} f_{X_i}(x) \log_2(f_{X_i}(x)) dx \quad (\text{A.2})$$

By assuming the distribution to be uniform in the range  $[x_i; x_{i+1}]$ , e.g. if  $\Delta$  tends to be small, we can express the distribution integral as a discrete variable  $p_i$  defined in (A.3).

$$p_i = \int_{\Delta_i} f_{X_i}(x) dx \quad (\text{A.3})$$

For  $f_{X_i}(x)$  stable enough, it can be approximated as  $f_{X_i}(x) = \frac{p_i}{\Delta}, \forall x$ , that further leads into (A.4).

$$\sum_i p_i = \sum_i \int_{\Delta_i} f_{X_i}(x) dx = \int f_{X_i}(x) dx = 1 \quad (\text{A.4})$$



Then, the rate can be expressed as a function of the entropy and  $\Delta$  as proven by the following developments:

$$H(X_i) = -\sum_i \int_{\Delta_i} f_{X_i}(x) \log_2(f_{X_i}(x)) dx \quad (\text{A.5})$$

$$H(X_i) = -\sum_i \int_{\Delta_i} \frac{p_i}{\Delta} \log_2\left(\frac{p_i}{\Delta}\right) dx \quad (\text{A.6})$$

$$H(X_i) = -\sum_i \frac{p_i}{\Delta} \log_2\left(\frac{p_i}{\Delta}\right) \underbrace{\int_{\Delta_i} dx}_{=\Delta} \quad (\text{A.7})$$

$$H(X_i) = -\sum_i p_i \log_2\left(\frac{p_i}{\Delta}\right) \quad (\text{A.8})$$

$$H(X_i) = -\sum_i p_i \log_2(p_i) + \sum_i p_i \log_2(\Delta) \quad (\text{A.9})$$

$$H(X_i) = -\underbrace{\sum_i p_i \log_2(p_i)}_{=H(Q)} + \log_2(\Delta) \underbrace{\sum_i p_i}_{=1} \quad (\text{A.10})$$

We consider the average rate  $R(\Delta)$  to be well approximated by the entropy of the reconstructed signal  $H(Q)$ . Consequently the rate is obtained in (A.11).

$$R(\Delta) \approx H(X_i) - \log_2(\Delta) \quad (\text{A.11})$$

# Appendix B

## RDSTQ equations

### B.1 Computing $D_{Tot}$

Notation are simplified by defining  $d'_{i_t} = c_{i_t}d_{i_t} + (1 - c_{i_t})\sigma_{src_{i_t}}^2$ . We start from the distortion defines with the temporal distortion propagation model as below,

$$D_{i_t} = d'_{i_t} + p_{i_t} \sum_{j_{t_{ref}} \in Ref(i_t)} r_{j_{t_{ref}}, i_t} D_{j_{t_{ref}}}. \quad (B.1)$$

We refine the overlapping ratio  $r_{j_{t_{ref}}, i_t}$  as follow:

$$r_{j_{t_{ref}}, i_t} = \begin{cases} 0 & \text{if } j_{t_{ref}} \notin Ref(i_t) \\ r\_j\_t\_ref, i\_t & \text{if } j_{t_{ref}} \in Ref(i_t) \end{cases} \quad (B.2)$$

To avoid dealing with the image number, and without losing in generality, the adopted notation will simply be:

$$D_{i_t} = d'_{i_t} + p_{i_t} \sum_{j_{t-1}} r_{j_{t-1}, i_t} D_{j_{t-1}} \quad (B.3)$$

Then we can write the following:

$$D_{i_1} = d'_{i_1} \quad (B.4)$$

$$D_{i_2} = d'_{i_2} + p_{i_2} \sum_{i_1} r_{i_1, i_2} D_{i_1} \quad (B.5)$$

$$D_{i_2} = d'_{i_2} + p_{i_2} \sum_{i_1} r_{i_1, i_2} d'_{i_1} \quad (B.6)$$

$$D_{i_3} = d'_{i_3} + p_{i_3} \sum_{i_2} r_{i_2, i_3} D_{i_2} \quad (\text{B.7})$$

$$D_{i_3} = d'_{i_3} + p_{i_3} \sum_{i_2} r_{i_2, i_3} \left( d'_{i_2} + p_{i_2} \sum_{i_1} r_{i_1, i_2} d'_{i_1} \right) \quad (\text{B.8})$$

The distortion on the CU  $i_{\tau-1}$  with  $\tau > 1$  is expressed as

$$D_{i_{\tau-1}} = p_{i_{\tau-1}} \sum_{i_{\tau-2}} r_{i_{\tau-2}, i_{\tau-1}} \left( p_{i_{\tau-2}} \sum_{i_{\tau-3}} r_{i_{\tau-3}, i_{\tau-2}} \left( \dots p_{i_2} \sum_{i_1} r_{i_1, i_2} d'_{i_1} + d'_{i_2} \right) + \dots \right) + d'_{i_{\tau-2}} + d'_{i_{\tau-1}}, \quad (\text{B.9})$$

and the total distortion  $D_{Tot}$  is expressed as:

$$\begin{aligned} D_{Tot} &= \sum_{t=1}^T \sum_{i=1}^N d'_i \Psi_i \\ &= \sum_{t=1}^T \left( \sum_{i=1}^N \Psi_i \left( p_{i_{t-1}} \sum_{i_{t-2}} r_{i_{t-2}, i_{t-1}} \left( p_{i_{t-2}} \sum_{i_{t-3}} r_{i_{t-3}, i_{t-2}} \left( \dots p_{i_2} \sum_{i_1} r_{i_1, i_2} d'_{i_1} + d'_{i_2} \right) + \dots \right) + d'_{i_{t-2}} \right) + d'_{i_{t-1}} \right) \right) \end{aligned} \quad (\text{B.10})$$

