



**HAL**  
open science

## Approches théoriques en codage vidéo robuste multi-terminal.

Olivier Crave

► **To cite this version:**

Olivier Crave. Approches théoriques en codage vidéo robuste multi-terminal.. domain\_other. Télécom ParisTech, 2008. English. NNT: . pastel-00004774

**HAL Id: pastel-00004774**

**<https://pastel.hal.science/pastel-00004774>**

Submitted on 10 Apr 2009

**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.



École Doctorale  
d'Informatique,  
Télécommunications  
et Électronique de Paris

# Thèse

présentée pour obtenir le grade de docteur  
de l'Ecole Nationale Supérieure des Télécommunications

Spécialité : Signal et Images

## Olivier CRAVE

### Approches théoriques en codage vidéo robuste multi-terminal.

Soutenue le 15 décembre 2008 devant le jury composé de

Michel Kieffer  
Marc Antonini  
Touradj Ebrahimi  
Vincent Ricordel  
Christine Guillemot  
Béatrice Pesquet-Popescu

Président  
Rapporteurs  
  
Examineurs  
Directrices de thèse



*« I have to write things down to feel I fully comprehend them. »*  
*Haruki Murakami – Norwegian Wood*

---



# Remerciements

Je remercie tout d'abord mes directrices de thèse, Béatrice Pesquet-Popescu et Christine Guillemot, pour m'avoir fait confiance et pour m'avoir guidé, encouragé, conseillé tout au long de ces trois années. Je tiens de plus à remercier Christophe Tillier pour sa précieuse collaboration au début de ma thèse.

J'adresse mes remerciements à tous les membres de mon jury : Michel Kieffer, maître de conférences à l'École Supérieure d'Électricité (SUPELEC) qui m'a fait l'honneur de présider ce jury ; Marc Antonini, directeur de recherche au laboratoire d'Informatique, Signaux et Systèmes de Sophia Antipolis (I3S) et Touradj Ebrahimi, professeur à l'École Polytechnique Fédérale de Lausanne (EPFL) qui ont bien voulu accepter la charge de rapporteur ; Vincent Ricordel, maître de conférences à l'Institut de Recherche en Communication et Cybernétique de Nantes (IRCCyN) qui a bien voulu juger ce travail.

Je ne saurais trop remercier tout ceux qui m'ont apporté leur aide ou leur soutien, en particulier Cédric Herzet, Khaled Lajnef, Denis Kubasov, Aline Roumy et tous les autres, qui me pardonneront de ne pas les citer.

Enfin, je salue tous les membres de l'équipe Temics que j'ai eu le plaisir de côtoyer durant ces quelques années passées à Rennes. Et je remercie les membres du département TSI pour être venus assister à ma soutenance.

---



---

# Table des matières

<b>Introduction</b>	<b>9</b>
<b>I Etat de l'art</b>	<b>15</b>
<b>1 Codage par descriptions multiples</b>	<b>17</b>
1.1 Introduction . . . . .	17
1.2 Principes théoriques . . . . .	19
1.2.1 Présentation du problème . . . . .	20
1.2.2 Limites théoriques . . . . .	21
1.2.3 Codage à $N$ descriptions . . . . .	23
1.3 Descriptions multiples basées sur la quantification . . . . .	26
1.3.1 Quantification scalaire à descriptions multiples . . . . .	26
1.3.2 Quantification vectorielle à descriptions multiples . . . . .	28
1.4 Descriptions multiples basées sur la transformation . . . . .	28
1.4.1 Appariement de doublets . . . . .	29
1.4.2 Descriptions multiples basées sur une décomposition en sous-bandes	30
1.4.3 Descriptions multiples basées sur les trames . . . . .	31
1.5 Descriptions multiples basées sur les codes correcteurs d'erreurs . . . . .	32
1.6 Réseaux et descriptions multiples . . . . .	34
1.7 Applications en vidéo . . . . .	35
1.8 Conclusion . . . . .	37
<b>2 Codage de source distribué</b>	<b>39</b>
2.1 Introduction . . . . .	39
2.2 Principes théoriques . . . . .	40
2.2.1 Codage de Slepian-Wolf . . . . .	40
2.2.2 Limites théoriques du codage de source avec information adjacente .	41
2.2.3 Codage de source multiterminal . . . . .	44
2.3 Techniques de codage de Slepian-Wolf . . . . .	44
2.3.1 Solutions basées sur le codage de source . . . . .	44
2.3.1.1 Technique de Al Jabri et Al-Issa . . . . .	45
2.3.1.2 Technique de Zhao et Effros . . . . .	46
2.3.2 Solutions basées sur le codage de canal . . . . .	48
2.3.2.1 Codage par cosets : DISCUS . . . . .	49
2.3.2.2 Solutions basées sur les turbo codes . . . . .	50
2.3.2.3 Solutions basées sur les codes LDPC . . . . .	51
2.4 Techniques de codage de Wyner-Ziv . . . . .	53

---



---

2.4.1	Solution basée sur DISCUS . . . . .	53
2.4.2	Solutions basées sur les turbo codes . . . . .	55
2.4.3	Solutions basées sur les codes LDPC . . . . .	56
2.5	Codage vidéo distribué . . . . .	57
2.5.1	Codage vidéo distribué sans voie de retour : PRISM . . . . .	57
2.5.1.1	Architecture de l'encodeur . . . . .	58
2.5.1.2	Architecture du décodeur . . . . .	60
2.5.2	Codage vidéo distribué avec voie de retour : le codeur de Stanford . . . . .	61
2.5.2.1	Architecture de l'encodeur . . . . .	61
2.5.2.2	Architecture du décodeur . . . . .	62
2.5.3	Derniers développements . . . . .	63
2.6	Conclusion . . . . .	66
<b>II Contributions</b>		<b>69</b>
<b>3</b>	<b>Avant-propos</b>	<b>71</b>
3.1	Codage de Wyner-Ziv pour une transmission vidéo sur des réseaux bruités . . . . .	71
3.2	Codage de Wyner-Ziv par couches . . . . .	74
3.3	Codage de source distribué robuste . . . . .	75
<b>4</b>	<b>Codage vidéo par descriptions multiples protégées par des principes de codage de source distribué</b>	<b>79</b>
4.1	Introduction . . . . .	79
4.2	Schémas de codage par descriptions multiples . . . . .	81
4.3	Codage de descriptions systématique avec pertes dans le domaine pixel . . . . .	84
4.4	Codage de descriptions systématique avec pertes dans le domaine MCTF . . . . .	88
4.5	Résultats de simulation . . . . .	90
4.5.1	Analyse des performances des schémas de codage MD . . . . .	90
4.5.2	Analyse des performances des schémas de codage MD distribué . . . . .	91
4.6	Conclusion . . . . .	100
<b>5</b>	<b>Codage par descriptions multiples avec information adjacente</b>	<b>103</b>
5.1	Introduction . . . . .	103
5.2	Contexte théorique . . . . .	105
5.2.1	Codage sans perte . . . . .	105
5.2.2	Codage avec perte . . . . .	105
5.3	Quantification scalaire à descriptions multiples avec information adjacente . . . . .	107
5.4	Décodage croisé de descriptions multiples avec information adjacente . . . . .	108
5.4.1	Décodage croisé de turbo codes avec information adjacente . . . . .	109
5.4.2	Décodage croisé de codes LDPC avec information adjacente . . . . .	111
5.5	Quantification inverse optimale . . . . .	112
5.6	Résultats expérimentaux sur des sources gaussiennes . . . . .	113
5.6.1	MDSQ avec information adjacente . . . . .	114
5.6.2	Décodage croisé de descriptions multiples avec information adjacente . . . . .	115
5.7	Conclusion . . . . .	117

---

---

<b>6</b>	<b>Codage vidéo distribué robuste</b>	<b>119</b>
6.1	Introduction . . . . .	119
6.2	Codage vidéo par descriptions multiples avec information adjacente . . . . .	120
6.2.1	Principes . . . . .	120
6.2.2	Codec vidéo . . . . .	121
6.2.3	Reconstruction optimale . . . . .	124
6.3	Résultats expérimentaux . . . . .	125
6.4	Conclusion . . . . .	132
	<b>Conclusion</b>	<b>134</b>
	<b>Glossaire</b>	<b>139</b>
<b>A</b>	<b>Les turbo codes</b>	<b>141</b>
<b>B</b>	<b>Les codes LDPC</b>	<b>145</b>
	<b>Publications</b>	<b>148</b>
	<b>Bibliographie</b>	<b>161</b>
	<b>Table des figures</b>	<b>163</b>

---



# Introduction

## Contexte

Les standards de compression connus aujourd'hui mettent en oeuvre un codage prédictif compensé en mouvement afin d'exploiter la corrélation temporelle, c'est à dire les similarités entre les images successives dans une séquence. Ceci conduit à une complexité de l'encodeur de l'ordre de 5 à 10 fois supérieure à celle du décodeur. Ces approches sont bien adaptées à des systèmes où, par exemple, des terminaux mobiles viennent télécharger ou streamer des vidéos à partir de serveurs. Cependant, elles sont peu adaptées pour des applications où l'encodage doit être réalisé sur des terminaux à faible capacité, par exemple un terminal mobile transmettant des flux vidéo vers une station de base fixe, pour des services de type MMS ou peer-to-peer avec des terminaux mobiles, ou bien pour la transmission en flux continu de séquences d'images par une multitude de caméras (environnement multi-vue).

Ainsi, nous avons vu apparaître très récemment un nouveau domaine de recherche connu sous le nom de codage de source distribué (ou encore de codage de source avec information adjacente) dont les fondements théoriques reposent sur les théorèmes de Slepian-Wolf et de Wyner-Ziv respectivement dans les cas du codage quasi- sans perte et avec pertes.

Le codage de source distribué suppose l'existence de sources multiples corrélées : c'est le cas des images successives dans une séquence vidéo. Afin de réduire la complexité du codeur, chacune de ces sources est codée indépendamment : cela peut être vu comme un cadre de communication multipoint-point avec des messages (les images) corrélés. La corrélation entre les images, au lieu d'être exploitée par le codeur, va être exploitée par le décodeur. Si nous considérons deux signaux corrélés, l'un peut être vu comme une version bruitée de l'autre. Le problème peut ainsi être traité en utilisant des concepts et des outils du codage de canal et du codage conjoint source-canal.

Le codage de source distribué peut également être mis à profit pour constituer une alternative aux techniques de correction d'erreur nécessaires à la protection d'informations transmises sur des canaux bruités. Pour faire face à des dégradations de ce type qui peuvent être brutales, l'approche actuelle en codage vidéo est de générer des représentations dites scalables qui seront ensuite protégées, éventuellement de manière inégale en fonction de leur importance, par du codage canal. Cette stratégie permet de limiter l'impact des erreurs de transmission. Une alternative à ce schéma classique peut être apportée là encore par le codage de source distribué qui se substitue alors au codage canal.

Le théorème de Wyner-Ziv donne les bornes débit-distorsion du codage avec information adjacente et suppose l'information adjacente connue parfaitement au décodeur. Cette hypothèse n'est pas réaliste pour des applications de compression vidéo. Une extension de ce principe au cas où l'information adjacente est comprimée avec un critère de fidélité est connue sous le nom de codage multiterminal.

---

## Objectifs

La thèse a pour objectif l'étude de techniques de compression vidéo robuste dans un contexte de codage multi-terminal. Ce domaine de recherche s'avère avoir de fortes similarités et dualités avec le codage par descriptions multiples. Ces dualités se traduisent par des fondements communs de théorie de l'information qui permettent d'envisager la transposition d'outils d'un domaine à l'autre.

Nous chercherons à réduire les effets produits par la propagation d'erreurs dans un système de codage vidéo par descriptions multiples dans le domaine temporel en utilisant des principes de codage de source distribué. Et, à l'inverse, nous entreprendrons d'augmenter la robustesse d'une architecture de codage vidéo distribué en utilisant des principes de codage par descriptions multiples.

## Organisation du document

Ce document est composé de 6 chapitres auxquels s'ajoutent la présente introduction et la conclusion.

Le chapitre 1 présente le problème général du codage par descriptions multiples, ainsi que les résultats théoriques obtenus dans le domaine de la théorie de l'information sur ce sujet. La théorie de l'information fournissant ici des limites de performances mais ne donnant pas de méthodes directes pour obtenir ces résultats, nous présenterons alors différentes techniques de codage par descriptions multiples. Les propriétés théoriques du codage par descriptions multiples sont ensuite étudiées dans le cadre d'une utilisation sur différents types de canaux et réseaux. Le chapitre se termine sur un état de l'art des codeurs vidéo par descriptions multiples rangés en trois familles, suivant qu'ils remédient directement ou non au problème de drift et suivant la redondance utilisée pour composer les descriptions.

Le chapitre 2 est un état de l'art du codage de source distribué. Après avoir défini les principes théoriques, sont présentées les principales techniques de codage sans perte et avec pertes, en commençant par celles qui utilisent le codage de source avant de voir celles reposant sur le codage de canal. Puis, nous verrons en détail les deux architectures de codage vidéo distribué mono-vue et nous intéresserons aux récentes avancées dans ce domaine.

Le chapitre 3 examine les champs d'investigation en codage vidéo robuste reposant sur le codage de source distribué et parfois le codage par descriptions multiples. Les trames codées en Wyner-Ziv par un code de canal dans un système de codage vidéo distribué sont «naturellement» protégées contre les erreurs de transmission à condition que les erreurs ne dépassent pas le pouvoir de correction du code. Nous verrons que le codage de Wyner-Ziv peut également servir à atténuer la propagation d'erreurs dans un codeur vidéo prédictif classique lors d'une transmission sur un canal bruité. Ensuite, on s'intéressera au codage Wyner-Ziv par couches où chaque couche d'amélioration est codée séparément et décodée à partir des couches les plus basses. Enfin, le problème de codage de source distribué robuste sera présenté, il consiste en une généralisation du problème de codage multiterminal et du problème de codage par descriptions multiples. Le cas particulier du codage par descriptions multiples avec information adjacente qui considère uniquement la robustesse pour la source codée en plusieurs descriptions, sera décrit.

Le chapitre 4 présente quatre schémas de codage par descriptions multiples dans le domaine temporel. Les descriptions sont codées par un codeur  $t+2D$  avec un filtrage temporel compensé en mouvement. Les performances des schémas sont comparés selon leur manière

---

de générer les descriptions et le filtre employé. Dans une deuxième partie, on proposera d'augmenter la robustesse de ces schémas en leur adjoignant un flux d'information codé en Wyner-Ziv correspondant aux trames manquantes dans chacune des descriptions. Ce flux additionnel, uniquement utilisé lorsqu'une description est perdue, permet de limiter les fluctuations en qualité au cours du temps.

Le chapitre 5 présente un schéma de codage de source distribué robuste asymétrique. Le système repose sur un codage par descriptions multiples avec information adjacente où la génération des descriptions est effectuée par une quantification scalaire à descriptions multiples. Après un rapide rappel du contexte théorique, nous donnerons une description de l'architecture de codage envisagé. Pour augmenter l'efficacité du décodeur central, nous proposerons d'exploiter à la fois la corrélation entre l'information adjacente et les descriptions mais également entre les descriptions elles-mêmes. Les résultats de simulation obtenus avec des sources gaussiennes seront comparés avec les limites théoriques. Nous étudierons en particulier l'influence de la corrélation avec l'information adjacente et l'efficacité du décodage central en fonction de la redondance introduite par le codeur.

Le chapitre 6 introduit un système de codage vidéo robuste basé sur les techniques vues dans le chapitre 5. La séquence vidéo est tout d'abord séparée en trames clé et en trames codées en Wyner-Ziv. Puis, une quantification scalaire à descriptions multiples génère deux descriptions pour chaque séquence. Une description de trames clé est combinée avec une description de trames codées en Wyner-Ziv pour former une description représentant l'intégralité de la séquence d'entrée. La robustesse vient alors de la diversité introduite à l'encodeur. En dehors des performances obtenues avec ce schéma pour le décodeur central et les décodeurs latéraux, nous porterons aussi une attention particulière sur l'influence du décodage croisé sur les performances du décodeur central.

---



Première partie

Etat de l'art

---





# Chapitre 1

## Codage par descriptions multiples

*Ce chapitre présente le problème du codage par descriptions multiples, ainsi que les limites théoriques obtenues par la théorie de l'information dans ce domaine. Une grande part de ce chapitre est consacrée à l'état de l'art des nombreuses solutions de codage par descriptions multiples disponibles dans la littérature. Enfin, on s'intéressera à l'application de ces techniques dans le domaine du codage vidéo.*

### 1.1 Introduction

Les réseaux tels qu'Internet sont de plus en plus nombreux et de plus en plus hétérogènes. Cette hétérogénéité et la présence de congestion sont les causes de plusieurs problèmes, principalement des problèmes de délais difficilement prévisibles, des pertes d'information et de la latence. L'enjeu est alors de permettre une qualité de service (en anglais Quality of Service, QoS) suffisante, même à bas débit, indépendamment de la distribution des erreurs sur les canaux, et éventuellement permettre des applications en temps réel avec plusieurs utilisateurs simultanément. Plusieurs technologies existent pour répondre à ces attentes.