$D_{Tot}$  can be written as a linear combination of  $U_i$  and  $d'_i$ , then

$$D_{Tot} = \sum_{t=1}^T \sum_{i=1}^N d'_i U_i. \quad (\text{B.11})$$

After calculation and rearranging we obtain:

$$\begin{aligned} U_{n_\tau} &= \frac{\partial D_{Tot}}{\partial d_{n_\tau}} \\ &= \left[ \Psi_{n_\tau} + \sum_{t=\tau+1}^T \left( \sum_{i_t} \sum_{i_{t-1}} \dots \sum_{i_{\tau+1}} \Psi_{i_t} p_{i_t} r_{i_{t-1}, i_t} \Psi_{i_{t-1}} p_{i_{t-1}} r_{i_{t-2}, i_{t-1}} \dots p_{i_{\tau+1}} r_{n_\tau, i_{\tau+1}} \right) \right] \end{aligned} \quad (\text{B.12})$$

## B.2 Accumulation factor in recursive form

Let have

$$\begin{cases} U_{j_T} &= \Psi_{j_T} \\ U_{j_{t-1}} &= \sum_{i_t} p_{i_t} r_{j_{t-1}, i_t} U_{i_t} + \Psi_{j_{t-1}}. \end{cases} \quad (\text{B.13})$$

We can confirm the recursive form as follow

$$U_{j_{T-1}} = \sum_{i_T} p_{i_T} r_{j_{T-1}, i_T} \Psi_{i_T} + \Psi_{j_{T-1}} \quad (\text{B.14})$$

$$U_{j_{T-2}} = \sum_{i_{T-1}} \sum_{i_T} p_{i_{T-1}} r_{j_{T-2}, i_{T-1}} p_{i_T} r_{j_{T-1}, i_T} \Psi_{i_T} + \sum_{i_{T-1}} p_{i_{T-1}} r_{j_{T-2}, i_{T-1}} \Psi_{i_{T-1}} + \Psi_{j_{T-2}} \quad (\text{B.15})$$

$$U_{j_1} = \Psi_{j_1} + \sum_{\tau=1}^T \sum_{i_\tau} \sum_{i_{\tau-1}} \dots \sum_{i_1} p_{i_\tau} r_{i_{\tau-1}, i_\tau} p_{i_{\tau-1}} r_{i_{\tau-2}, i_{\tau-1}} \dots p_{i_1} r_{j_1, i_2} \Psi_{i_\tau} \quad (\text{B.16})$$

### B.3 Computing the Lagrangian

$$\frac{\partial J_{Tot}}{\partial Q_{i_t}} = \frac{\partial d_{i_t}}{\partial Q_{i_t}} c_{i_t} U_{i_t} + \lambda \frac{\partial R_{i_t}}{\partial Q_{i_t}} c_{i_t} = 0 \quad (\text{B.17})$$

The minimization of  $J_{Tot}$  is independent of  $c_{i_t}$  which is removed from equations. Then to isolate the  $\lambda$  we derivate  $J_{Tot}$  by  $R_{i_t}$  results in

$$\frac{\partial J_{Tot}}{\partial R_{i_t}} = \frac{\partial d_{i_t}}{\partial Q_{i_t}} \frac{\partial Q_{i_t}}{\partial R_{i_t}} U_{i_t} + \lambda \quad (\text{B.18})$$

$$\frac{\partial J_{Tot}}{\partial R_{i_t}} = 0 \Leftrightarrow \lambda = -\frac{\partial d_{i_t}}{\partial Q_{i_t}} \frac{\partial Q_{i_t}}{\partial R_{i_t}} U_{i_t} \quad (\text{B.19})$$

$$\lambda = -U_{i_t} \frac{\frac{\partial d_{i_t}}{\partial Q_{i_t}}}{\frac{\partial R_{i_t}}{\partial Q_{i_t}}} = -U_{i_t} \frac{\partial d_{i_t}}{\partial R_{i_t}} = -U_{i_t} \frac{\partial D_{i_t}}{\partial R_{i_t}} \quad (\text{B.20})$$

By using the R-D Shannon bound  $R_{i_t} = -\frac{1}{2} \log_2 \left( \frac{D_{i_t}}{\alpha \sigma_{i_t}^2} \right)$ , we obtain

$$\frac{\partial R_{i_t}}{\partial D_{i_t}} = \frac{1}{2 \cdot \ln(2) \cdot D_{i_t}}. \quad (\text{B.21})$$

Finally, the optimal  $\lambda$  is defined by

$$\lambda = 2 \cdot \ln(2) \cdot U_{i_t} \cdot D_{i_t}. \quad (\text{B.22})$$



# Appendix C

## QuadTree consideration in RDSTQ

### C.1 Derivation of optimal quantizer considering QuadTree partitioning

Hereafter, we verify that the  $\lambda$  value used for total R-D cost  $J_{Tot}$  minimization is independent of the probability of existence and scale factor. We start from the total R-D cost partial derivative according to quantization step  $\Delta_{i_t}$  to be equal to zero as:

$$\frac{\partial J_{Tot}}{\partial \Delta_{i_t}} = \frac{\partial D_{Tot}}{\partial \Delta_{i_t}} + \lambda \frac{\partial}{\partial \Delta_{i_t}} \sum_{t=1}^T \sum_{i=1}^N R_{i_t} = 0 \quad (C.1)$$

We express the rate  $R_{i_t}$  of a CU  $i_t$  depends of the probability of existence of the CU. Since  $R_{i_t}$  is an average rate per samples, it also requires to be scaled, as explained for the distortion Section 4.6.2. Thus, by  $s'_{i_t}$  s a function of  $R_{i_t}^C$  and  $R_{i_t}^S$  as the rates of a CU  $i_t$  to be coded in Inter/Intra and Skip, respectively. Thus,  $J_{Tot}$  derivative is turned into (C.2).

$$\frac{\partial J_{Tot}}{\partial \Delta_{i_t}} = \frac{\partial d_{i_t}}{\partial \Delta_{i_t}} s'_{i_t} c_{i_t} U_{i_t} + \lambda \frac{\partial R_{i_t}}{\partial \Delta_{i_t}} s'_{i_t} c_{i_t} = 0 \quad (C.2)$$

Developments to obtain the optimal  $\lambda$  value are the same than in Appendix B.3 and injecting the R-D Shannon bound we obtain the same result:

$$\lambda = 2 \ln(2) U_{i_t} D_{i_t} \quad (C.3)$$

The following equations are very similar to the initial RDSTQ analytical solution.

$$\log_2(\lambda') = \log_2(U_{i_t} D_{i_t}). \quad (C.4)$$

We exhibit the total rate in order to introduce the rate constraint, i.e. the average GOP bitrate  $R_{Tot}$  is desired to remain identical. By summing the log values weighted by the  $c_i$  and  $s'_i$  on both sides of (4.39) over all CUs of the GOP, we have

$$\log_2(\lambda') \underbrace{\sum_{t=1}^T \sum_{i=1}^N s'_i c_i}_{=N_{Tot}} = \sum_{t=1}^T \sum_{i=1}^N s'_i c_i \log_2(U_i D_i), \quad (\text{C.5})$$

$$\log_2(\lambda') = \frac{1}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N s'_i c_i \log_2(U_i D_i). \quad (\text{C.6})$$

We observe that  $s'_i$  takes the same place in developments than  $c_i$ . Indeed the probability of a CU to not be coded in Skip mode have the same use that the CU probability to occur. We can then anticipate the final equation of optimal quantizer as:

$$dQP_{k_\tau} = -str \left( \log_2(U_{k_\tau}) - \frac{1}{N_{Tot}} \sum_{t=1}^T \sum_{i=1}^N s'_i c_i \log_2(U_i) \right) \quad (\text{C.7})$$

# References

- [AACP16] E. Alshina, A. Alshin, K. Choi, and M. Park. Performance of JEM 1 tools analysis. In *Document JVET-B0044 3rd 2nd JVET Meeting: San Diego, CA, USA*, February 2016.
- [Arn75] D. Arnstein. Quantization error in predictive coders. *IEEE Transactions on Communications*, 23(4):423–429, April 1975.
- [ATR74] N. Ahmed, Natarajan T., and K. R. Rao. Discrete Cosine Transform. *IEEE Transactions on Computers*, C-23(1):90–93, January 1974.
- [Bjo01a] G. Bjontegaard. Calculation of average PSNR differences between RD-curves. In *ITU-T VCEG, Texas, USA, Proposal VCEG-M33*, Austin, Texas, April 2001.
- [Bjø01b] G. Bjøntegaard. VCEG-M33: Calculation of Average PSNR Differences Between RD-Curves, April 2001.
- [BLTR<sup>+</sup>17] M. Bichon, J. Le Tanou, M. Ropert, W. Hamidouche, L. Morin, and L. Zhang. Inter-Block Dependencies Consideration for Intra Coding in H.264/AVC and HEVC standards. In IEEE, editor, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017.
- [BLTR<sup>+</sup>18a] M. Bichon, J. Le Tanou, M. Ropert, W. Hamidouche, L. Morin, and L. Zhang. Low Complexity Joint RDO of Prediction Units Couples for HEVC Intra Coding. In IEEE, editor, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018.
- [BLTR<sup>+</sup>18b] M. Bichon, J. Le Tanou, M. Ropert, W. Hamidouche, L. Morin, and L. Zhang. Temporal Adaptive Quantization using accurate estimations of Inter and Skip probabilities. In IEEE, editor, *Picture Coding Symposium*, June 2018.
- [Bos13] F. Bossen. Common test conditions and software reference configurations. *Tech. Rep. JCTVC-L1100, Joint Collaborative Team on Video Coding (JCTVC)*, January 2013.
- [BvdPKS05] J. Breebaart, S. van de Par, A. Kohlrausch, and E. G. P. Schuijers. Parametric Coding of Stereo Audio. *EURASIP Journal on Advances in Signal Processing*, pages 1305–1322, December 2005.
- [CAS<sup>+</sup>17] J. Chen, E. Alshina, G. J. Sullivan, J. R. Ohm, and J. Boyce. Algorithm Description of Joint Exploration Model 7. In *JVET-G1001*, July 2017.