L'hétérogénéité ajoute des erreurs et de la congestion. Si le débit de transmission ne correspond pas à la capacité du réseau, certains paquets peuvent être perdus. Le système de distribution de paquets a la possibilité de donner une priorité différentes aux paquets : les paquets les plus importants bénéficient alors d'un traitement préférentiel, tandis que les paquets les moins importants sont les premiers à être éliminés en cas de congestion. Cependant, sur la plupart des réseaux, il est difficile de prévoir quels paquets seront perdus. La probabilité de perte de paquets n'est pas constante ; au contraire, elle peut varier considérablement. Cette incertitude rend la conception de système de transmission d'information très difficile. Habituellement, deux options s'offrent alors :

- le système peut être conçu pour répondre au pire scénario possible,
- ou il peut être adaptatif.

S'il est conçu pour le pire scénario possible, il sera particulièrement inefficace lorsque le canal de transmission devient meilleur, c'est-à-dire la plupart du temps. Inversement, s'il est conçu pour s'adapter aux conditions du réseau, il sera probablement trop lent pour répondre à l'état actuel du réseau.

Une des techniques les plus connues pour améliorer la fiabilité est la retransmission des paquets perdus. Cette technique, appelée «Automatic Repeat reQuest» (ARQ), est à la base du protocole «Transmission Control Protocol» (TCP). Si les pertes sont sporadiques,

---

cette technique est très efficace : la plupart des paquets ne sont envoyés qu'une seule fois. Mais si les pertes sont fréquentes, les retransmissions systématiques peuvent augmenter les risques de congestion et également le taux de pertes, ce qui conduit à un cercle vicieux (qui peut cependant être évité dans le cas de TCP). La retransmission est très utile dans les communication point à point où un canal de retour est présent. Cependant, lors d'une transmission vers plusieurs récepteurs, le diffuseur aura des difficultés pour s'occuper de toutes les requêtes de retransmission en même temps. Le délai supplémentaire dû à la retransmission nécessite au moins un temps de transfert correspondant au temps nécessaire pour qu'un paquet effectue un aller-retour entre le transmetteur et le récepteur. Mais chaque paquet retransmis peut lui aussi être perdu, et le délai peut alors devenir arbitrairement long. Cette incertitude sur les délais peut être rédhibitoire pour le streaming de vidéo car elle peut conduire à une interruption de diffusion.

Une autre technique consiste à utiliser les principes de codage de canal pour augmenter la robustesse sur des canaux à pertes, elle consiste à transmettre des paquets redondants qui permettent de corriger les erreurs dues à la perte de paquets au décodeur. Cette technique est appelée «Forward Error Correction» (FEC). Si le taux de pertes est connu, la redondance introduite peut être choisie de façon à compenser les pertes prévues. Cependant, en pratique, non seulement le taux de pertes n'est pas connue, mais il peut également varier au cours du temps. Cela, couplé avec le fait que cette technique offre des performances *tout ou rien*, rend un tel système très difficile à mettre en place : le taux d'erreurs peut être sous estimé ou sur estimé. Si les pertes sont trop nombreuses, les capacités de correction peuvent être dépassées. Une redondance supplémentaire ne sera pas suffisante et les pertes ne seront pas compensées. La qualité décodée sera alors très mauvaise (on parle alors d'effet de *cliff* qui correspond à une chute brutale de la qualité). De ce fait, il est souvent préférable de ne pas prendre de risques et de considérer des systèmes qui répondent au pire scénario possible en choisissant d'augmenter la redondance au dépend de l'information à transmettre. Si les erreurs sont plus faibles que prévu, les pertes seront compensées et la QoS garantie. Cependant, la capacité du canal aura été mal exploitée, car moins de redondance aurait pu permettre une plus faible compression du signal à transmettre. Un système adaptatif pourrait contrôler dynamiquement le taux de redondance et le taux de compression de l'information, mais cette solution est rarement envisagée en pratique du fait de sa complexité.

Plus un codeur est efficace, plus un paquet d'information est vital. Lorsque la compression est très efficace, la perte d'un paquet peut avoir un effet désastreux sur les performances. Dans ce cas, il est nécessaire d'utiliser des techniques de correction d'erreurs tels que des mécanismes de type FEC pour réduire la probabilité d'une telle éventualité. Réciproquement, lorsque le taux de compression est faible, la perte d'un paquet n'a que peu de conséquences. Dans ce cas, des techniques de masquage d'erreurs (en anglais *error concealment*) existent pour réduire ou même totalement masquer les effets d'une perte.

Ainsi, un système de compression, pour être parfaitement réglé, doit prendre en compte les effets des pertes, l'efficacité des techniques de masquage d'erreurs au récepteur et l'efficacité des mécanismes de correction d'erreurs. La bande passante disponible peut alors être utilisée de manière optimale pour transmettre les données et la redondance. En d'autres mots, il est toujours utile d'optimiser les paramètres de codage de source et de codage de canal conjointement (on parle alors de codage conjoint source-canal). Dans le cas de communications de contenu multimédia, cela signifie exploiter la résistance aux erreurs qui peut être directement présente dans les flux d'information multimédia compressés, plutôt que d'utiliser des codes FEC ou des protocoles de communication complexes.

---

---

Les codeurs vidéo utilisent un certain nombre de techniques pour compresser efficacement la vidéo : la prédiction (aussi connue sous le nom d'estimation de mouvement de compensation), la transformée, la quantification et le codage entropique. La prédiction est une des techniques les plus importantes d'un point de vue d'efficacité en compression : la trame actuelle est prédite à partir de ses trames voisines. A cause de cela, les paquets vidéo sont interdépendants. Si des paquets ne sont pas correctement reçus, un autre paquet reçu peut ne pas être décodable. On assiste alors à une propagation des pertes. L'efficacité de compression peut être réduite pour augmenter la robustesse en diminuant la quantité de prédiction (si les trames sont codées indépendamment, on parle de codage Intra par opposition au codage Inter lorsque les trames sont prédites les unes par rapport aux autres). Les dépendances seraient alors réduites, empêchant efficacement la propagation des pertes.

Les délais de transmission peuvent également permettre un compromis en robustesse. Les paquets vidéo peuvent être réorganisés de façon à ce que des paquets consécutifs ne représentent pas des données vidéo voisines. Ceci permet de délocaliser l'effet des pertes et faciliter ainsi le masquage d'erreurs. Un *burst* ou une *rafale* de paquets perdus n'affectera que des portions de la vidéo très éloignées les unes des autres. Les portions perdues peuvent être ensuite approximées en exploitant les données voisines.

Toutes ces techniques sont très efficaces, mais il est généralement très difficile de trouver un ensemble de paramètres optimal, surtout si de nombreux récepteurs expérimentent des conditions de canal différentes.

Le codage par descriptions multiples (codage MD) (en anglais Multiple Description Coding, MDC) est une autre technique de codage conjoint source-canal qui peut être vue comme un moyen d'améliorer la résistance aux erreurs sans utiliser des techniques complexes de codage de canal. Le principe du codage MD consiste à créer plusieurs descriptions corrélées du signal et les transmettre sur plusieurs canaux. Les descriptions peuvent avoir la même importance (on parle alors de schémas équilibrés) ou des importances différentes (schémas déséquilibrés). Au récepteur, plus le nombre de descriptions reçues est grand, plus la qualité du signal reconstruit est élevée. Contrairement aux algorithmes FEC, il n'existe pas de seuil au delà duquel la qualité décroît très fortement (l'effet de *cliff*). On parle alors de dégradation progressive. La robustesse vient du fait qu'il est très peu vraisemblable qu'une même portion d'une même information soit corrompue dans toutes les descriptions. L'efficacité de codage est réduite du fait de la présence de redondance dans les descriptions.

Nous commencerons ce chapitre par une présentation du problème de codage MD dans le cas de deux descriptions puis nous donnerons les limites théoriques pour ce problème ainsi que pour  $N$  descriptions. Dans une deuxième partie, nous présenterons les techniques de codage MD dans le domaine transformé, quantifié et au niveau du codage de canal. Puis, nous nous intéresserons aux propriétés des descriptions multiples pour une utilisation sur des canaux ou réseaux avant de terminer par un aperçu des applications du codage MD en vidéo.

## 1.2 Principes théoriques

C'est lors d'un atelier sur la théorie de Shannon en 1979 que la question suivante a été posée par Gersho, Ozarow, Witsenhausen, Wolf, Wyner et Ziv [1] : Si une source d'information est décrite par deux descriptions distinctes, quelles sont les limitations en qualité de ces descriptions prises séparément et ensemble ? Cette question a été à l'origine du problème de codage MD dans la communauté de la théorie de l'information. Les premiers résultats théoriques sont apparus dans les années 1980 [1, 2, 3].

---

### 1.2.1 Présentation du problème

Le problème de codage MD sera d'abord décrit avec deux descriptions. Le problème est illustré sur la figure 1.1. La source  $\{X_k\}_{k=1}^N$  est transmise sur deux canaux distincts. Les canaux ont deux états de fonctionnement possibles : soit ils sont en état de marche et transmettent sans erreur tous les symboles, soit ils sont défectueux et ne transmettent aucun symbole. Le récepteur connaît l'état des canaux mais pas le transmetteur qui envoie systématiquement deux descriptions sur les deux canaux.

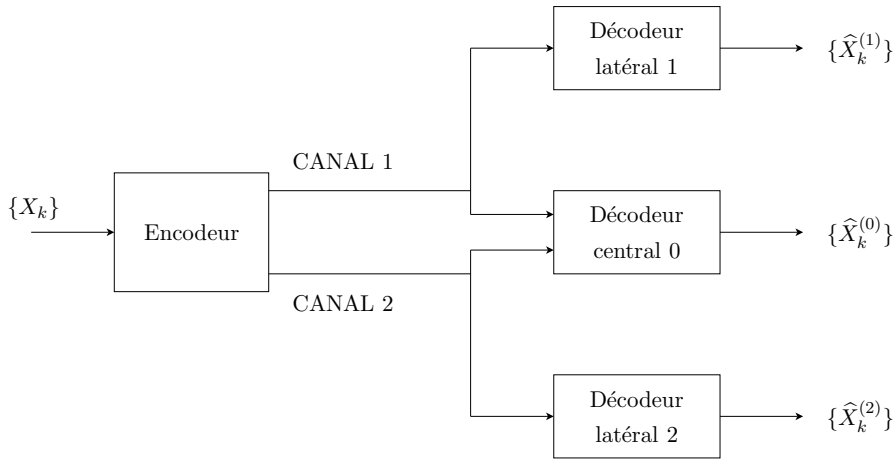


FIGURE 1.1 – Schéma générique de codage MD avec deux descriptions.

Au récepteur, les trois décodeurs correspondent à trois situations différentes. Le décodeur central est utilisé quand les deux canaux sont en état de fonctionnement, les deux décodeurs latéraux sont utilisés quand seul le canal qui leur est associé fonctionne. On suppose qu'il existe toujours au moins un canal en état de fonctionnement. Le débit du canal  $i$  est noté  $R_i$ . En notant  $\{\widehat{X}_k^{(i)}\}_{k=1}^N$  la séquence reconstruite par le décodeur  $i$ , les performances du système pour un débit donné sont caractérisées par trois distorsions : la distorsion centrale  $D_0$  qui est l'erreur de reconstruction au décodeur central, et les deux distorsions latérales,  $D_1$  et  $D_2$ , aux décodeurs latéraux. Ces distorsions s'expriment par :

$$D_i = \frac{1}{N} \sum_{k=1}^N E \left[ \delta_i(X_k, \widehat{X}_k^{(i)}) \right], \quad i \in \{0, 1, 2\} \quad (1.1)$$

où les  $\delta_i$  sont des mesures de distorsion, à valeurs réelles, non négatives, et éventuellement distinctes.

La région débit-distorsion pour la source  $\{X_k\}$  est définie comme étant l'ensemble des quintuplets  $(R_1, R_2, D_0, D_1, D_2)$  atteignables au sens de Shannon. Comme le décodeur 1 reçoit  $R_1$  bits, sa distorsion ne peut pas être inférieure à  $D(R_1)$  où  $D(\cdot)$  est la fonction débit-distorsion de la source. En utilisant la même notation pour les deux autres décodeurs, on obtient les inégalités suivantes :

$$D_0 \geq D(R_1 + R_2) \quad (1.2)$$

$$D_1 \geq D(R_1) \quad (1.3)$$

$$D_2 \geq D(R_2) \quad (1.4)$$

Une égalité parfaite pour (1.2), (1.3) et (1.4) signifierait un débit optimal  $R_1 + R_2$  pour la description centrale qui se diviserait en deux débits optimaux  $R_1$  et  $R_2$  pour les descriptions latérales. Malheureusement, il n'est généralement pas possible d'atteindre cet état idéal car des descriptions optimales individuellement sont souvent très similaires et donc redondantes lorsqu'elles sont combinées. Le compromis fondamental de ce type de problème réside dans la construction de descriptions suffisamment bonnes individuellement tout en étant suffisamment différentes.

### 1.2.2 Limites théoriques

C'est à l'occasion de l'atelier de septembre 1979 précédemment mentionné que Wyner présenta des résultats préliminaires obtenus avec Witsenhausen, Wolf, et Ziv pour une source binaire et une distorsion de Hamming. Au cours du même atelier, El Gamal et Cover ont déterminé la région des débits atteignables dans le cas général publiée plus tard dans [1]. La région débit-distorsion est totalement connue uniquement pour une source gaussienne sans mémoire et une mesure de distorsion quadratique. Ce résultat, obtenu par Ozarow [3], peut être résumé par le théorème suivant :

**Théorème 1** *Soit  $X(1), X(2) \dots$  une séquence de variables aléatoires gaussiennes indépendantes et identiquement distribuées (i.i.d.) où  $\mathbb{E}[X^2] = \sigma_X^2$ . L'ensemble des valeurs de débit et de distorsion atteignables pour une mesure de distorsion quadratique est l'ensemble des quintuplets  $(R_1, R_2, D_0, D_1, D_2)$  satisfaisant :*

$$D_i \geq \sigma_X^2 2^{-2R_i}, \quad i \in \{1, 2\} \quad (1.5)$$

$$D_0 \geq \sigma_X^2 2^{-2(R_1+R_2)} \gamma(R_1, R_2, D_1, D_2, \sigma_X^2) \quad (1.6)$$

où

$$\begin{aligned} \gamma &= \frac{1}{1 - \left( \left| \sqrt{\Pi} - \sqrt{\Delta} \right|^+ \right)^2}, \\ \Pi &= \left( 1 - \frac{D_1}{\sigma_X^2} \right) \left( 1 - \frac{D_2}{\sigma_X^2} \right), \\ \Delta &= \left( \frac{D_1}{\sigma_X^2} \right) \left( \frac{D_2}{\sigma_X^2} \right) - e^{-2(R_1+R_2)} \\ |x|^+ &\triangleq \begin{cases} x, & \text{si } x > 0 \\ 0, & \text{sinon.} \end{cases} \end{aligned} \quad (1.7)$$

Le signe  $|\cdot|^+$  dans (1.7) n'apparaît pas dans [3]; il devient opérant seulement si  $D_1 + D_2 > \sigma_X^2 [1 + 2^{-2(R_1+R_2)}]$ , c'est-à-dire pour des cas où au moins une des deux descriptions latérales est très mauvaise. Son omission n'a aucune conséquence sur les bornes de la région débit-distorsion.

Les inégalités (1.5) représentent simplement les limites débit-distorsion des deux canaux latéraux, elles correspondent aux inégalités (1.3) et (1.4). L'inégalité (1.6) nous indique que la distorsion centrale doit dépasser la limite débit-distorsion classique pour un débit  $R_1 + R_2$  d'un facteur  $\gamma$ .

Si les inégalités (1.5) sont des égalités, ce qui veut dire que les descriptions sont individuellement bonnes, nous pouvons écrire :

$$D_0 \geq \frac{D_1 D_2}{\sigma_X^2 [1 - (1 - D_1/\sigma_X^2)(1 - D_2/\sigma_X^2)]}. \quad (1.8)$$

La distorsion centrale  $D_0$  devient alors légèrement plus faible que la plus faible des deux distorsions latérales, la description centrale est donc particulièrement mauvaise. A l'inverse, si on minimise autant que possible la distorsion centrale, alors  $\gamma = 1$ , et

$$D_1 + D_2 > \sigma_X^2 \left[ 1 + 2^{-2(R_1+R_2)} \right]. \quad (1.9)$$

Rappelons qu'une valeur de distorsion de 1 est obtenue sans aucune information, en estimant simplement la source par sa moyenne. Donc, pour un débit très faible, l'inégalité (1.9) signifie qu'au moins un des décodeurs latéraux aura une reconstruction de très mauvaise qualité.

En plus de ces deux cas extrêmes, il est intéressant de s'attarder sur le cas *équilibré* où  $R_1 = R_2$  et  $D_1 = D_2$ . Nous prendrons deux exemples pour tenter de mieux comprendre la signification des bornes en fixant  $\sigma_X^2 = 1$  pour des raisons de simplicité. Le premier concerne le taux de décroissance exponentielle des trois distorsions et le deuxième, les limites théoriques pour la distorsion latérale en fonction de l'écart entre  $D_0$  et  $2^{-2(R_1+R_2)}$ .

Tout d'abord, estimons  $\gamma$  sachant  $R_1 = R_2 \gg 1$  et  $D_1 = D_2 \ll 1$ . Dans ce cas,  $\gamma \approx (4D_1)^{-1}$ , et en substituant  $\gamma$  dans (1.6), on trouve  $D_0 \geq 2^{-4R_1}(4D_1)^{-1}$ . Donc le produit des distorsions centrale et latérales est minoré approximativement par  $4^{-1}2^{-4R_1}$ . Si  $D_1 = D_2 \approx 2^{-2(1-\alpha)R_1}$  avec  $0 < \alpha \leq 1$ , alors la distorsion centrale la plus faible est :

$$D_0 \approx 4^{-1}2^{-2(1+\alpha)R_1}. \quad (1.10)$$

Ceci montre que si l'on veut accroître le taux de décroissance exponentielle de  $D_1$  (en augmentant  $\alpha$ ), on diminue d'autant le taux de décroissance de  $D_0$ .

Une deuxième manière d'interpréter le compromis entre les distorsions centrale et latérales nécessite de réarranger l'inégalité (1.6), sous la condition  $D_1 = D_2$ . On obtient alors

$$D_1 \geq \min \left\{ \frac{1}{2} \left[ 1 + D_0 - (1 - D_0) \sqrt{1 - 2^{-2(R_1+R_2)} D_0} \right], 1 - \sqrt{1 - 2^{-2(R_1+R_2)} D_0} \right\}. \quad (1.11)$$

Maintenant séparons le débit total en deux composantes : un débit de base  $r$  et une redondance  $\rho$ . Le débit de base  $r$  correspond au débit minimal pour obtenir une distorsion centrale  $D_0$ . Dans le cas d'une source gaussienne sans mémoire de variance  $\sigma_X^2 = 1$ ,  $r$  est défini par  $D_0 = 2^{-2r}$ . La redondance  $\rho$  est l'excès de débit global qu'il est nécessaire d'ajouter pour réduire les distorsions latérales :  $\rho = R_1 + R_2 - r$ . L'inégalité (1.11) devient :

$$D_1 \geq \begin{cases} \frac{1}{2} [1 + 2^{-2r} - (1 - 2^{-2r}) \sqrt{1 - 2^{-2\rho}}], & \text{si } \rho \leq r - 1 + \log_2(1 + 2^{-2r}) \\ 1 - \sqrt{1 - 2^{-2\rho}}, & \text{sinon.} \end{cases} \quad (1.12)$$

La seconde expression exprime ici le fait que si la redondance est suffisamment grande il n'y a plus de conflits entre la distorsion centrale et les distorsions latérales. Cette limite est illustrée sur la figure 1.2 pour différentes valeurs de  $r$ .

Il est aussi intéressant d'étudier la pente de la distorsion  $D_1$  en fonction de  $\rho$  :

$$\frac{\partial D_1}{\partial \rho} = -\frac{1 - 2^{-2r}}{2} \frac{2^{-2\rho} \ln 2}{\sqrt{1 - 2^{-2\rho}}}. \quad (1.13)$$

On remarque que lorsque  $\rho$  tend vers 0, la pente devient infinie. Cela signifie que si, par exemple, le système est conçu pour minimiser la distorsion centrale et que l'on cherche également à réduire les distorsions latérales, une légère augmentation du débit aura un

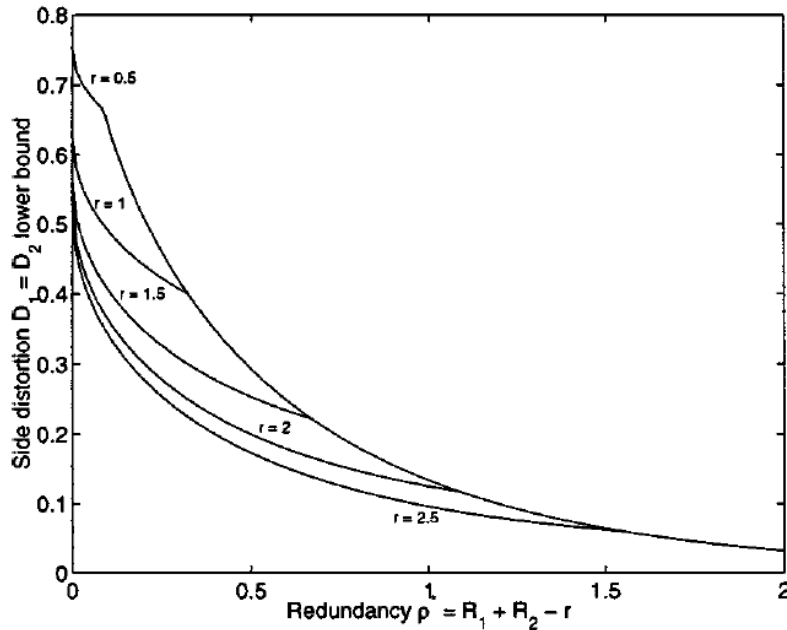


FIGURE 1.2 – Borne inférieure (1.12) des distorsions latérales  $D_1 = D_2$  lorsque le débit de base est  $r$ , c'est-à-dire lorsque la distorsion centrale est  $D_0 = 2^{-2r}$ . Pour des redondances élevées, les courbes se rejoignent car la borne est indépendante de  $r$  (Extrait de [4]).

plus grand impact si elle est dédiée à la réduction de la distorsion latérale plutôt qu'à une réduction de la distorsion centrale.

La région débit-distorsion pour des sources gaussiennes dans le cas équilibré est représentée sur la figure 1.3.

La région débit-distorsion n'est pas totalement connue pour une source non gaussienne. Zamir [6] a trouvé des bornes internes et externes de la région débit-distorsion pour n'importe quelle source sans mémoire de variables continues pour une mesure de distorsion quadratique. Il s'agit d'une extension des bornes de Shannon des fonctions débit-distorsion.

### 1.2.3 Codage à $N$ descriptions

Le problème de codage MD peut être généralisé à plus de deux canaux et plus de trois récepteurs. L'extension naturelle est  $N$  descriptions avec  $2^N - 1$  décodeurs. Cette généralisation a été étudiée par Witsenhausen [7] pour le cas où la source a un débit entropique fini et où une communication sans perte est requise quelque soit le nombre de descriptions reçues. En normalisant le débit de la source et en supposant une utilisation identique de chaque canal, chaque canal doit alors recevoir un débit de  $1/(N - k)$ , où  $k$  est le nombre de descriptions reçues. Cette limite est atteinte en utilisant des codes de Reed-Solomon tronqués. Des résultats similaires sont présentés dans un cadre plus général dans [8] et une situation avec trois canaux et sept décodeurs a été étudiée par Zhang et Berger dans [9].

Plus récemment, Venkataramani, Kramer et Goyal [10] ont proposé une généralisation des résultats obtenus par El Gamal et Cover [1] pour  $N = 2$  descriptions. Ils ont défini une borne externe de la région débit-distorsion pour une source gaussienne et une mesure de distorsion quadratique et ont montré que cette borne correspond à la région des débits



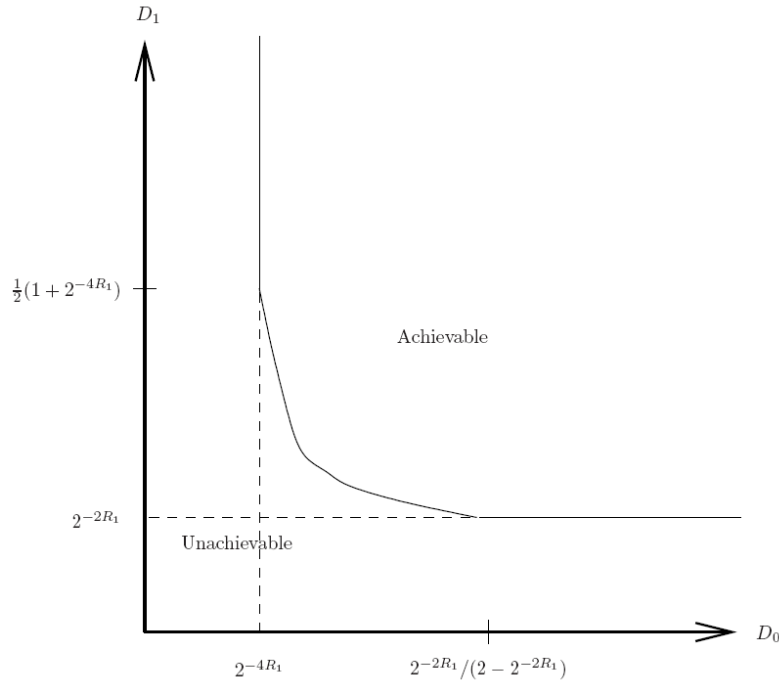


FIGURE 1.3 – Région débit-distorsion pour une source gaussienne sans mémoire avec  $D_1 = D_2$  et  $R_1 = R_2$  et pour une mesure de distorsion quadratique (Extrait de [5]).

atteignables seulement lorsque l'on s'intéresse aux performances du décodeur central et des  $N$  décodeurs à description simple (codage SD) (en anglais Single Description Coding, SDC) (ayant tous la même distorsion).

Puri, Pradhan et Ramchandran [11, 12] sont également parvenus à caractériser les performances du problème du codage à  $N$  descriptions. Pour ce faire, ils ont appliqué les résultats présentés dans [13] au problème de codage symétrique à  $N$  descriptions en utilisant la structure de codage proposée par Albanese *et al.* [14] et détaillée dans la section 1.5. Cette approche leur permet d'obtenir des tuples atteignables jusqu'alors inconnus dans la littérature. Les techniques utilisées pour trouver cette région de débit sont inspirées de celles utilisées dans la démonstration du théorème de Wyner-Ziv [15] et sont basées sur une philosophie de *codage avec information adjacente*. L'utilisation de ces principes est intéressante car elle contraste avec les approches basées sur une structure de *raffinement progressif* utilisées dans [1, 10].

Leur approche sera décrite pour  $N = 3$  descriptions mais est généralisable au cas  $N > 3$ . Soit  $X$  une source ayant pour densité de probabilité la fonction  $q(x)$ .  $X$  est segmentée en 3 couches  $Y_1$ ,  $Y_2$  et  $Y_3$ . Soient  $\mathcal{Y}_3$  l'alphabet de  $Y_3$  et  $\mathcal{Y}_{ij}$  l'alphabet de  $Y_{ij}$  où  $i \in I$ ,  $I = \{1, 2\}$  et  $j \in J$ ,  $J = \{1, 2, 3\}$ . Le débit total est partitionné en trois débits :  $R = R_1 + R_2 + R_3$ . La couche de base est codée par un code correcteur d'erreurs (3, 1) en trois variables  $Y_{11}$ ,  $Y_{12}$ ,  $Y_{13}$  pour un débit  $R_1$  comme le montre la figure 1.4. La réception de deux descriptions doit être suffisante pour décoder la deuxième couche. On fait l'hypothèse que la transmission de  $Y_2$  utilise le paradigme distribué. La réception de deux descriptions permet de recevoir une des paires  $(Y_{11}, Y_{12})$ ,  $(Y_{12}, Y_{13})$  ou  $(Y_{13}, Y_{11})$ . Chacune de ces paires peut être traitées comme information adjacente pour estimer  $X$ . Grâce au codage de source avec information adjacente, une seule couche de raffinement est alors nécessaire. C'est cette information qui

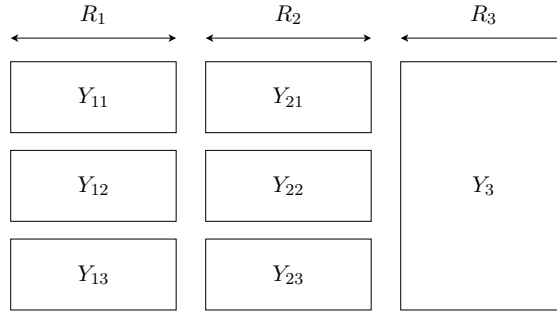


FIGURE 1.4 – Codage de canal par descriptions multiples utilisant une concaténation de codes correcteur  $(3, 1)$ ,  $(3, 2)$  et  $(3, 3)$ .

doit être décodable lors de la réception de 2 paquets. L'information adjacente pour le décodage de  $Y_{21}$  est donc constitué de  $Y_{22}$  et  $Y_{23}$ , mais également de  $Y_{12}$ ,  $Y_{13}$  et  $Y_{11}$ . La dernière couche,  $Y_3$ , qu'il est possible de décoder seulement si tous les paquets ont été reçus, peut être envoyée à un débit  $R_3$  par paquet en utilisant une approche de raffinement progressif. La région débit-distorsion pour ce problème est définie par le théorème suivant [11] :

**Théorème 2** Une région de débits atteignables pour un triplet de distorsions  $(D_1, D_2, D_3)$  est donnée par  $R > R_1 + R_2 + R_3$ , où,

$$R_1 > H(Y_{11}) - \frac{1}{3}H(Y_{1J}|X), \quad R_2 > \frac{1}{2}H(Y_{2I}|Y_{1I}) - \frac{1}{3}H(Y_{2J}|X) \quad (1.14)$$

$$R_3 > \frac{1}{3}I(X; Y_3|Y_{IJ}) \quad (1.15)$$

pour une fonction densité de probabilité  $p(x, y_{IJ}, y_3) = q(x)p(y_{1J}|x)p(y_{2J}|x)p(y_3|x, y_{IJ})$ , où  $Y_{1J} \rightarrow X \rightarrow Y_{2J}$  est une chaîne de Markov et un ensemble de fonctions de décodage

$$g_i^1 : \mathcal{Y}_{1i} \rightarrow \hat{\mathcal{X}}, \quad i \in J, \quad (1.16)$$

$$g_S^2 : \bigotimes_{k \in I, i \in S} \mathcal{Y}_{ki} \rightarrow \hat{\mathcal{X}} \quad \forall S \subset J, |S| = 2 \quad (1.17)$$

$$g^3 : \mathcal{Y}_3 \times \bigotimes_{k \in I, i \in J} \mathcal{Y}_{kj} \rightarrow \hat{\mathcal{X}} \quad (1.18)$$

telles que

$$\mathbb{E}[d(X, g_i^1(Y_{1i}))] \leq D_1, \quad \forall i \in J \quad (1.19)$$

$$\mathbb{E}[d(X, g_S^2(Y_{I,S}))] \leq D_2, \quad \forall S \subset J, |S| = 2 \quad (1.20)$$

$$\mathbb{E}[d(X, g^3(Y_{I,J}, Y_3))] \leq D_3 \quad (1.21)$$

On pourra se référer à [16, 17] pour une analyse plus détaillée de ces résultats.

Récemment, le codage MD a connu un regain d'intérêt et de nombreuses techniques mettant en œuvre ce principe ont été proposées. Nous allons présenter dans les sections suivantes un certain nombre d'entre elles.

## 1.3 Descriptions multiples basées sur la quantification

### 1.3.1 Quantification scalaire à descriptions multiples

Les premiers résultats en codage MD ont été obtenus par la technique de quantification scalaire à descriptions multiples (en anglais Multiple Description Scalar Quantization, MDSQ) introduite par Vaishampayan dans [18]. La MDSQ est utilisée essentiellement pour le codage de deux descriptions et est constituée de deux parties : un quantificateur scalaire (au sens classique), et une *assignation d'index* qui sépare l'information de chaque échantillon quantifié en deux descriptions complémentaires du même échantillon. Un exemple est illustré sur la figure 1.5.