- [Cis14] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2013-2018, June 2014.
- [Cis18] Cisco. Cisco Visual Networking Index: Forecast and Trends, 2017–2022, November 2018.
- [CN07] Z. Chen and K. N. Ngan. Recent advances in rate control for video coding. *Signal Processing: Image Communication*, 22(1):19–38, January 2007.
- [CT06] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley and Sons, Hoboken, NJ, USA, 2006.
- [DM38] A. De Moivre. *Doctrine of Chances: Edition 2*. Woodfall, 1738.
- [Eri18] Ericsson. Ericsson Mobility Report, November 2018.
- [Eve63] H. Everett. Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources. *Operations Research*, 11(3):399–417, May 1963.
- [FAA<sup>+</sup>12] C. M. Fu, E. Alshina, A. Alshin, Y. W. Huang, C. Y. Chen, C. Y. Tsai, C. W. Hsu, S. M. Lei, J. H. Park, and W. J. Han. Sample Adaptive Offset in the HEVC Standard. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1755–1764, October 2012.
- [FCCPP17] A. Fiengo, G. Chierchia, M. Cagnazzo, and B. Pesquet-Popescu. Rate Allocation in Predictive Video Coding Using a Convex Optimization Framework. *IEEE Transactions on Image Processing*, 26(1):479–489, January 2017.
- [FN13] I. G. Fuss and D. J. Navarro. Open Parallel Cooperative and Competitive Decision Processes: A Potential Provenance for Quantum Probability Decision Models. *Topics in Cognitive Science*, pages 818–843, 2013.
- [fOM] Alliance fo Open Media. Av1 Bitstream and Decoding Process Specification, Online at: <https://aomedia.org/av1-bitstream-and-decoding-process-specification/>.
- [GG11] J. Garrett-Glaser. A novel macroblock-tree algorithm for high performance optimization of dependent video coding in H.264/AVC, 2011.
- [Gra90] R. M. Gray. *Source Coding Theory*. Kluwer Academic Publishers, Norwell, MA, USA, 1990.
- [Gro15] V. Grozman. Evaluating the CU-tree algorithm in an HEVC encoder, 2015.
- [GWY<sup>+</sup>16] Y. Gong, S. Wan, K. Yang, Y. Yang, and B. Li. Rate-Distortion-Optimization-Based Quantization Parameter Cascading Technique for Random-Access Configuration in H.265/HEVC. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, March 2016.
- [GZL16] Y. Gao, Ce. Zhu, and S. Li. Hierarchical Temporal Dependent Rate-Distortion Optimization for Low-Delay Coding. In IEEE, editor, *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2016.

- [GZLY17] Y. Gao, Ce. Zhu, S. Li, and T. Yang. Source Distortion Temporal Propagation Analysis for Random-Access Hierarchical Video Coding Optimization. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, December 2017.
- [HMK<sup>+</sup>10] W. J. Han, J. Min, I. K. Kim, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. M. Hong, M. S. Cheon, N. Shlyakhov, K. McCann, T. Davies, and J. H. Park. Improved Video Compression Efficiency Through Flexible Unit Representation and Corresponding Extension of Coding Tools. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 20(12):1709–1720, December 2010.
- [HOB<sup>+</sup>12] P. Helle, S. Oudin, B. Bross, D. Marpe, M.-O. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand. Block Merging for Quadtree-Based Partitioning in HEVC. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1720–1731, December 2012.
- [Huf52] D. A. Huffman. A Method for the Construction of Minimum-Redundancy Codes. In IEEE, editor, *Proceedings of the IRE*, September 1952.
- [IC15] S. K. Im and K. H. Chan. Multi-lambda search for improved rate-distortion optimization of H.265/HEVC. In IEEE, editor, *IEEE International Conference on Information, Communications and Signal Processing (ICICS)*, December 2015.
- [IGC15] S. K. Im, M. M. Ghandi, and K. H. Chan. Accurate, Non-integer Bit Estimation for H.265/HEVC and H.264/AVC Rate-Distortion Optimization. In IEEE, editor, *International Conference on Computer Science and Intelligent Communication*, June 2015.
- [IGL12] S. K. Im, M. M. Ghandi, and C. T. Lam. Non-integer bit estimation for rate-distortion optimized video coding. In IEEE, editor, *IEEE International Conference on Consumer Electronics (ICCE)*, March 2012.
- [ISOa] ISO/IEC. International Standard 14496-10: Advanced Video Coding.
- [ISOb] ISO/IEC. International Standard 23008-2: High Efficiency Video Coding.
- [ITa] ITU-T. Recommendation H.264: Advanced Video Coding.
- [ITb] ITU-T. Recommendation H.265: High Efficiency Video Coding.
- [Kay93] S. M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, Englewood Cliffs, N.J., USA, 1993.
- [KML<sup>+</sup>12] I.-K. Kim, J. Min, T. Lee, W.-J. Han, and J. Park. Block Partitioning Structure in the HEVC Standard. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1697–1706, December 2012.
- [Lap12] P. S. Laplace. *Théorie Analytique des Probabilités*. 1812.