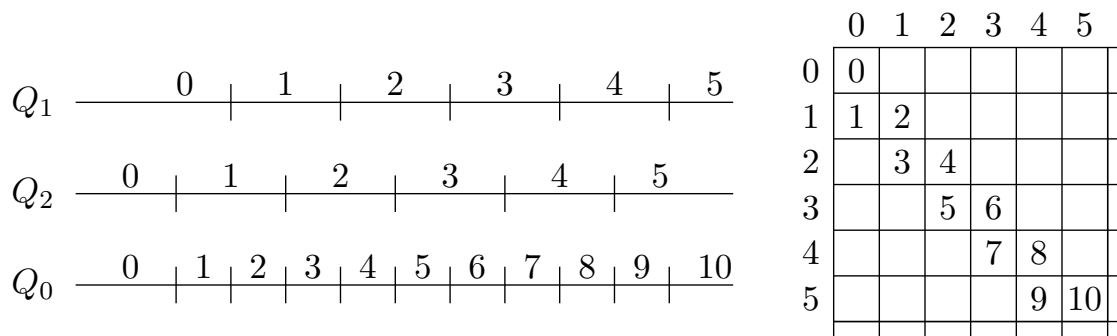


FIGURE 1.5 – Exemple de MDSQ avec des cellules entrelacées. Représentation *naturelle* des cellules (gauche) et représentation matricielle (droite).

La sortie de chaque quantificateur est un indice qui est utilisé pour estimer la source. Notons alors  $Q_0 : \mathbb{R} \rightarrow \{0, 1, \dots, 10\}$ , la carte d'encodage du quantificateur 0 et  $Q_i : \mathbb{R} \rightarrow \{0, 1, \dots, 5\}, i = 1, 2$ , la carte d'encodage du quantificateur  $i$ . Au niveau des décodeurs latéraux, la reconstruction de  $Q_i(x) = k_i$  est le centroïde de la cellule  $Q_i^{-1}(k_i)$ . Le décodeur central reçoit quant à lui les deux informations,  $Q_1(x) = k_1$  et  $Q_2(x) = k_2$ , et choisit alors pour la reconstruction le centroïde de la cellule d'intersection  $Q_1^{-1}(k_1) \cap Q_2^{-1}(k_2)$ . Dans l'exemple, les cellules d'intersection sont deux fois plus petites que celles des quantificateurs individuels, la distorsion centrale est alors quatre fois plus petite que la distorsion latérale. Asymptotiquement, si les débits latéraux sont égaux  $R_1 = R_2 = R$ , alors les trois distorsions  $D_0, D_1$  et  $D_2$  sont toutes de l'ordre de  $O(2^{-2R})$ . Cette solution est optimale pour  $D_1$  et  $D_2$ , mais est loin d'être optimale pour  $D_0$ .

Nous savons à partir du théorème 1 qu'il est possible de diminuer la distorsion  $D_0$  en augmentant la distorsion  $D_1$  et/ou la distorsion  $D_2$ . Soit  $n_i, i = 1, 2$ , le nombre de cellules du quantificateur  $Q_i$ , et  $n_0$  le nombre d'intersections non vides entre les cellules de  $Q_1$  et de  $Q_2$ . Nous pouvons observer que dans le cas de la figure 1.5, nous avons  $n_0 = n_1 + n_2 - 1$ . Quand  $n_1 = n_2 = n$ , le taux de décroissance de  $D_0$  change seulement si  $n_0$  croît plus que linéairement en fonction de  $n$ , et cela s'obtient avec des cellules formées d'intervalles disjoints. Le nombre maximum de cellules pour le décodeur central est  $n_0 = n_1 \times n_2$ , ce qui est obtenu quand  $Q_1^{-1}(k_1) \cap Q_2^{-1}(k_2)$  est non vide quelque soient  $k_1$  et  $k_2$ . Un exemple de ce cas est illustré sur la figure 1.6. Le quantificateur  $Q_2$  est alors individuellement de qualité médiocre. Les performances asymptotiques de ce schéma quand  $R_1 = R_2 = R$  sont à l'opposé de l'exemple précédent :  $D_0 = O(2^{-4R})$  mais au moins une des deux distorsions  $D_1$  et  $D_2$  est de l'ordre  $O(1)$ .

Si la distorsion centrale  $D_0$  dépend essentiellement du nombre de cellules, la distorsion

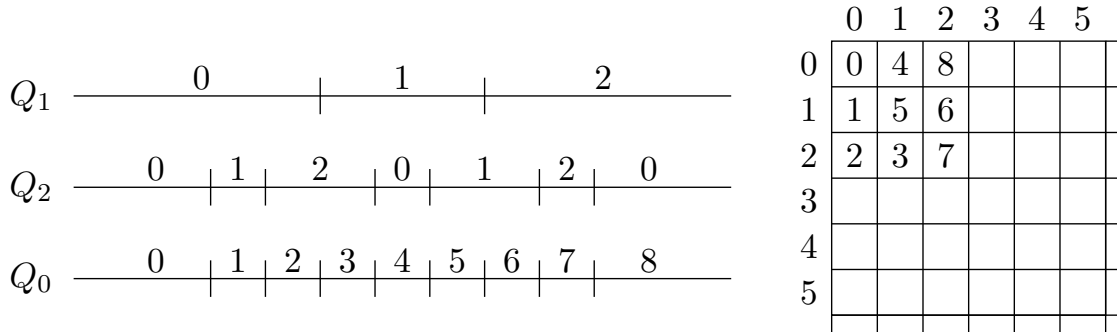


FIGURE 1.6 – Exemple de MDSQ qui minimise  $D_0$  à  $R$  fixé. Représentation *naturelle* des cellules (gauche) et représentation matricielle (droite)

latérale dépend surtout de l'indexation de chaque quantificateur : selon cette indexation, l'étalement des cellules peut beaucoup varier. Vaishampayan donne alors dans [18] un algorithme pour remplir les tables d'index de façon à obtenir des résultats optimaux pour les distorsions centrales et latérales. Un exemple d'indexation entrelacée donnée par Vaishampayan est illustré sur la figure 1.7. Par rapport au cas de la figure 1.6, les distorsions latérales sont approximativement égales, et pour un nombre  $n$  donné de cellules pour les quantificateurs  $Q_i$ , la distorsion centrale est plus petite (au prix d'une distorsion latérale plus grande) que dans le schéma de MDSQ décrit dans la figure 1.5. Par la suite, Berger-Wolf et Reingold [19] ont généralisé la construction de matrices d'assignation d'index pour  $N$  descriptions.

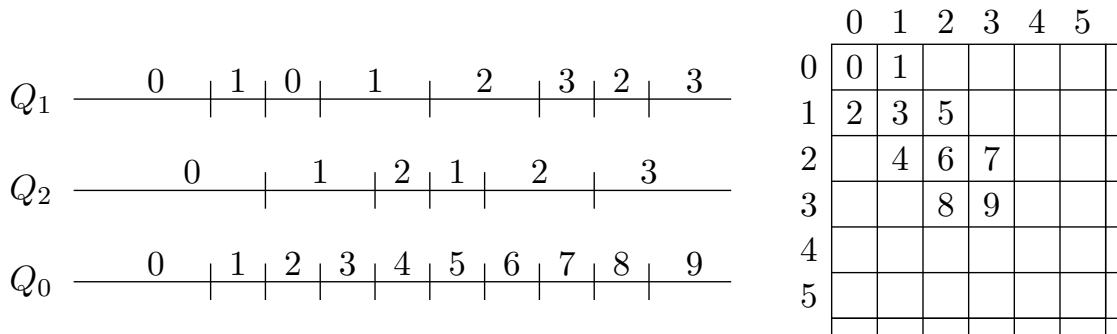


FIGURE 1.7 – Exemple de MDSQ basée sur une indexation entrelacée modifiée décrite dans [18]. Cette indexation permet d'obtenir un compromis entre les distorsions centrale et latérale :  $D_0$  est plus faible que dans le cas décrit par la figure 1.5, mais  $D_1$  et  $D_5$  sont plus élevées. Représentation *naturelle* des cellules (gauche) et représentation matricielle (droite).

Jusqu'à présent, nous avons supposé que les indices produits par les quantificateurs étaient transmis avec un code de longueur fixe. Dans le but d'améliorer les performances du système, on peut malgré tout souhaiter utiliser un code de longueur variable, tel qu'un code de Huffman ou un code arithmétique. Un tel codeur est appelé quantificateur scalaire à descriptions multiples avec codage entropique (entropy-constrained multiple description scalar quantizer ECMDSQ) et nous pouvons trouver un exemple d'étude sur ce sujet dans [20].

### 1.3.2 Quantification vectorielle à descriptions multiples

Nous avons vu la construction de descriptions avec une quantification scalaire. Il est également possible de créer des descriptions avec une quantification vectorielle [21]. Ce problème, appelé quantification vectorielle sur réseau de points régulier à descriptions multiples (en anglais Multiple Description Lattice Vector Quantization, MDLVQ) a été particulièrement étudié pour deux descriptions équilibrées par Servetto, Vaishampayan et Sloan [22, 23]. Il revient à assigner des paires d'indices aux différents points d'un dictionnaire de quantificateur vectoriel.

La source  $\{X_n\}$  est divisée en vecteurs de longueur  $L$ . Les vecteurs sont ensuite quantifiés par un treillis fin  $\Lambda$ . A chaque point sur  $\Lambda$  est assigné une unique paire de points  $(\Lambda'_1, \Lambda'_2) \in \Lambda' \times \Lambda'$  par une fonction d'indexation vectorielle  $l : \Lambda \rightarrow \Lambda' \times \Lambda'$ .  $\Lambda' \times \Lambda'$  est un sous-treillis de  $\Lambda$ , correspondant donc à une quantification plus grossière. Pour des raisons de simplicité,  $\Lambda'$  est généralement similaire à  $\Lambda$  à une rotation et à une homothétie près. Toute la difficulté de la MDLVQ consiste à trouver la fonction d'indexation qui donne des résultats optimaux pour un treillis donné. Dans [23], les auteurs proposent d'exploiter les symétries des réseaux de points pour construire des sous-treillis optimaux.

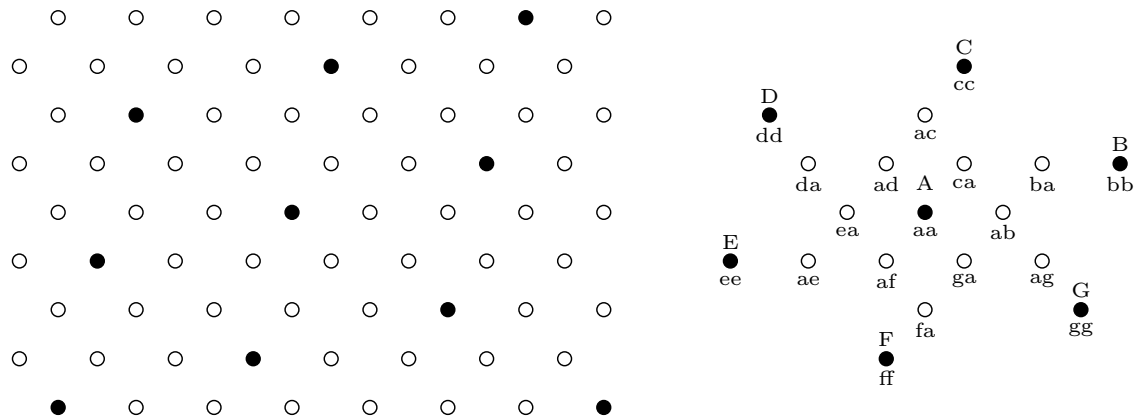


FIGURE 1.8 – Exemple de treillis pour la MDLVQ : le treillis hexagonal (gauche), et l'indexation optimale associée (droite).

Considérons par exemple le treillis hexagonal représenté sur la figure 1.8 (droite). Tous les points appartiennent au treillis  $\Lambda$  mais seuls les points noirs appartiennent au treillis  $\Lambda'$ . Si par exemple la paire codée est le point  $ab$ , le symbole reconstruit au premier décodeur latéral sera le point  $A$ , et celui du deuxième décodeur sera le point  $B$ .

## 1.4 Descriptions multiples basées sur la transformation

Les techniques de codage de source se servent de transformations orthogonales tels que la transformée en cosinus discrète (Discrete Cosine Transform, DCT) ou la transformée en ondelettes discrète (Discrete Wavelet Transform, DWT) pour décorréler le signal et en obtenir une représentation compacte qui supporte la suppression d'une partie de l'information au prix d'une perte de qualité acceptable lors de la reconstruction. Cette opération contribue à diminuer la robustesse sur des canaux bruités, car en effet, si un symbole est perdu, les symboles voisins décorrélés ne permettront pas d'en faire une estimation correcte.

Deux approches sont possibles pour introduire de la corrélation dans le domaine transformé. La première consiste à remplacer la transformation classique par une transformation qui permet à la fois de générer des descriptions et de contrôler leur niveau de redondance. Par exemple, la DCT peut être remplacée par une transformée orthogonale recouvrante (en anglais Lapped Orthogonal Transform, LOT) [24] suivie d'une répartition des blocs dans des descriptions [25]. De son côté, la DWT peut être remplacée par des bancs de filtres multi-ondelettes [26, 27] dont les sorties sont les descriptions [28]. La deuxième approche consiste à ajouter une transformation linéaire entre la transformation classique et la quantification de façon à introduire de la corrélation entre les symboles indépendants. Dans cette section, nous présentons les méthodes reposant sur cette dernière approche.

### 1.4.1 Appariement de doublets

Cette technique, connue sous le nom de «Multiple Description Transform Coding» (MDTC), a été introduite par Wang, Orchard, Reibman et Vaishampayan [29, 30] pour le cas de deux variables et a été généralisée par Goyal et Kovacevic pour  $n$  variables et une plus grande classe de transformées [31], et en introduisant l'utilisation de la théorie des expansions sur bases de fonctions redondantes [32]. Ils considèrent le cas de deux variables  $x = (x_1, x_2)$  indépendantes ayant comme variance  $\sigma_1^2$  et  $\sigma_2^2$ , respectivement. Le principe du MDTC consiste à transformer linéairement ces deux variables non corrélées en deux variables  $y_1$  et  $y_2$  corrélées. Les paramètres de la transformée contrôlent la corrélation entre  $y_1$  et  $y_2$  qui, à son tour, contrôle la redondance du codeur MDTC. Le schéma de codage de base est illustré sur la figure 1.9.

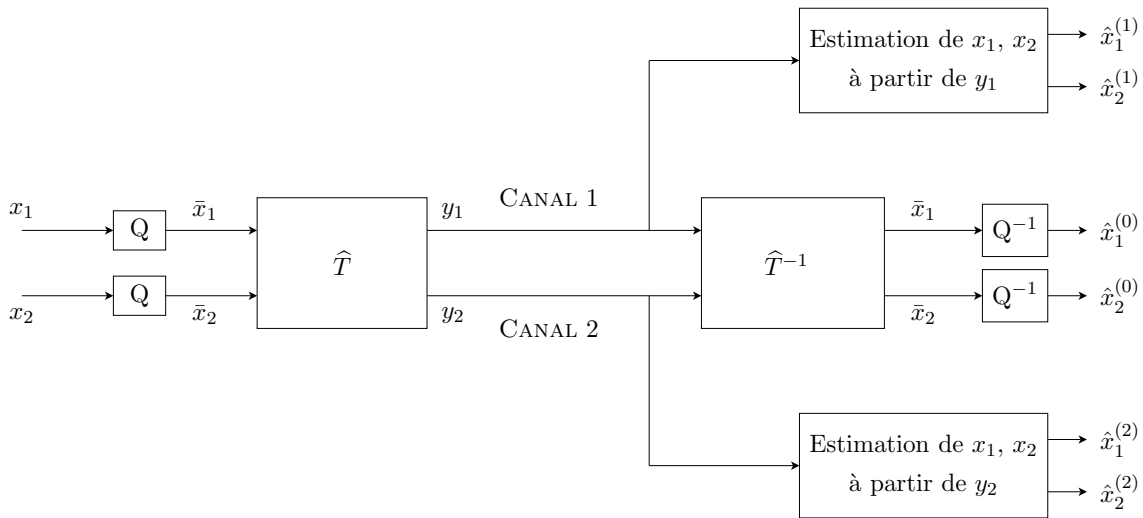


FIGURE 1.9 – Schéma MDTC pour une paire de variables gaussiennes indépendantes.