- [LBH<sup>+</sup>12] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur. Intra Coding of the HEVC Standard. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1792–1801, December 2012.
- [LCFC15] Y.C. Liu, Z.Y. Chen, J.T. Fang, and P.C. Chang. SVM-Based Fast Intra CU Depth Decision for HEVC. In IEEE, editor, *IEEE Data Compression Conference (DCC)*, page 458, April 2015.
- [ICvH92] S. le Cessie and J.C. van Houwelingen. Ridge estimators in logistic regression. *Applied Statistics*, 41(1):191–201, 1992.
- [LJK11] W. Lin and C.-C. Jay Kuo. Perceptual visual quality metrics: A survey. *Journal of Visual Communication and Image Representation*, 22(4):297–312, 2011.
- [LJL<sup>+</sup>03] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz. Adaptive Deblocking Filter. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 13(7):614–619, July 2003.
- [LLLZ14] B. Li, H. Li, L. Li, and J. Zhang.  $\lambda$ -Domain Rate Control Algorithm for High Efficiency Video Coding. *IEEE Transactions on Image Processing*, 23(9):3841–3854, September 2014.
- [LOHK07] X. Li, N. Oertel, A. Hutter, and A. Kaup. Extended Lagrange Multiplier Selection for Hybrid Video Coding Using Interframe Correlation. In IEEE, editor, *Picture Coding Symposium (PCS)*, September 2007.
- [LOHK09] X. Li, N. Oertel, A. Hutter, and A. Kaup. Laplace Distribution Based Lagrangian Rate Distortion Optimization for Hybrid Video Coding. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 19(2):193–205, February 2009.
- [LTB18] J. Le Tanou and M. Blestel. Analysis of Emerging Video Codecs: Coding Tools, Compression Efficiency, and Complexity. In *SMPTE Symposium*, October 2018.
- [LW00] E. Y. Lam and Goodman J. W. A mathematical analysis of the DCT coefficient distributions for images. *IEEE Transactions on Image Processing (TIP)*, 9(10):1661–1666, October 2000.
- [LXL13] B. Li, J. Xu, and H. Li. Refining QP to improve coding efficiency in AVS. In IEEE, editor, *IEEE International Conference on Image Processing (ICIP)*, September 2013.
- [Lyy82] S. Lyyod. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, March 1982.
- [LZ94] T. Linder and R. Zamir. On the Asymptotic Tightness of the Shannon Lower Bound. *IEEE Transactions on Information Theory*, 40(6):2026–2031, November 1994.

- [LZZR18] W. Li, F. Zhao, E. Zhang, and P. Ren. Optimal Frame-Level Bit Allocation in HEVC with Distortion Dependency Model. In IEEE, editor, *IEEE International Conference on Big Data and Smart Computing (BigComp)*, January 2018.
- [MAH<sup>+</sup>17] A. Mercat, A. Arrestier, W. Hamidouche, M. Pelcat, and D. Menard. Energy Reduction Opportunities in an HEVC Real-Time Encoder. In IEEE, editor, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, June 2017.
- [Mak75] J. Makhoul. Linear prediction: A tutorial review. *Proceedings of the IEEE*, 63(4):561–580, April 1975.
- [Max60] J. Max. Quantizing for Minimum Distortion. *IRE Transactions on Information Theory*, 6(1):7–12, March 1960.
- [MHK<sup>+</sup>10] K. McCann, W.-J. Han, I.-K. Kim, J.-H. Min, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y.-M. Hong, M.-S. Cheon, and N. Shlyakhov. JCTVC-A124: Video coding technology proposal by Samsung (and BBC), April 2010.
- [MRB<sup>+</sup>14] K. McCann, C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. Sullivan. JCTVC-R1002: High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Encoder Description, July 2014.
- [MSW10] D. Marpe, H. Schwarz, and T. Wiegand. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 13(7):620–636, July 2010.
- [NBF<sup>+</sup>12] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Van der Auwera. HEVC Deblocking Filter. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1746–1754, October 2012.
- [OR95] A. Ortega and K. Ramchandran. Forward-adaptive quantization with optimal overhead cost for image and video coding with applications to MPEG video coders. *SPIE, Digital Video Compression: Algorithms and Technologies*, April 1995.
- [OR98] A. Ortega and K. Ramchandran. Rate-Distortion Methods for Image and Video Compression. *IEEE Signal Processing Magazine*, 15(6):23–50, November 1998.
- [OSS<sup>+</sup>12] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand. Comparison of the Coding Efficiency of Video Coding Standards - Including High Efficiency Video Coding (HEVC). *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1669–1684, December 2012.

- [PAZ<sup>+</sup>11] C. Pang, O. C. Au, F. Zou, J. Dai, and R. Cha. Optimal distortion redistribution in block-based image coding using successive convex optimization. In *International Conference on Multimedia and Expo (ICME)*, pages 1–5. IEEE, 2011.
- [PAZ<sup>+</sup>13a] C. Pang, O. C. Au, F. Zou, J. Dai, X. Zhang, and W. Dai. An Analytical Framework for Frame-Level Dependent Bit Allocation in Hybrid Video Coding. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, February 2013.
- [PAZ<sup>+</sup>13b] C. Pang, O. C. Au, F. Zou, X. Zhang, W. Hu, and P. Wan. Optimal dependent bit allocation for AVS intra-frame coding via successive convex approximation. In *International Conference on Image Processing (ICIP)*. IEEE, 2013.
- [PAZD11] C. Pang, O. C. Au, F. Zou, and J. Dai. An Analytical Framework for Frame-Level Dependent Bit Allocation in Hybrid Video Coding. In IEEE, editor, *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, October 2011.
- [Pur99] H. Purnhagen. Advances in Parametric Audio Coding. In IEEE, editor, *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, October 1999.
- [QXB<sup>+</sup>14] W. U. Qingbo, J. Xiong, L. U. O. Bing, C. Huang, and X. U. Linfeng. A Novel Joint Rate Distortion Optimization Scheme for Intra Prediction Coding in H.264/AVC. *IEICE Transactions on Information and Systems*, 97(4):989–992, 2014.
- [RDaM09] S. Rimac-Drlje, D. Žagar, and G. Martinović. Spatial Masking and Perceived Video Quality in Multimedia Applications. In *International Conference on Systems, Signals and Image Processing (IWSSIP)*, Chalkida, Greece, June 2009.
- [RG79] J. J. Rissanen and Langdon G. G. *Arithmetic Coding*, volume 23. IBM Journal of Research and Development, March 1979.
- [RLTBB17] M. Ropert, J. Le Tanou, M. Bichon, and M. Blestel. R-D Spatio-Temporal Adaptive Quantization based on Temporal Distortion Backpropagation in HEVC. In IEEE, editor, *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, October 2017.
- [ROV94] K. Ramchandran, A. Ortega, and M. Vetterli. Bit Allocation for Dependent Quantization with Applications to Multiresolution and MPEG Video Coders. *IEEE Transactions on Image Processing*, 3(5):553–545, September 1994.
- [RR13] M. Ropert and F. Ropert. RD Optimization of uniform threshold scalar quantization for Laplacian distributions. In IEEE, editor, *Picture Coding Symposium (PCS)*, San Jose, CA, USA, December 2013.
- [RS01] M. A. Robertson and R. L. Stevenson. DCT quantization noise in compressed images. In IEEE, editor, *IEEE International Conference on Image Processing (ICIP)*, October 2001.

- [SAD<sup>+</sup>12] L. Sun, O. C. Au, W. Dai, R. Zou, and S. Li. Modified distortion redistribution problem for High Efficiency Video Coding (HEVC). In *International Workshop on Multimedia Signal Processing (MMSP)*, pages 278–282. IEEE, 2012.
- [Say02] K. Sayood. *Lossless Compression Handbook*. Academic Press, San Diego, CA, USA, 2002.
- [SAZ<sup>+</sup>13] Y. Shi, O.C. Au, X. Zhang, H. Zhang, R. Ma, and L. Jia. Content Based Fast Prediction Unit Quadtree Depth Decision Algorithm for HEVC. In IEEE, editor, *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 225–228, 2013.
- [SB12] V. Sze and M. Budagavi. High Throughput CABAC Entropy Coding in HEVC. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1778–1791, October 2012.
- [SBS14] V. Sze, M. Budagavi, and G. J. Sullivan. *High Efficiency Video Coding (HEVC): Algorithms and architectures*. Integrated Circuits and Systems. Springer International Publishing, New York, NY, USA, 2014.
- [SCF<sup>+</sup>12] R. Sjöberg, Y. Chen, A. Fujibayashi, M. M. Hannuksela, J. Samuelsson, T. K. Tan, Y.-K. Wang, and S. Wenger. Overview of HEVC High-Level Syntax and Reference Picture Management. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1858–1870, December 2012.
- [Sha48] C.E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3):379–423, July 1948.
- [SJN<sup>+</sup>12a] J. Sole, R. Joshi, N. Nguyen, T. Ji, M. Karczewicz, G. Clare, F. Henry, and A. Duenas. Transform Coefficient Coding in HEVC. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1765–1777, December 2012.
- [SJN<sup>+</sup>12b] J. Sole, R. Joshi, N. Nguyen, T. Ji, M. Karczewicz, G. Clare, F. Henry, and A. Duenas. Transform Coefficient Coding in HEVC. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1765–1777, December 2012.
- [SKG<sup>+</sup>14] J. Stankowski, D. Karwowski, T. Grajek, K. Wegner, J. Siast, K. Klimaszewski, O. Stankiewicz, and M. Domański. Bitrate distribution of syntax elements in the HEVC encoded video. In *International Conference on Signals and Electronic Systems (ICSES)*, September 2014.
- [SMW05] H. Schwarz, D. Marpe, and T. Wiegand. Hierarchical B pictures, Joint Video Team (JVT), Document JVT-P014, Poznan, July 2005.
- [SOHW12] G.-J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand. Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 22(12):1649–1668, December 2012.

- [SRS<sup>+</sup>16] H. Schwarz, C. Rudat, M. Siekmann, B. Bross, D. Marpe, and T. Wiegand. Coding Efficiency / Complexity Analysis of JEM 1.0 coding tools for the Random Access Configuration. In *Document JVET-B0044 3rd 2nd JVET Meeting: San Diego, CA, USA*, February 2016.
- [Sul05] G. J. Sullivan. Adaptive quantization encoding technique using an equal expected-value rule. Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, JVT-N011, January 2005.
- [SW98] G.-J. Sullivan and T. Wiegand. Rate-Distortion Optimization for Video Compression. *IEEE Signal Processing Magazine*, pages 1755–1764, November 1998.
- [TC06] A. Tanizawa and T. Chujoh. Simulation results of Adaptive Quantization Matrix Selection on KTA software. VCEG Document, VCEG-AC07, July 2006.
- [TG12] G. Tian and S. Goto. Content adaptive prediction unit size decision algorithm for HEVC intra coding. In IEEE, editor, *Picture Coding Symposium (PCS)*, pages 405–408, May 2012.
- [TLSS09] A. M. Tourapis, A. Leontaris, K. Sühring, and G. Sullivan. JVT-AE010: H.264/14496-10 AVC Reference Software Manual, July 2009.
- [Twi] Twitch. [Online]. Twitch Statistics and Charts: <https://twitchtracker.com/statistics>.
- [WBSS04] Z. Wang, A.-C. Bonvik, H.-R. Sheikh, and E.-P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.
- [WG01] T. Wiegand and B. Girod. Lagrange multiplier selection in hybrid video coder control. In IEEE, editor, *IEEE International Conference on Image Processing (ICIP)*, pages 542–545, Thessaloniki, October 2001.
- [Wie03] M. Wien. Variable block-size transforms for H.264/AVC. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 13(7):604–613, 2003.
- [Wie15] M. Wien. *High Efficiency Video Coding: Coding Tools and Specification*. Signals and Communication Technology. Springer-Verlag Berlin Heidelberg, 2015.
- [Win05] S. Winkler. *Digital Video Quality: Vision Models and Metrics*. John Wiley and Sons, Ltd, 2005.
- [WLV00] J. Wen, M. Luttrell, and J. Villasenor. Trellis-based R-D optimal quantization in H.263+. *IEEE Transactions on Image Processing*, 9(8):1431–1434, August 2000.
- [WMW<sup>+</sup>13] S. Wang, S. Ma, S. Wang, D. Zhao, and W. Gao. Rate-GOP Based Rate Control for High Efficiency Video Coding. *IEEE Journal of Selected Topics in Signal Processing (J-STSP)*, December 2013.