Comme la transformée est, en général, non orthogonale, la quantification de  $y_1$  et  $y_2$  donnerait de très mauvaises performances. C'est pourquoi,  $x_1$  et  $x_2$  sont d'abord quantifiées en  $\bar{x}_1$  et  $\bar{x}_2$  avant d'être transformées par  $\hat{T}$ , une version discrète d'une transformée linéaire continue et inversible  $T$ . On peut montrer que si  $T$  est une transformée linéaire continue de déterminant 1, elle est inversible. La première étape pour rendre discrète une transformée linéaire consiste à la factoriser en un produit de matrices triangulaires inférieures et

supérieures à diagonale unité  $T = T_1 T_2 \dots T_k$ . La version discrète est alors donnée par

$$\hat{T}(x_q) = [T_1 [T_2 \dots [T_k x_q]_{\Delta}]_{\Delta}]_{\Delta}, \quad (1.22)$$

où  $\Delta$  est le pas du quantificateur utilisé par le codeur. Une structure *lifting* peut alors être implémentée en inversant les calculs de (1.22) :

$$\hat{T}^{-1}(y) = [T_k^{-1} \dots [T_2^{-1} [T_1^{-1} y]_{\Delta}]_{\Delta}]_{\Delta}. \quad (1.23)$$

Au récepteur, si les deux séquences  $y_1$  et  $y_2$  sont disponibles, la transformée discrète est inversée pour obtenir  $\bar{x}_1$  et  $\bar{x}_2$  et une quantification inverse est appliquée pour obtenir  $\hat{x}_1^{(0)}$  et  $\hat{x}_2^{(0)}$ . La qualité de la reconstruction ne dépend alors que de la quantification initiale. Dans le cas où une seule description est reçue, par exemple  $y_1$ ,  $y_2$  est estimé à partir des composants reçus en utilisant la corrélation statistique introduite par la transformée  $\hat{T}$ . Les valeurs estimées  $\hat{x}_1^{(1)}$  et  $\hat{x}_2^{(1)}$  sont alors générées par inversion de  $T$ . L'estimé de  $x$  qui minimise l'erreur quadratique est donné par  $E[x|y_1]$  :

$$\hat{x}^{(1)} = E[x|y_1] = E[T^{-1} T x | y_1] = T^{-1} E[T x | y_1] \quad (1.24)$$

$$= T^{-1} \begin{bmatrix} y_1 \\ E[y_2 | y_1] \end{bmatrix}. \quad (1.25)$$

Goyal et Kovacevic [5] ont montré que si la probabilité de défaillance est la même pour les deux canaux, la transformée optimale est de la forme :

$$T = \begin{bmatrix} \alpha & 1/(2\alpha) \\ -\alpha & 1/(2\alpha) \end{bmatrix}, \quad (1.26)$$

où  $\alpha$  est un réel non nul permettant de contrôler la redondance.

#### 1.4.2 Descriptions multiples basées sur une décomposition en sous-bandes

Cette technique a été proposée simultanément par Dragotti, Servetto et Vetterli [33, 34] et Yang et Ramchandran [35, 36] dans le cas de bancs de filtres à deux canaux. Elle repose sur l'utilisation d'une décomposition en sous-bandes, ou d'une DWT souvent utilisée dans les systèmes de codage d'image ou de vidéo.

Considérons par exemple le problème de codage illustrée sur la figure 1.10. L'utilisation de bancs de filtres permet d'introduire de la corrélation entre les deux descriptions  $y_1[n]$  et  $y_2[n]$ . L'approche proposée dans [35, 36] pour synthétiser les filtres consiste à minimiser la distorsion obtenue avec une seule description pour un niveau de redondance donnée. L'optimisation repose sur une formulation Lagrangienne qui est résolue directement dans le domaine fréquentiel pour produire les réponses des filtres. Si un canal est défaillant, par exemple le canal 1, les symboles manquants  $y_1[n]$  sont estimés linéairement (par un filtre de Wiener) à partir des symboles présents au décodeur  $y_2[n]$ . Les auteurs remarquent que pour une redondance nulle, leur solution rejoint le cas d'une optimisation de gain de codage classique. En revanche, pour une redondance maximale, leur optimisation aboutit à une transformée polyphase triviale. Ce système est en réalité une généralisation de la technique de descriptions multiples par transformée (MDCT), exposée dans la section 1.4.1, où, contrairement à cette dernière, la taille des filtres n'est pas limitée, et l'hypothèse faite sur la source est moins contraignante.

Le codage par sous-bandes est peu utilisé en pratique pour produire des descriptions multiples car il nécessite de synthétiser un banc de filtres différent pour chaque niveau de redondance, ce qui limite sa capacité d'adaptation à des conditions de transmission variables.

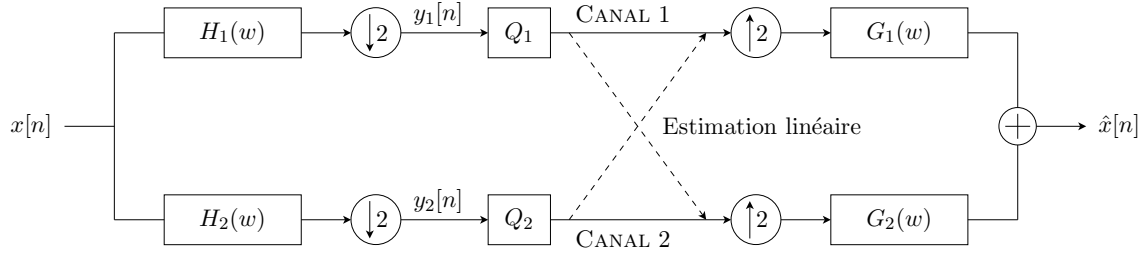


FIGURE 1.10 – Schéma de codage MD basé sur une décomposition en sous-bandes. Les traits pointillés représentent une estimation linéaire.

### 1.4.3 Descriptions multiples basées sur les trames

Les trames peuvent aussi être utilisées pour augmenter la robustesse sur des réseaux avec pertes [37]. Les trames ajoutent de la redondance dans le message à travers une expansion sur base de fonctions redondantes (les trames). Le récepteur peut alors reconstruire le message transmis sur un réseau sensible aux erreurs avec une précision suffisante si les opérateurs de la trame associés aux coefficients reçus possèdent des propriétés idoines.

La théorie des trames a été développée par Duffin et Schaeffer [38]. Une trame est une famille de vecteurs  $\{\phi_k\}$  génératrice dans un espace hilbertien [39] : contrairement à la décomposition sur une base, la décomposition d'un vecteur sur une trame est redondante. Cette redondance est déterministe puisque, si  $K$  est la dimension de l'espace,  $K$  coefficients  $\langle \mathbf{x}, \phi_k \rangle$  sont suffisants pour déterminer  $\mathbf{x}$ . Les coefficients supplémentaires sont alors de la pure redondance et n'apportent aucune information.

Soit  $\mathbb{C}^K$  un espace complexe à dimension  $K$ . La famille de vecteurs à  $K$  dimension  $\Phi_F \equiv \{\phi_k\}_{k=1}^N$  est une trame s'il existe  $A$  et  $B$  tels que  $0 < A \leq B < \infty$  et

$$A\|\mathbf{x}\|^2 \leq \sum_{k=1}^N |\langle \mathbf{x}, \phi_k \rangle|^2 \leq B\|\mathbf{x}\|^2, \quad \text{pour tout } \mathbf{x} \in \mathbb{C}^K \quad (1.27)$$

où l'opérateur de trame  $(F\mathbf{x})_k \equiv \langle \mathbf{x}, \phi_k \rangle$  est le produit scalaire de  $\mathbf{x}$  et  $\phi_k$ , et  $\|\mathbf{x}\|$  est la norme euclidienne de  $\mathbf{x}$ .  $A$  et  $B$  sont appelées les bornes inférieure et supérieure de la trame.  $(F\mathbf{x})_k$  correspond au  $k$ -ème coefficient de l'expansion sur la trame de  $\mathbf{x}$ . Si l'ensemble  $\{\phi_k\}$  est une trame de  $\mathbb{C}^K$ , alors il est un sous-espace vectoriel de  $\mathbb{C}^K$ . Autrement, il existerait toujours au moins un  $\mathbf{x} \in \mathbb{C}^K$  qui serait orthogonal à tous les  $\phi_k$ . Ainsi, une trame aura toujours  $N \geq K$ . Le ratio  $N/K$  correspond à la redondance de la trame.

Si  $\{\phi_k\}$  est une trame,  $F$  est inversible sur son image, avec une inverse bornée. La pseudo-inverse de l'opérateur de trame  $F$  vérifie alors l'équation :

$$\tilde{F}^{-1} = (F^*F)^{-1}F^* \quad (1.28)$$

où  $F^*$  est l'opérateur adjoint de l'opérateur  $F$ . La famille de vecteurs  $\{\tilde{\phi}_k\}_{k=1}^N$  est alors définie par :

$$\tilde{\phi}_k = (F^*F)^{-1}\phi_k \quad (1.29)$$

Cette famille de vecteurs  $\Phi_{\tilde{F}} \equiv \{\tilde{\phi}_k\}_{k=1}^N$  est aussi une trame, appelée trame duale de  $\Phi_F$ , et d'opérateur  $\tilde{F} = F(F^*F)^{-1}$ .



Connaissant les coefficients d'expansion sur la trame de n'importe quel vecteur  $\mathbf{x}$ , le vecteur  $\mathbf{x}$  peut être reconstruit en utilisant l'opérateur de la trame duale :

$$\mathbf{x} = \tilde{F}^*(F\mathbf{x}) = \sum_{k=1}^N \langle \mathbf{x}, \phi_k \rangle \tilde{\phi}_k = \sum_{k=1}^N \langle \mathbf{x}, \tilde{\phi}_k \rangle \phi_k \quad (1.30)$$

Le transposé conjugué de l'opérateur de la trame duale minimise l'erreur de reconstruction [37]. La sensibilité d'une trame au bruit dépend de l'écart entre ses bornes : des petites perturbations sur les coefficients peuvent entraîner d'importantes erreurs de reconstruction si les bornes sont très éloignées.

Une trame est dite uniforme si les vecteurs  $\phi_k$  ont la même norme, c'est-à-dire  $\|\phi_k\| = c$ ,  $\forall k = 1, 2, \dots, N$ . Une trame est dite normalisée si  $A = B = 1$ . Une trame est dite serrée (en anglais, tight) si ses bornes sont égales ( $A = B$ ).  $\Phi_F$  est serrée si et seulement si  $F^*F = AI_K$ , où  $I_K$  est une matrice identité d'ordre  $K$ . Cela implique que les colonnes de  $F$  sont orthogonales. Parmi les trames, les trames serrées et uniformes (en anglais, uniform tight frames) sont les plus importantes à cause de leurs bonnes propriétés : le calcul de la trame se fait simplement par  $\tilde{\phi}_k = A^{-1}\phi_k$  et la reconstruction est très rapide.

La théorie des trames constitue une base théorique adaptée à une application au codage MD. Dans [32], Goyal *et al.* proposent d'utiliser des transformées linéaires de  $\mathbb{C}^n$  vers  $\mathbb{C}^m$  pour décrire une source avec  $m$  paquets de telle sorte que la réception de n'importe quel sous-ensemble de paquets soit utilisable. Il s'agit d'une extension du problème posé par la MDTC qui se limite au cas où  $m = n$ .

Ils proposent de coder la source  $\mathbf{x} \in \mathbb{C}^n$  par  $Q(F(\cdot))$ , où  $Q$  représente une quantification scalaire et  $F$  est l'opérateur associé à une trame serrée et uniforme  $\Phi_F \equiv \{\phi_k\}_{k=1}^m \subset \mathbb{C}^n$ . Les  $m$  coefficients quantifiés de  $\hat{\mathbf{y}} = Q(F(\mathbf{x}))$  sont transmis séparément dans des descriptions.

Soit  $\mathcal{E}$  l'ensemble des indices des effacements, autrement dit supposons que les projections  $\{\langle \mathbf{x}, \phi_k \rangle\}_{k \in \mathcal{E}}$  sont effacées. Si  $\Phi'_{F'} = \Phi_F / \{\phi_k\}_{k \in \mathcal{E}}$  est une trame, le calcul de l'estimé  $\hat{x}$  de  $x$  qui minimise l'erreur quadratique peut seulement se faire sur un sous-espace et la connaissance de la distribution est nécessaire pour obtenir une bonne estimation. La quantification est modélisée par un bruit gaussien additif  $\eta$  de variables i.i.d. indépendantes de  $x$  et de variance  $\sigma^2$  tel que  $\eta = \hat{y} - Fx$ . Lorsqu'il n'y a pas d'erreur de transmission, l'erreur quadratique moyenne (EQM) dépend seulement du bruit de quantification et est donnée par  $n^2\sigma^2/m$ .

Dans cet exemple, l'utilisation des trames peut s'apparenter à une méthode de codage de canal. Contrairement au codage SD, les symboles ajoutés par l'opérateur de trame ne sont pas que de la pure redondance, mais permettent de limiter le bruit de quantification [40, 41].

## 1.5 Descriptions multiples basées sur les codes correcteurs d'erreurs

Les méthodes vues jusqu'à présent introduisaient de la redondance au niveau du codage de source. La méthode présentée dans [14] et [42] propose d'utiliser des techniques de codage de canal pour ajouter de la redondance dans le signal à transmettre. On suppose pour cela que le flux d'information à coder est organisé de façon hiérarchique et segmenté en couches d'importances décroissantes. Nous pouvons alors protéger ces couches par des codes correcteurs d'erreurs de redondance également décroissante. Cette idée est connue sous le nom de protection inégale contre les erreurs (en anglais Unequal Error Protection, UEP).

Cette technique peut être directement décrite dans le cas général de  $N$  descriptions. L'application visée est la transmission par paquets sur un réseau où les données à transmettre sont représentées par un flux binaire hiérarchique progressif. Le mécanisme de paquetisation du flux est illustré sur la figure 1.11. La première étape consiste à segmenter le flux hiérarchique en  $N$  couches de résolution croissante. La  $i$ -ème couche doit être décodable lorsque  $i$  descriptions parviennent au décodeur, et ce quelles que soient les descriptions reçues. Pour parvenir à cela, on divise la  $i$ -ème couche en  $i$  parties égales, que l'on place dans les  $i$  premiers paquets. Pour compléter les autres paquets, on utilise des codes correcteurs d'erreurs en bloc (FEC) de paramètres  $(N, i, N - i + 1)$ . Ces codes transforment  $i$  bits source en  $N$  bits codés par l'ajout de  $N - i$  bits de redondance. Cette redondance permet de corriger  $N - i$  bits erronés.

Ces blocs sont appliqués verticalement aux  $i$  tronçons de la  $i$ -ème couche (voir figure 1.11). Grâce à ce mécanisme, chaque description contient une partie de chacune des  $N$  couches du flux initial, avec la propriété que la  $i$ -ème couche peut être récupérée à partir de  $i$  paquets quelconques.

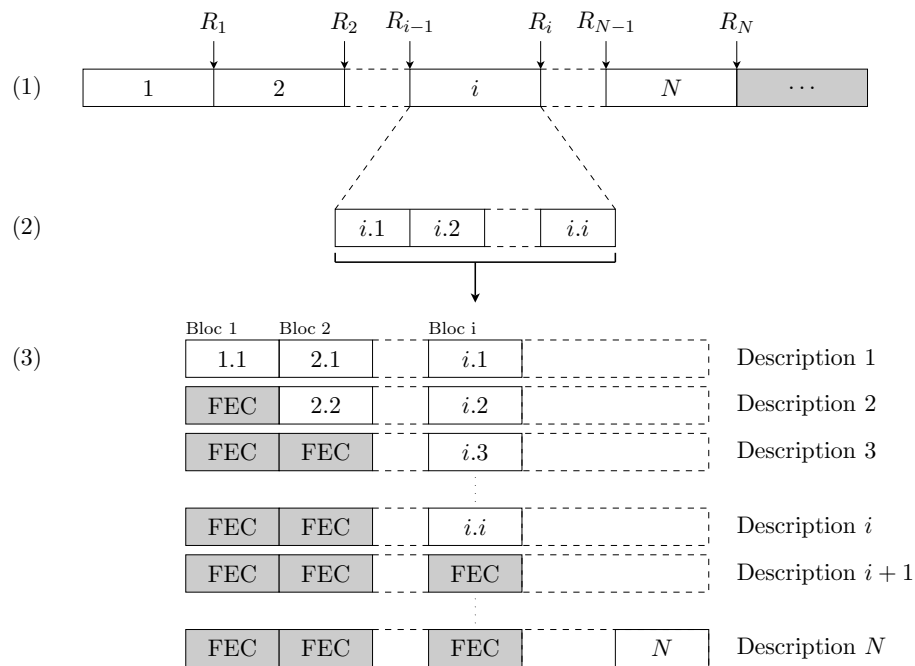


FIGURE 1.11 – Schéma de paquetisation d'un flux hiérarchique binaire. (1) Le flux est segmenté en  $N$  couches successives.  $R_i$  est le nombre de bits des  $i$  premières couches. (2) la  $i$ -ème couche est divisée en  $i$  parties égales. (3) Un code en bloc (FEC) de redondance  $(N, i)$  est appliqué à la  $i$ -ème couche, qui est ensuite répartie dans les  $N$  paquets.

Le codage de source et le codage de canal étant faits séparément, il est nécessaire d'utiliser des techniques d'allocation de débit entre les deux types de codage. La complexité de cette tâche étant très élevée, deux méthodes permettant d'obtenir une partition optimale du flux minimisant la distorsion moyenne au niveau du récepteur sont décrites dans [43, 44] et [45]. Un des avantages de l'UEP est qu'elle peut être utilisée en conjonction avec n'importe quel codeur de source scalable.