- [WPU<sup>+</sup>09] L. Wang, L. M. Po, Y. Md. S. Uddin, K. M. Wong, and S. Li. A novel weighted cross prediction for H.264 intra coding. In IEEE, editor, *IEEE International Conference on Multimedia and Expo (ICME)*, July 2009.
- [WRSW15] M. Winken, A. Roth, H. Schwarz, and T. Wiegand. Multi-frame optimized quantization for high efficiency video coding. In IEEE, editor, *Picture Coding Symposium (PCS)*, July 2015.
- [WS11] T. Wiegand and H. Schwarz. *Source Coding: Part I of Fundamentals of Source and Video Coding*. Foundations and Trends in Signal Processing, 2011.
- [WSBL03] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 13(7):560–576, July 2003.
- [WSJ<sup>+</sup>03] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G.-J. Sullivan. Rate-constrained coder control and comparison of video coding standards. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 13(7):688–703, July 2003.
- [x26] x265. [Online]. Available: <https://bitbucket.org/multicoreware/x265>.
- [XJGZ07] L. Xu, X. Ji, W. Gao, and D. Zhao. Laplacian Distortion Model (LDM) for Rate Control in Video Coding. In *Advances in Multimedia Information Processing – Pacific Rim Conference on Multimedia (PCM)*, pages 638–646, Hong Kong, China, December 2007.
- [XJY<sup>+</sup>17] G. Xiang, H. Jia, M. Yang, Y. Li, and X. Xie. A novel adaptive quantization method for video coding. *Multimedia Tools and Applications*, 77(12):14817–14840, 2017.
- [XSX<sup>+</sup>15] J. Xie, L. Song, R. Xie, Z. Luo, and X. Wang. Temporal Dependent Bit Allocation Scheme for Rate Control in HEVC. In IEEE, editor, *IEEE Workshop on Signal Processing Systems (SiPS)*, December 2015.
- [XZZ<sup>+</sup>16] L. Xu, C. Zhu, Y. Zhou, Y. Wang, and Y. Gao. Introducing GOP-level quantization parameter offset in high efficiency video coding. In IEEE, editor, *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, June 2016.
- [YCJ08] J. You, C. Choi, and J. Jeong. Modified rate distortion optimization using inter-block dependence for H.264/AVC intra coding. *IEEE Transactions on Consumer Electronics*, 54(3):1383–1388, 2008.
- [YGCZ08] S. Yu, Y. Gao, J. Chen, and J. Zhou. Distance-based weighted prediction for H.264 intra coding. In IEEE, editor, *International Conference on Audio, Language and Image Processing (ICALIP)*, July 2008.
- [YHY09] X. Yu, D. K. He, and E. H. Yang. Adaptive quantization with balanced distortion distribution and its application to H.264 intra coding. In IEEE, editor, *IEEE International Conference on Image Processing (ICIP)*, November 2009.



- [YTT13a] C. Yeo, H. L. Tan, and Y. H. Tan. On Rate Distortion Optimization Using SSIM. *Transactions on Circuits and Systems for Video Technology (TCSVT)*, 23(7):1170–1181, 2013.
- [YTT13b] C. Yeo, H. L. Tan, and Y. H. Tan. SSIM-Based Adaptive Quantization in HEVC. In IEEE, editor, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2013.
- [YWG<sup>+</sup>17] K. Yang, S. Wan, C. Gong, H. R. Wu, and Y. Feng. An efficient Lagrangian multiplier selection method based on temporal dependency for rate–distortion optimization in H.265/HEVC. *Signal Processing: Image Communication*, 57:68–75, September 2017.
- [YWGF17] K. Yang, S. Wan, C. Gong, and Y. Feng. Content adaptive quantization parameter cascading for random-access structure in HEVC. In IEEE, editor, *IEEE International Conference on Image Processing (ICIP)*, September 2017.
- [YWGW16] Y. Yang, S. Wan, Y. Gong, and K. Wang. Adaptive quantization parameter cascading for random-access prediction in H.265/HEVC based on dependent R-D models. In IEEE, editor, *IEEE International Conference on Image Processing (ICIP)*, September 2016.
- [YWXS15] H. Yin, Z. Wang, Z. Xia, and Y. Shen. Temporally Adaptive Quantization Algorithm in Hybrid Video Encoder. In Springer, editor, *Advances in Multimedia Information Processing – Pacific Rim Conference on Multimedia (PCM)*, November 2015.
- [YZFP12] T. Yang, C. Zhu, X. Fan, and Q. Peng. Source Distortion Temporal Propagation Model for Motion Compensated Video Coding Optimization. In IEEE, editor, *IEEE International Conference on Multimedia and Expo (ICME)*, September 2012.
- [ZMG10] L. Zhang, S. Ma, and W. Gao. Enhanced intra coding with joint line and pixel prediction. In IEEE, editor, *International Packet Video Workshop (PV)*, December 2010.
- [ZWC16] T. Zhao, Z. Wang, and C. W. Chen. Adaptive Quantization Parameter Cascading in HEVC Hierarchical Coding. *IEEE Transactions on Image Processing*, 25(7):2997–3009, July 2016.
- [ZWTZ17] Y. Zhou, H. Wang, L. Tian, and Ce. Zhu. Temporal correlation based hierarchical quantization parameter determination for HEVC video coding. *IEEE International Conference on Image Processing (ICIP)*, September 2017.



## AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

**Titre de la thèse:**

Considération des dépendances pour l'optimisation débit-distortion globale : application à HEVC

**Nom Prénom de l'auteur : BICHON MAXIME**

**Membres du jury :**

- Monsieur ZHU Ce
- Monsieur LE TANOU Julien
- Madame GUILLEMOT Christine
- Monsieur WIEN Mathias
- Madame MORIN Luce
- Monsieur HAMIDOUCHE Wassim
- Monsieur CAGNAZZO Marco
- Monsieur DUFAUX Frédéric

Président du jury : *Christine GUILLEMOT*

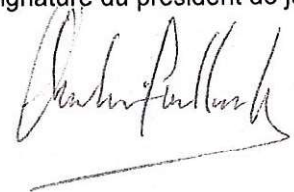
Date de la soutenance : 01 Mars 2019

Reproduction de la these soutenue

- Thèse pouvant être reproduite en l'état  
 Thèse pouvant être reproduite après corrections suggérées

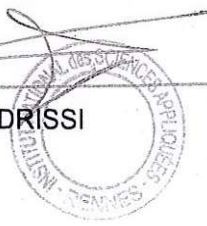
Fait à Rennes, le 01 Mars 2019

Signature du président de jury



Le Directeur,

M'hamed DRISSI



---

**Titre :** Considération des dépendances pour l'optimisation débit-distorsion globale : application à HEVC

**Mots clés :** Compression Vidéo, Dépendances de codage, Optimisation débit-distorsion globale, HEVC

**Résumé :** La compression vidéo est critique pour le déploiement de nouvelles technologies vidéo, telles que l'ultra haute définition (UHD) ou le débit d'images élevé (HFR). De nouvelles normes sont périodiquement conçues afin de répondre au besoin d'une meilleure efficacité de codage vidéo. Une nouvelle norme introduit généralement de nouveaux outils de compression qui permettent d'accroître les capacités de la norme précédente. Cependant, une norme spécifie uniquement le processus de décodage. Pour s'assurer une compression optimale, il est nécessaire d'optimiser l'utilisation de ces outils et donc de l'encodage. Alors que de nombreux encodeurs sont basés sur l'optimisation indépendante des paramètres de codage, ce qui conduit à une succession d'optimums locaux, cette thèse adresse l'optimisation jointe de ces paramètres afin d'atteindre une efficacité globale.

Tout d'abord, nous fournissons une description exhaustive des dépendances qui lient les paramètres de codage au sein de la dernière norme de compression vidéo publiée: HEVC. Ensuite, nous estimons les possibilités, en termes d'efficacité de codage, d'optimiser plusieurs paramètres de codage conjointement et exhaustivement, dans le cas de codage Intra-image. Enfin, nous proposons de modéliser la propagation de la distorsion temporelle entre blocs de pixels, introduite par la prédiction temporelle. A partir d'un modèle mathématique simplifié, nous pouvons obtenir une solution analytique qui définit le quantificateur optimal pour chaque bloc de pixels. Les quantificateurs optimaux permettent d'atteindre une efficacité de codage élevée, pour une séquence de plusieurs images, du point de vue de l'optimisation débit-distorsion.

---

**Title:** Dependencies consideration for global rate-distortion optimization: application to HEVC

**Keywords:** Video Compression, Coding Dependencies, Global Rate-Distortion Optimization, HEVC

**Abstract:** Video compression remains one of the key challenges for the deployment of new video technologies, such as Ultra High Definition (UHD) or High Frame Rate (HFR). Requirements for better coding efficiency are periodically answered by the release of new standards, introducing additional coding tools with coding capabilities beyond previous standards. However, encoding optimization is necessary for guaranteeing that an encoder can provide the best efficiency in the scope of a standard. Many encoders are based on independent optimization of coding parameters, which leads to successive local optima. In this thesis, the emphasis is put on parameters joint optimization to achieve global optimum.

First, we provide an exhaustive description of coding parameters dependencies related to the latest released video coding standard, HEVC. Then, we try to estimate the opportunities, in terms of coding efficiency, to exhaustively and jointly optimize multiple coding parameters, in the use case of Intra-picture coding. Finally, we propose to model the temporal distortion propagation between blocks of pixels, introduced by the Inter-frame prediction. Based on a simplified mathematical model, we can draw an analytical solution that sets the optimal quantizer for each block of pixels. Optimal quantizers allow achieving high coding efficiency, in a sequence of frames, from a rate-distortion optimization standpoint.