## 1.6 Réseaux et descriptions multiples

Les travaux fondateurs du codage MD se basent sur un modèle «on-off» pour les canaux ou les réseaux, sans imposer de contraintes sur les délais de transmission. Plus précisément, les systèmes de codage sont conçus en considérant que chacune des descriptions est soit complètement disponible sans erreur au récepteur, soit complètement perdue. De plus, la probabilité qu'une description soit perdue est indépendante du débit choisi pour coder la source. Dans ces conditions, il n'est pas étonnant de voir le codage MD surpasser le codage SD. Ces conditions sont très différentes de celles que l'on trouve en pratique sur les canaux ou les réseaux. Par exemple, dans une situation où une communication en temps réel est requise, les descriptions pourraient être partitionnées en paquets, chacun pouvant être reçu ou perdu individuellement. De plus, la présence de congestion dépend souvent du débit de transmission : plus le débit augmente et plus le risque de congestion augmente. C'est pourquoi il est important d'étudier le codage MD sur des modèles de canaux concrets.

Coward, Knopp et Servetto [46] ont examiné les performances du codage MD pour différents modèles de canal, parmi lesquels une paire de canaux à erreur-symbole, une paire de canaux sans mémoire à erreur-symbole et une paire de canaux à évanouissements par bloc. Pour des contraintes de délai très strictes, ils ont montré que le codage MD est plus efficace que le codage SD ; pour des contraintes de délai plus souples, permettant de mettre en œuvre des techniques de codage de canal plus sophistiquées, le codage SD s'est avéré plus performant. Ce résultat montre qu'il est important de prendre en compte la contrainte de délai au moment du choix de la technique de protection face aux erreurs. Laneman *et al.* [47] ont réalisé une étude similaire en considérant une plus grande variété d'options pour l'encodage et le décodage de canal. Ils considèrent deux classes d'encodeur :

- a) la source  $s$  est encodée en  $\hat{s}$  par un codeur de source SD. Puis,  $\hat{s}$  est encodé en  $(x_1, x_2)$  par le codeur de canal et transmis sur des canaux parallèles ;
- b) la source  $s$  est encodée en  $\hat{s}_1$  et  $\hat{s}_2$  par un codeur de source MD. Puis, chaque  $\hat{s}_i$  est encodé séparément en  $x_i$  par un codeur de canal et transmis sur le canal approprié.

Le premier, l'encodeur (a) introduit la diversité au niveau du codage de canal, le second, l'encodeur (b) au niveau du codage de source. Les auteurs de [47] ont montré que l'encodeur (b) est plus performant que l'encodeur (a) sur des canaux «on-off». Ce résultat s'explique facilement lorsque la qualité du signal reconstruit est grande. Dans ce cas, l'encodeur (b) peut introduire une redondance partielle en envoyant des signaux légèrement différents sur chaque canal, tandis que l'encodeur (a) est obligé d'introduire une redondance très forte pour éviter qu'un canal défaillant empêche le décodage de l'information véhiculée par le canal en état de marche. Dans ce cas, la distorsion sera la même que les canaux soient défaillants ou non. Cependant, l'encodeur (a) est plus performant que l'encodeur (b) pour des modèles de canal à valeurs continues tel qu'un canal à bruit gaussien additif (en anglais Additive White Gaussian Noise, AWGN) lorsque le système basé sur l'encodeur (b) effectue séparément le décodage de source et le décodage de canal. Si, par contre, un décodage conjoint-source canal est possible, l'encodeur (b) est au moins aussi bon que l'encodeur (a). Finalement, quelque soit le modèle canal envisagé, les auteurs de [47] concluent que le système le plus performant repose sur une architecture source-canal conjointe et un codage MD. Des solutions pratiques utilisant cette architecture ont été proposées dans [48, 49, 50]. Dans [48, 49] une MDSQ et des codes convolutifs récursifs systématiques (en anglais Recursive Systematic Convolutional, RSC) sont combinés dans un schéma de turbo décodage conjoint. Dans [50], un réseau bayésien est utilisé pour résoudre

---

le problème de décodage conjoint de descriptions multiples obtenues par une MDSQ avec un codage de Huffman.

L'utilisation du codage MD a été envisagée pour différents types de réseaux (Internet, réseaux *ad hoc*, pair-à-pair, réseaux de capteur, etc.) et suivant différentes topologies (réseaux en arbre, réseaux maillés, etc.). Alasti *et al.* [51] ont étudié l'utilisation du codage MD dans des réseaux ayant des problèmes de congestion. Dans le contexte d'un réseau simple représenté par un ensemble de queues parallèles, ils ont montré que lorsque la charge du réseau est très élevée, le codage MD s'avère plus efficace que les systèmes de codage SD. Cela est essentiellement dû au fait que les paquets MD sont plus compactes que les paquets SD. Ce résultat suggère qu'il est important de prendre en compte l'influence du débit sur la congestion. Les réseaux *ad hoc* sont des réseaux capables de s'organiser sans infrastructure définie préalablement. Les nœuds (ou stations) du réseau communiquent directement entre eux sans passer par un point d'accès, chaque nœud faisant office de routeur pour les autres nœuds du réseau. Sur ces réseaux, un protocole de routage est utilisé par chaque nœud pour construire les chemins entre les différents nœuds. La topologie du réseau pouvant changer à tout moment, l'utilisation de protocole de routage classique tels que ceux utilisés pour Internet est impossible. Barrenechea *et al.* [52] ont montré que le codage MD combiné avec un algorithme de routage multi-chemins qui répartit la charge uniformément sur les nœuds du réseau peut améliorer la QoS dans les réseaux ad-hoc maillés denses.

## 1.7 Applications en vidéo

Comme nous avons pu le voir, ces dernières années ont vu naître de nombreux et divers algorithmes de codage MD. Une des applications du codage MD qui a reçu le plus d'attention récemment est la transmission vidéo. La motivation principale pour l'utilisation du codage MD dans ce domaine est sa capacité d'assurer une qualité minimum sans nécessiter la retransmission de paquets perdus. Cette particularité rend le codage MD particulièrement intéressant pour des applications interactives, en temps réel, pour lesquelles une retransmission de paquets serait jugée inacceptable. Le problème posé par le codage vidéo par des descriptions multiples est lié à la prédiction de mouvement. La prédiction de mouvement est un composant fondamental dans tous les codeurs vidéo standard. Cependant, du fait de la prédiction, une erreur de transmission sur une description peut se répercuter sur un nombre important d'images ; on parle alors de drift. Il est donc impératif que les codeurs vidéo par descriptions multiples soient construits en tenant compte de l'erreur de propagation que des erreurs de transmission peuvent engendrer. La manière la plus simple de se prémunir contre cet effet de drift est d'introduire périodiquement des trames codées en Intra, ce qui a pour effet de réduire l'efficacité de codage. On suppose en général que l'erreur de prédiction dans la distorsion latérale n'est pas corrélée avec le reste de l'erreur introduite par l'encodage MD. Cette hypothèse n'est cependant plus valide si les codeurs encodent également l'erreur de prédiction.

Wang, Reibman et Lin *et al.* [53] ont proposé une classification des codeurs vidéo par descriptions multiples suivant le fonctionnement de leur prédicteur et suivant le type de redondance utilisé. Les prédicteurs sont rangés en quatre classes :

- A) Un prédicteur de classe A ne souffre pas de problème de prédiction. Ainsi, quelque soit l'état du décodeur, l'encodeur et le décodeur utilisent les mêmes signaux de prédiction.
  - B) Un prédicteur de classe B est identique à celui qui serait utilisé pour le codage SD. Ce prédicteur minimise l'erreur de prédiction et donc n'introduit aucune redondance.
-

En général, un décodeur avec ce type de prédicteur ne pourra pas utiliser le signal de prédiction si les deux descriptions ne sont pas reçues, ce qui aura pour conséquence d'augmenter la distorsion latérale.

- C) Enfin, un prédicteurs de classe C peut contrôler le compromis entre l'efficacité de la prédiction et l'introduction de drift.

Les quatre types de redondance envisagés sont les suivants :

- a)  $\rho_a$  est la redondance introduite en codant l'erreur de prédiction par le codage MD.
- b)  $\rho_b$  est la redondance introduite en utilisant un prédicteur moins efficace que le prédicteur SD correspondant.
- c)  $\rho_c$  est la redondance introduite en envoyant une information pour réduire l'effet de drift.
- d)  $\rho_d$  est la redondance introduite en envoyant une information adjacente.

La fonction débit-distorsion du décodeur central qui pour un débit  $R$  donne une distorsion  $D_0$  est alors exprimé ainsi :

$$R(D_0) = R^*(D_0) + \rho = R^* + \rho_a + \rho_b + \rho_c + \rho_d. \quad (1.31)$$

où  $R^*(D_0)$  est le débit utilisé par le meilleur codeur SD pour obtenir une distorsion  $D_0$ .

Parmi les codeurs MD utilisant un ou plusieurs prédicteurs de classe A se trouvent ceux qui effectuent un sous-échantillonnage pour décomposer le signal d'origine en deux sous-ensembles, dans le domaine spatial, temporel, ou dans le domaine fréquentiel avant de réaliser la prédiction, en utilisant par exemple les techniques présentées dans [54]. Par exemple dans [55, 56], un sous-échantillonnage temporel produit deux descriptions codées séparément par un codeur prédictif. La redondance vient alors de la sous-optimalité de la prédiction qui s'effectue sur des trames moins corrélées. Dans ce cas, le système n'introduit pas de redondance  $\rho_a$  mais produit une redondance  $\rho_b$ . Une autre stratégie consiste à utiliser plusieurs boucles de prédiction. Par exemple, dans [57], un schéma de codage MD utilisant la technique DPCM est présenté. Deux prédicteurs sont utilisés à l'encodeur afin de simuler les deux scénarios possibles : les deux descriptions sont reçues ou l'une des deux est perdue. Les erreurs de prédiction sont ensuite codées par une DCT suivie d'une MDSQ. Un quantificateur grossier est utilisé dans les boucles de prédiction pour faire en sorte que les deux descriptions contribuent à améliorer la qualité au décodeur central. Dans ce cas, lorsque  $\rho_a$  augmente, l'efficacité de la prédiction augmente et  $\rho_b$  décroît. Une technique n'utilisant qu'un seul prédicteur est présentée dans [58]. Elle utilise un codage SNR-scalable et génère les deux descriptions en dupliquant la couche de base et en lui adjoignant une couche supplémentaire de raffinement différente pour chaque description. La prédiction n'est alors effectuée que sur la couche de base.

Les prédicteurs de classe B sont des prédicteurs SD donc  $\rho_b = 0$ . Dans [58, 59], le premier algorithme utilise une DCT pour coder l'erreur de prédiction, suivie d'une MDTC pour introduire de la corrélation. Un membre de chaque paire de coefficients est envoyé dans chaque description. Les vecteurs de mouvement sont dupliqués dans les deux descriptions donc  $\rho_d$  est non nulle. La redondance  $\rho_a$  est contrôlée par la MDCT. Dans le deuxième algorithme, les boucles de prédiction n'envoient qu'une partie de l'erreur de prédiction des décodeurs latéraux. Dans ce cas,  $\rho_c$  peut être ajustée indépendamment de  $\rho_a$ . Dans le troisième algorithme, seule la boucle de prédiction principale est utilisée. Dans ce cas,  $\rho_c$  est plus faible qu'avec le deuxième algorithme mais la distorsion latérale est plus grande. En général, lorsqu'on utilise un prédicteur de classe B, l'effet de drift ne disparaît

jamais totalement. Cependant, on peut réduire son impact en augmentant la fréquence d'apparition des trames codées en Intra.

Dans [60, 61], la prédiction d'une trame s'effectue à partir des deux trames précédemment décodées. Le schéma de codage MD réalise un sous-échantillonnage temporel de l'erreur de prédiction par un facteur 2 afin d'obtenir deux descriptions. Lorsqu'une seule description est reçue, le décodeur ne peut se servir que d'une seule des trames qui ont servies à faire la prédiction. Dans ce cas, l'effet de drift n'est pas totalement éliminé mais il est possible de le diminuer en ajustant les paramètres de la prédiction, c'est-à-dire en augmentant la redondance  $\rho_b$ .

Enfin, une dernière stratégie consiste à utiliser des codes correcteur d'erreurs pour ajouter une protection inégale sur les différentes couches d'un flux vidéo scalable [45, 62, 63].

Le codage MD a également été utilisé en codage vidéo t+2D avec filtrage temporel compensé en mouvement. Par exemple, dans [64, 65], la redondance temporelle est introduite par des bancs de filtre 3-bandes. Une autre direction envisagée pour générer des descriptions dans un système de codage vidéo basé ondelettes est l'utilisation d'une décomposition polyphase des coefficients dans le domaine temporel et spatio-temporel [66, 67, 68].

De nombreux problèmes de communication sur réseau ne permettent pas la mise en œuvre du théorème de séparation source/canal. De plus, des contraintes telles que la nécessité de communiquer en temps réel ou un contrôle strict de la charge du réseau peuvent empêcher la retransmission de paquets. Lorsque de telles conditions sont réunies, seul le codage conjoint source-canal peut permettre d'augmenter la robustesse. McCanne *et al.* [69, 70] ont proposé l'utilisation du codage conjoint source-canal pour la transmission multi-points de vidéo sur un réseau hétérogène. Dans leur approche, chaque récepteur peut choisir dynamiquement la capacité du réseau local en ajustant la qualité de la vidéo reçue. Sur un réseau tel qu'Internet, plusieurs descriptions peuvent être envoyés à un récepteur suivant plusieurs chemins [71]. Dans [72], Pereira *et al.* ont proposé un schéma de codage MD basé sur une DWT et une technique d'allocation de débit dérivée de [73] et adaptée au codeur vidéo 3D présenté dans [74]. Le taux de redondance est alors contrôlé automatiquement en fonction de l'état et du modèle du réseau. Ils considèrent des canaux binaires symétriques, gaussiens et UMTS. Une approche de streaming pair à pair robuste a été présentée par Padmanabhan, Wang et Chou [75], basée sur un algorithme de construction de multiples arbres de distribution afin d'introduire une diversité dans les chemins réseau utilisés, et combiné avec un codage MD pour introduire la redondance dans les données transmises. Un autre système pour transmettre une vidéo sur un réseau pair à pair est présenté dans [76]. Il repose sur une architecture de codage MD qui adapte le nombre de descriptions, leur type et la quantité de redondance qu'elles contiennent en fonction de l'état du réseau.

## 1.8 Conclusion

Dans ce chapitre nous avons présenté le problème général du codage MD. Après une étude des résultats théoriques, nous nous sommes intéressés aux solutions de codage pratiques, leur utilisation sur des canaux et réseaux concrets, ainsi que des applications en codage vidéo.

Un concurrent potentiel du codage MD est le codage par couches (en anglais layered coding) qui consiste à partitionner la source en différentes couches, la qualité au décodeur dépendant alors du nombre de couches reçues. La différence principale avec le codage MD réside dans la dépendance entre les couches. La couche principale est appelée la couche de base, les autres couches servant à améliorer sa qualité. La couche de base ne doit jamais être

perdue : si cette couche est perdue, les autres couches ne pourront pas être décodées. C'est pourquoi la couche de base doit être davantage protégée contre les erreurs de transmission que les autres couches. En vidéo, le codage par couches peut être considéré comme du codage vidéo scalable (en anglais Scalable Video Coding). Cette technique diffère du codage MD dans le sens où, dans ce dernier, tous les sous-ensembles de descriptions doivent être décodables, tandis qu'avec un codage par couches, seules quelques combinaisons de couches sont décodables. Le codage par couches nécessite un système de gestion de la priorité des paquets, qui n'est pas forcément disponible sur le réseau. L'utilisation de l'une ou l'autre technique dépend de l'application envisagée et des outils disponibles sur le réseau.

Un codeur MD peut choisir de ne pas envoyer toutes les descriptions au décodeur. Par exemple, le transmetteur peut décider de ne pas envoyer certaines descriptions pour mieux exploiter la capacité du canal, ou si le récepteur n'a pas besoin de toutes les descriptions, ou s'il n'est pas capable de toutes les utiliser. Cette scalabilité est un autre atout du codage MD qui n'a cependant pas été conçu dans ce but.

Nous avons vu que le codage MD est particulièrement adapté à la transmission de paquets vidéo sur des réseaux denses, à forte charge ou avec des contraintes de transmission en temps réel. Cependant, l'effet de drift, lié à la prédiction, que génèrent les descriptions rend l'application du codage MD délicate dans un codeur conventionnel. Cela n'a pas empêché le développement de nombreux systèmes de codage MD en vidéo. La performance de ces systèmes dépend de leurs stratégies pour générer des descriptions, gérer la prédiction et contrôler la propagation des erreurs dues au drift. Les structures de codage capable de diminuer les effets produits par le drift en contrôlant efficacement la redondance, et celles qui permettent de générer plus de deux descriptions nous semblent les plus prometteuses.

---

## Chapitre 2

# Codage de source distribué

*Ce chapitre commence par une présentation du problème du codage de source distribué et de ses limites théoriques pour le codage sans perte et avec pertes. Dans les deux parties suivantes, nous verrons différentes techniques de codage de source distribué sans perte reposant soit sur le codage de source, soit sur le codage de canal, puis nous verrons les techniques de codage avec pertes reposant exclusivement sur le codage de canal. Enfin, la dernière partie sera consacrée au codage vidéo distribué mono-vue. Après avoir donné les motivations, nous décrirons les deux architectures principales de codage vidéo distribué : une architecture de codage par bloc sans voie de retour et une architecture de codage par trame avec voie de retour. Enfin, nous terminerons ce chapitre avec les dernières avancées dans ce domaine en vidéo.*

### 2.1 Introduction

Dans les systèmes de codage vidéo classiques, tels que H.26x ou MPEG-X, les dépendances temporelles entre les trames sont exploitées à l'encodeur par un codage prédictif compensé en mouvement, dont l'objectif est de donner à la vidéo une représentation plus compacte et donc plus simple à comprimer. Dans ces systèmes, la complexité de l'encodeur est 5 à 10 fois supérieure à celle du décodeur. En effet, l'encodeur effectue l'estimation de mouvement et transmet au décodeur le champ de mouvement. Le décodeur se sert des vecteurs de mouvement pour retrouver les prédicteurs et décodeur la vidéo. Cette architecture de codage est adaptée à la transmission de vidéo à partir d'un serveur vers des terminaux mobiles mais ne convient pas à l'envoi de vidéo par des terminaux mobiles vers une station de base. Pour ce genre d'applications, il est intéressant de rechercher un système de codage dual du précédent où la complexité serait déplacée de l'encodeur vers le décodeur.

Le codage de source distribué (en anglais *distributed source coding*, DSC) est un paradigme de codage basé sur deux résultats majeures en théorie de l'information : les théorèmes de Slepian-Wolf [77] et de Wyner-Ziv [15]. Le codage de source distribué est le codage séparé de deux (ou plus) séquences de variables aléatoires dépendantes. Leur encodage est fait séparément mais leur décodage est fait conjointement. Dans ce contexte, seul le décodeur peut exploiter les dépendances statistiques entre les sources. Un cas particulier de codage de source distribué est le codage de source avec information adjacente où une des séquences, appelée information adjacente, est parfaitement connue au décodeur mais pas à l'encodeur.

Partant d'une configuration de codage de source avec information adjacente, un nou-

---



veau paradigme en codage vidéo, appelé le codage vidéo distribué (en anglais Distributed Video Coding, DVC) a été établi. Bien que les fondements théoriques du codage de source distribué datent des années 1970, leurs applications dans des schémas de codage vidéo pratiques n'ont été proposées que très récemment. Une des raisons principales derrière ce développement tardif est l'évolution récente du codage de canal, en particulier l'émergence des turbo codes et des codes LDPC, qui a permis la construction des codes nécessaires au développement du codage vidéo distribué. Dans un système de codage vidéo distribué, une partie des dépendances statistiques, qui dans un système de codage vidéo classique seraient exploitées à l'encodeur, sont exploitées au décodeur. Par exemple, l'estimation de mouvement peut être réalisée au décodeur plutôt qu'à l'encodeur en codant une trame sur deux avec un codeur classique en mode Intra et en la considérant comme information adjacente au moment du décodage de ses trames voisines. Un scénario applicatif pour le codage vidéo distribué est la communication sans fil de signaux vidéo entre un terminal mobile et une station de base.

Dans ce chapitre, après avoir présenté les limites théoriques en codage de source distribué pour le codage sans perte et avec pertes, nous décrirons différentes techniques de codage sans perte et avec pertes, et nous terminerons par une présentation de deux architectures de codage vidéo distribué, ainsi que des derniers développements dans ce domaine.

## 2.2 Principes théoriques

Dans cette section, nous présentons les deux théorèmes principaux en codage de source distribué : le théorème de Slepian-Wolf [77] pour le codage sans perte et le théorème de Wyner-Ziv [15] pour le codage avec pertes.

### 2.2.1 Codage de Slepian-Wolf

Le codage de source distribué correspond au codage de deux (ou plusieurs) séquences aléatoires dépendantes, où l'encodage de chaque séquence est fait séparément et où leur décodage est fait conjointement (voir la figure 2.1). Seul le décodeur peut utiliser la corrélation statistique entre les deux séquences.

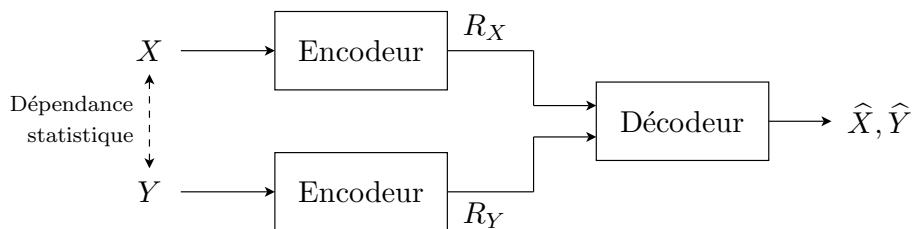


FIGURE 2.1 – Problème posé par Slepian et Wolf [77]. Deux sources corrélées sont codées séparément et décodées conjointement.

Soient deux séquences de variables aléatoires  $X$  et  $Y$  i.i.d. sur un alphabet fini. Le théorème de Shannon [78] nous dit que si les deux séquences sont encodées et décodées conjointement, un débit suffisant pour coder sans perte  $X$  et  $Y$  est leur entropie conjointe  $H(X, Y)$ . Par exemple, il est possible de compresser  $Y$  en  $H(Y)$  bits et, en utilisant la connaissance complète de  $Y$  à l'encodeur et au décodeur, on peut compresser  $X$  en  $H(X|Y)$  bits. Mais que se passe-t-il si l'encodage de  $X$  et  $Y$  est fait séparément ?

Une manière naïve de répondre à cette question serait de coder les séquences séparément avec un débit  $R = H(X) + H(Y)$ , plus grand que  $H(X, Y)$  si les deux séquences sont corrélées. Slepian et Wolf [77] ont montré que  $R = H(X, Y)$  est suffisant même lorsque l'encodage est fait séparément. Plus précisément, le théorème de Slepian-Wolf nous dit que :

$$R_X \geq H(X|Y), \quad (2.1)$$

$$R_Y \geq H(Y|X), \quad (2.2)$$

$$R_X + R_Y \geq H(X, Y). \quad (2.3)$$

La région de Slepian-Wolf des débits atteignables est illustrée sur la figure 2.2.

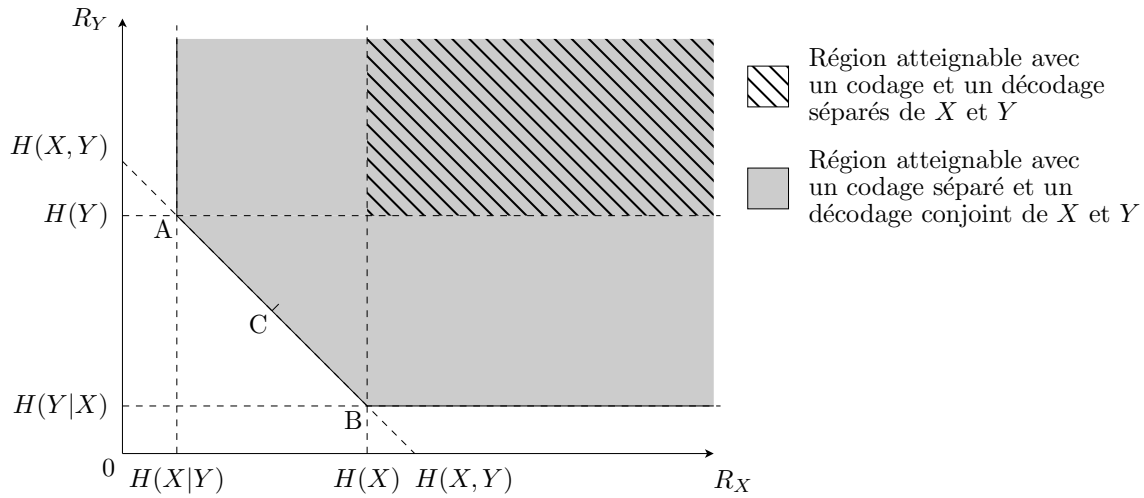


FIGURE 2.2 – Région de Slepian et Wolf [77] des débits atteignables lorsque deux sources  $X$  et  $Y$  sont codées séparément et décodées conjointement.

Le problème de la compression avec information adjacente au décodeur (voir la figure 2.3) ou codage asymétrique est un cas particulier du problème de codage de source distribué. Il correspond aux points A et B sur la figure 2.2. Prenons l'exemple du point A. Si  $R_Y = H(Y)$  bits sont suffisants pour décrire  $Y$ , alors  $R_X = H(X|Y)$  bits doivent être suffisants pour décrire  $X$ . Plus précisément, le débit minimum pour décrire  $X$  correspond à son entropie  $H(X)$  à laquelle on aurait substitué l'information mutuelle entre  $X$  et  $Y$  :

$$H(X|Y) = H(X) - I(X; Y) \quad (2.4)$$

En effet,  $Y$  étant connue au décodeur, elle peut servir d'*information adjacente* pour *aider* au décodage de  $X$  et ainsi contribuer à réduire le débit nécessaire pour représenter  $X$ . Le même raisonnement peut s'appliquer au point B. L'étude du cas asymétrique est très populaire dans la littérature car il est plus simple à mettre en oeuvre que le cas symétrique.

### 2.2.2 Limites théoriques du codage de source avec information adjacente

Dans la section précédente, nous avons vu les résultats obtenus par Slepian et Wolf en 1973 dans le cas du codage sans perte. Trois ans plus tard, Wyner et Ziv [15] ont déterminé la région débit-distorsion pour deux sources corrélées  $X$  et  $Y$  de variables aléatoires prenant leurs valeurs dans deux ensembles finis  $\mathcal{X}$  et  $\mathcal{Y}$  et une mesure de distorsion  $d(X, \hat{X})$  où

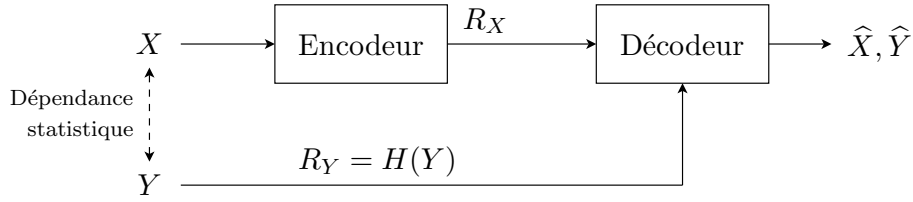


FIGURE 2.3 – Codage de source avec information adjacente.

$\hat{X}$  est la version reconstruite de  $X$ . Le problème considéré est le codage de source avec information adjacente, où  $X$  représente la source et  $Y$  l'information adjacente. Les fonctions débit-distorsion pour ce problème ne sont pas connues dans le cas général.

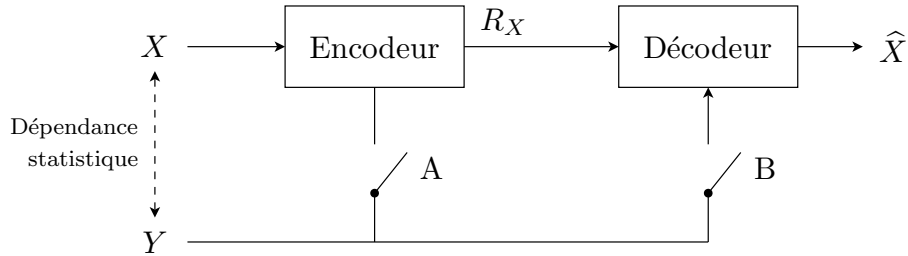


FIGURE 2.4 – Codage d'une source à valeurs continues avec information adjacente.

Le problème est illustrée sur la figure 2.4. Trois cas de figure sont considérés, suivant l'état des interrupteurs A et B :

- i) Les interrupteurs A et B sont ouverts : L'information adjacente n'est pas disponible ;
- ii) Les interrupteurs A et B sont fermés : L'encodeur et le décodeur ont accès à l'information adjacente (codage classique) ;
- iii) L'interrupteur A est ouvert et l'interrupteur B est fermé : Seul le décodeur a accès à l'information adjacente (c'est le cas du codage distribué).

Soient  $R_X(D)$ ,  $R_{X|Y}(D)$  et  $R^*(D)$  les fonctions débit-distorsion des cas i), ii) et iii) respectivement pour des blocs de grande taille et une distorsion moyenne proche de  $D$  :  $\mathbb{E}[d(X, \hat{X})] \leq D$ . Pour  $D \geq 0$ , on définit  $\mathcal{M}_0$  comme l'ensemble des distributions de probabilité  $p(x, y, \hat{x})$ ,  $x \in \mathcal{X}$ ,  $y \in \mathcal{Y}$ ,  $\hat{x} \in \hat{\mathcal{X}}$ , tel que la distribution marginale  $\sum_{\hat{x} \in \hat{\mathcal{X}}} p(x, y, \hat{x})$  est la distribution de  $(X, Y)$ , et :

$$\sum_{x, \hat{x}, y} D(x, \hat{x}) p(x, y, \hat{x}) \leq D. \quad (2.5)$$

Dans le cas de i), la théorie de Shannon nous dit que :

$$R_X(D) = \min_{p \in \mathcal{M}_0(D)} I(X; \hat{X}), \quad (2.6)$$

et dans le cas de ii) :

$$R_{X|Y}(D) = \min_{p \in \mathcal{M}_0(D)} I(X; \hat{X}|Y). \quad (2.7)$$

Dans le cas de iii), Wyner et Ziv ont déterminé la fonction débit-distorsion suivante :

$$R_{X|Y}^*(D) = \min_{p \in \mathcal{M}(D)} [I(X; Z) - I(Y; Z)] \quad (2.8)$$

où  $Z$  est également un ensemble de variables aléatoires prenant ses valeurs dans  $\mathcal{Z}$ , tel que la distribution marginale de  $X, Y$  ne dépende pas de  $Z$ , et tel que  $Y$  et  $Z$  soit conditionnellement indépendantes suivant  $X$  :

$$p(x, y, z) = Q(x, y)p_t(z|x), \quad (2.9)$$

où  $p_t(z|x)$  peut être vue comme une probabilité de transition d'un canal de test dont l'entrée serait  $X$  et la sortie  $Z$ .  $\mathcal{M}(D)$  est l'ensemble de  $p(x, y, z)$  qui satisfont (2.9) et qui ont la propriété qu'il existe une fonction  $f : \mathcal{Y} \times \mathcal{X} \rightarrow \hat{\mathcal{X}}$  telle que :

$$\mathbb{E}[D(X, \hat{X})] \leq D, \quad \text{où } \hat{X} = f(Y, Z). \quad (2.10)$$

Pour mieux comprendre la signification de  $Z$ , on peut penser à  $X, Y, Z, \hat{X}$  comme étant générés par la configuration de la figure 2.5.

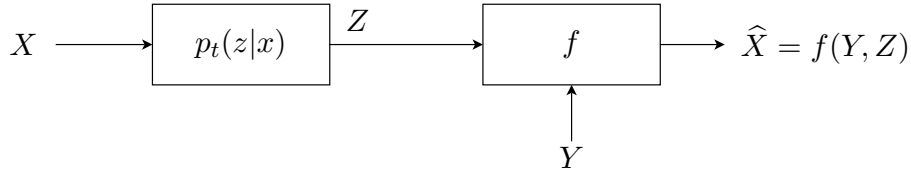


FIGURE 2.5 – Illustration des relations entre  $X, Y$  et  $Z$ .

Wyner et Ziv ont également comparé les différentes fonctions débit-distorsion et ont montré que :

$$R_{X|Y}(D) \leq R_{X|Y}^*(D) \leq R_X(D) \quad (2.11)$$

Lorsque  $D = 0$ , l'inégalité de gauche dans (2.11) devient  $R_{X|Y}^*(0) = R_{X|Y}(0) = H(X|Y)$ . Dans ce cas, la limite est donnée par le théorème de Slepian-Wolf.

Dans le cas de deux sources gaussiennes, Wyner et Ziv nous donnent le théorème suivant :

**Théorème 3** Soit  $(X(1), Y(1)), (X(2), Y(2)) \dots$  une séquence de variables aléatoires *i.i.d.* conjointement gaussiennes où  $Y(k) = X(k) + U(k)$ ,  $\mathbb{E}[X^2] = \sigma_X^2$  et  $\mathbb{E}[U^2] = \sigma_U^2$ . Seul le décodeur a accès à l'information adjacente  $\{Y(k)\}$ . La mesure de distorsion est  $d(x, \hat{x}) = \|x - \hat{x}_m\|^2$ . L'ensemble des paires atteignables  $(R_{X|Y}^*, D)$  est alors donné par

$$R_{X|Y}^*(D) = R_{X|Y}(D) = \begin{cases} \frac{1}{2} \log \left[ \frac{\sigma_X^2 \sigma_U^2}{(\sigma_X^2 + \sigma_U^2) D} \right], & \text{si } 0 < D \leq \frac{\sigma_X^2 \sigma_U^2}{\sigma_X^2 + \sigma_U^2}, \\ 0, & \text{si } D > \frac{\sigma_X^2 \sigma_U^2}{\sigma_X^2 + \sigma_U^2}. \end{cases} \quad (2.12)$$

Ce théorème nous dit que pour deux sources gaussiennes et une mesure de distorsion quadratique, nous avons  $R_{X|Y}^*(D) = R_{X|Y}(D)$ . Par conséquent, la fonction débit-distorsion est la même que l'information adjacente  $Y$  soit connue ou non à l'encodeur. Plus récemment, Pradhan *et al.* [79] ont montré que ce théorème peut être généralisé pour n'importe quelles distributions pour  $X$  et  $Y$  à condition que le modèle de corrélation  $U$  entre  $X$  et  $Y$  soit une Gaussienne. Ce résultat fondamental a motivé la recherche dans la construction de codes de type Wyner-Ziv. Il faudra cependant attendre une vingtaine d'années avant de voir apparaître les premiers systèmes de codage [80].

### 2.2.3 Codage de source multiterminal

Une autre extension du codage de Slepian-Wolf est celle où on autorise l'introduction de distorsion dans la reconstruction des sources. Ce problème s'appelle le codage de source multiterminal (MT) [81]. Il existe deux classes de problèmes MT présentées sur la figure 2.6. Si les encodeurs ont directement accès à une des deux sources, on parle de codage de source MT direct [81, 82]. Si, par contre, aucun des encodeurs ne peut observer directement la source qui doit ensuite être reconstruite au décodeur, mais observe à sa place une de ses versions bruitées, alors on parle de codage de source MT indirect [83, 84].

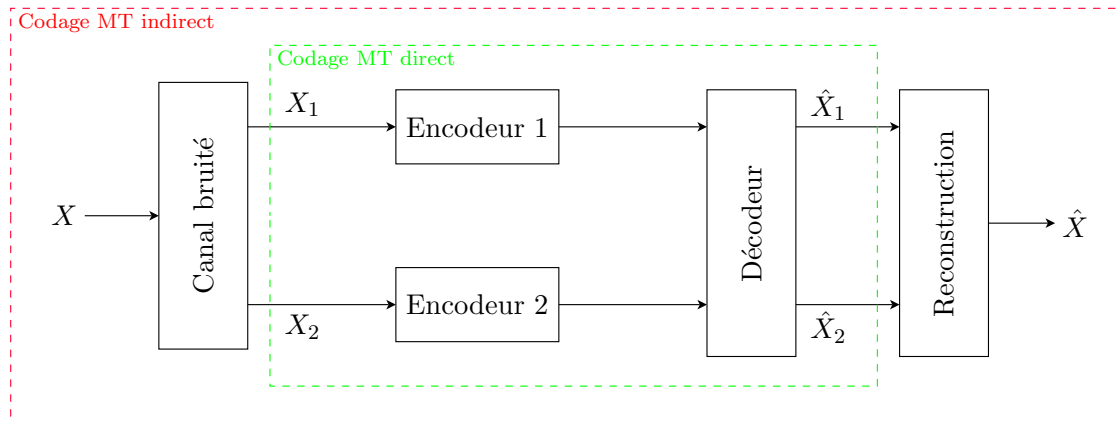


FIGURE 2.6 – Codage multiterminal direct et indirect de deux sources corrélées.

Le problème du codage MT consiste à déterminer la région des débits auxquels les sources (ou les observations bruitées) peuvent être comprimées séparément, de façon à ce qu'au décodeur, elles puissent être retrouvées conjointement avec une certaine distorsion. Dans le cas général, les régions des débits atteignables ne sont pas connues, seules des limites internes et externes ont été définies dans [81]. Le cas gaussien avec une mesure de distorsion quadratique est étudié dans [81, 82] pour le problème MT direct et dans [83, 84] pour le problème MT indirect.

## 2.3 Techniques de codage de Slepian-Wolf

Dans cette partie nous présentons différentes techniques permettant de coder séparément deux sources discrètes et de les décoder conjointement (codage de Slepian-Wolf). Les premières méthodes que nous allons voir sont basées sur du codage de source. Les secondes utilisent des techniques de codage de canal.

### 2.3.1 Solutions basées sur le codage de source

Les premières solutions de codage de Slepian-Wolf proposées dans la littérature reposent sur l'utilisation de codes à longueur variable. Dans cette section, nous commencerons par décrire la technique de Al Jabri et Al-Issa [85] puis nous présenterons le schéma de codage de source distribué de Zhao et Effros [86].

### 2.3.1.1 Technique de Al Jabri et Al-Issa

Les travaux de Al Jabri et Al-Issa [85] ont abouti au premier schéma de codage de sources binaires distribué utilisant les codes à longueur variable dans le cas de séquences binaires.

Soient  $X$  et  $Y$  deux sources binaires corrélées. La corrélation est représentée par la fonction de densité de probabilité discrète  $P_{XY}(x, y)$ . Comme le montre la figure 2.7, le problème posé est un problème de codage de source avec information adjacente, où  $X$  représente la source et  $Y$  l'information adjacente. L'encodeur de  $Y$  est appelé codeur primaire, il garantit à la source  $Y$  un débit de transmission proche de son entropie  $H(Y)$ . L'encodeur de  $X$  est le codeur secondaire. Ce dernier exploite les statistiques  $P_{XY}$  pour coder la source  $X$  avec un débit de l'ordre de  $H(X|Y)$ . Le décodeur, après avoir décodé  $Y$ , l'utilise comme information adjacente pour décodé  $X$ .

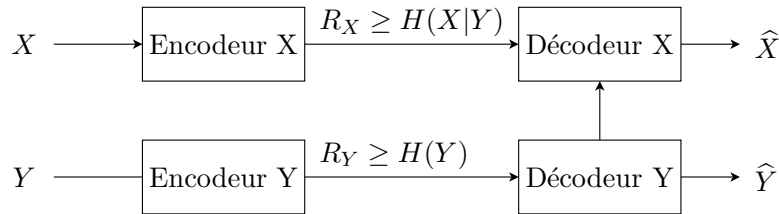


FIGURE 2.7 – Schéma de codage distribué de Al Jabri et Al-Issa [85].

Considérons l'exemple de deux sources  $X$  et  $Y$  prenant leurs valeurs dans les alphabets  $\mathbb{X} = \{x_1, x_2, x_3, x_4, x_5\}$  et  $\mathbb{Y} = \{y_1, y_2, y_3, y_4, y_5\}$ , respectivement. La fonction de densité de probabilité conjointe est donnée par la matrice suivante :

$$P_{XY}(x_i, y_j) = \begin{pmatrix} 3/20 & 1/15 & 1/20 & 0 & 0 \\ 3/20 & 0 & 0 & 1/30 & 1/12 \\ 0 & 1/15 & 1/20 & 0 & 0 \\ 0 & 1/15 & 1/20 & 1/30 & 1/12 \\ 0 & 0 & 0 & 1/30 & 1/12 \end{pmatrix} \quad (2.13)$$

Les probabilités marginales de  $X$  et  $Y$  sont données dans le tableau 2.1.

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
$p_X(x_i)$	4/15	4/15	7/60	7/30	7/60
$p_Y(y_i)$	3/10	1/5	3/20	1/10	1/4

TABLE 2.1 – Probabilités marginales de  $X$  et  $Y$ .

L'encodage de  $Y$  est fait séparément en utilisant les probabilités marginales du tableau 2.1 avec, par exemple, un codage de Huffman, de façon à s'approcher le plus possible de  $H(Y)$ . L'encodage de  $X$  doit prendre en compte le fait que  $Y$  est parfaitement connue au décodeur. L'idée consiste à représenter par un seul mot de code les symboles de  $\mathbb{X}$  qui peuvent être décodés sans ambiguïté en connaissant  $Y$ . Par exemple, on remarque, à partir de la matrice (2.13), que  $p(x_2, y_i)$ ,  $i = 2, 3$  et  $p(x_3, y_i)$ ,  $i = 1, 4, 5$  sont nulles. Donc, si on utilise un seul mot de code pour représenter  $x_2$  et  $x_3$ , la réception de  $y_2$  ou  $y_3$  nous indiquera que le mot de code correspond à  $x_2$ , tandis que  $y_1$ ,  $y_4$  et  $y_5$  décodéront  $x_3$ .

Symboles de $\mathbb{X}$	Sous-ensembles $\mathbb{Y}_i$
$x_1$	$\{y_1, y_2, y_3\}$
$x_2$	$\{y_1, y_4, y_5\}$
$x_3$	$\{y_2, y_3\}$
$x_4$	$\{y_2, y_3, y_4, y_5\}$
$x_5$	$\{y_4, y_5\}$

TABLE 2.2 – Correspondance des symboles  $\mathbb{X}$  avec les sous-ensembles  $\mathbb{Y}_i$ .

L'algorithme d'Al Jabri et Al-Issa consiste à trouver un groupement optimal pour les symboles de  $\mathbb{X}$ . On commence par construire, pour chaque symbole  $x_i \in \mathbb{X}$ , un sous-ensemble  $\mathbb{Y}_i$  de  $\mathbb{Y}$ , tel que :

$$\mathbb{Y}_i = \{y : y \in \mathbb{Y} \text{ et } p(y|x_i) > 0\} \quad (2.14)$$

Pour l'exemple considéré, la correspondance entre les symboles de  $\mathbb{X}$  et les sous-ensembles  $\mathbb{Y}_i$  est donnée par le tableau 2.2. Ensuite, on cherche toutes les partitions de  $\mathbb{X}$  en faisant en sorte qu'aucun sous-ensemble n'ait une correspondance avec les sous-ensembles  $\mathbb{Y}_i$  définis dans l'étape précédente. On obtient alors les partitions de  $\mathbb{X}$  données dans la tableau 2.3. A titre d'exemple, on ne pourrait pas mettre  $x_1$  et  $x_2$  dans un même sous-ensemble puisqu'il y aurait alors une ambiguïté au niveau du décodage. En effet, la correspondance des symboles  $x_1$  et  $x_2$  avec les sous-ensembles  $\mathbb{Y}_i$  est respectivement  $\{y_1, y_2, y_3\}$  et  $\{y_1, y_4, y_5\}$  qui ont en commun le symbole  $y_1$ . Enfin, on choisit la partition qui fournit la plus faible entropie. Dans notre exemple, il s'agit de  $\{x_1, x_5\}$ ,  $\{x_2, x_3\}$ ,  $\{x_4\}$ . Les sous-ensembles sont ensuite codés à l'aide d'un code de Huffman.

Partitions de $\mathbb{X}$	Entropie
$\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}, \{x_5\}$	2.23012
$\{x_1\}, \{x_2\}, \{x_3, x_5\}, \{x_4\}$	1.99679
$\{x_1\}, \{x_2, x_3\}, \{x_4\}, \{x_5\}$	1.89028
$\{x_1, x_5\}, \{x_2\}, \{x_3\}, \{x_4\}$	1.89028
$\{x_1, x_5\}, \{x_2, x_3\}, \{x_4\}$	1.55044

TABLE 2.3 – Partitions de  $\mathbb{X}$  et leur entropie correspondante.

### 2.3.1.2 Technique de Zhao et Effros

La technique de Zhao et Effros [86] est également basée sur un code à longueur variable (code de Huffman ou code arithmétique) pour comprimer deux sources  $(X, Y) \in \mathcal{X} \times \mathcal{Y}$  sans mémoire et aux alphabets finis. Le schéma de codage est le même que pour Al Jabri Al-Issa (voir la figure 2.7). La source  $Y$  est codée séparément à un taux proche de son entropie  $H(Y)$  et  $X$  est codé à l'aide d'un code optimal à un taux proche de  $H(X|Y)$ . La fonction de densité de probabilité de  $X$  et  $Y$  est  $p(x, y)$ . L'algorithme de Zhao et Effros comprend deux parties. La première est la construction d'un arbre de partition. La deuxième consiste à assigner un mot de code à chaque nœud de l'arbre.

Commençons par donner quelques définitions :

- Soient  $x_1, x_2 \in \mathcal{X}$ . On dit que  $x_1$  et  $x_2$  sont associés sous  $p(x, y)$  si  $p(x_1, y)p(x_2, y) = 0$  pour tout  $y \in \mathcal{Y}$ .

- L'ensemble  $\mathcal{G} = (x_1, x_2, \dots, x_m)$  est appelé groupe de niveau 1 pour  $p(x, y)$ , si chaque paire de membres distincts  $x_i, x_j \in \mathcal{G}$  est associée sous  $p(x, y)$ .
- $\mathcal{G} = (\mathcal{R} : \mathcal{C}(\mathcal{R}))$  est appelé groupe de niveau  $M$  pour  $p(x, y)$  si :
  - pour chaque  $\mathcal{G}' \in \mathcal{C}(\mathcal{R})$ , chaque paire  $y_1 \in \mathcal{R}$  et  $y_2 \in \mathcal{G}'$  est associée pour  $p(x, y)$ ,
  - $\mathcal{R}$  est un groupe de niveau 1 et  $\mathcal{C}(\mathcal{R})$  est un ensemble de groupes de niveaux  $M - 1$  ou moins, dont au moins un groupe est de niveau  $M - 1$ .
- $\mathcal{P} = \{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_m\}$  est une partition de  $\mathcal{Y}$  pour  $p(x, y)$  si  $\bigcup_{i=1}^m \mathcal{G}_i = \mathcal{Y}$  et  $\mathcal{G}_j \cap \mathcal{G}_k = \emptyset$  pour tout  $j \neq k$  et  $\mathcal{G}_i \in \mathcal{P}$  est un groupe pour  $p(x, y)$ .

La construction de l'arbre de partition,  $\mathcal{T}(\mathcal{P})$ , commence par la construction de  $\mathcal{T}(\mathcal{G}_i)$  pour chaque  $i$ , puis par l'établissement de liens entre la racine de tous les  $\mathcal{T}(\mathcal{G}_i)$ ,  $i \in \{1, \dots, m\}$  à un seul nœud, défini comme étant la racine de  $\mathcal{T}(\mathcal{P})$ .

Prenons l'exemple du tableau 2.4. La figure 2.8 (gauche) montre l'arbre de partition  $\mathcal{T}(\mathcal{P})$  avec  $\mathcal{P} = \{(x_3, x_6), \mathcal{G}(3)\}$  où  $\mathcal{G}(3) = ((x_7) : \{(x_0), (x_1), ((x_2) : \{(x_4), (x_5)\})\})$  est un groupe de niveau 3. Il est clair que  $x_7$  et  $\{x_0, x_1, x_2\}$  sont associés sous  $p(x, y)$  de même que  $x_2$  et  $\{x_4, x_5\}$ .

	$x/y$	$y_0$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$y_6$	$y_7$
$p(x_i, y_i)$	$x_0$	0.04	0	0.04	0.02	0	0	0	0
	$x_1$	0	0.04	0	0	0.05	0.1	0	0
	$x_2$	0.15	0	0.05	0	0	0	0	0
	$x_3$	0	0.05	0	0.06	0	0	0	0
	$x_4$	0	0.06	0	0	0.05	0	0	0
	$x_5$	0	0	0	0.01	0.02	0.03	0	0
	$x_6$	0	0	0.01	0	0	0.06	0.02	0.01
	$x_7$	0	0	0	0	0	0	0.05	0.08

TABLE 2.4 – Fonctions de densité de probabilité  $p(x, y)$ .

Soit  $\mathcal{G}$  un groupe de niveau 1 situé à une profondeur  $d$  dans  $\mathcal{T}(\mathcal{P})$ . Soit  $\mathbf{n}$  le chemin de la racine de l'arbre vers le nœud  $\mathcal{T}(\mathcal{G})$  dans l'arbre. Pour décrire chaque chemin dans l'arbre de manière unique, un ordre de descente d'un nœud à un autre est fixé et un nombre leur est attribué (de gauche à droite). Les enfants de  $\mathbf{n}$  sont étiquetés  $\mathbf{n}1, \mathbf{n}2, \dots, \mathbf{n}K(\mathbf{n})$ , où  $\mathbf{n}k$  est un vecteur créé par la concaténation de  $k$  à  $\mathbf{n}$  et  $K(\mathbf{n})$  est le nombre d'enfants de  $\mathbf{n}$ . Les étiquettes de l'arbre de la figure 2.8 (gauche) sont illustrées sur la figure 2.8 (centre).

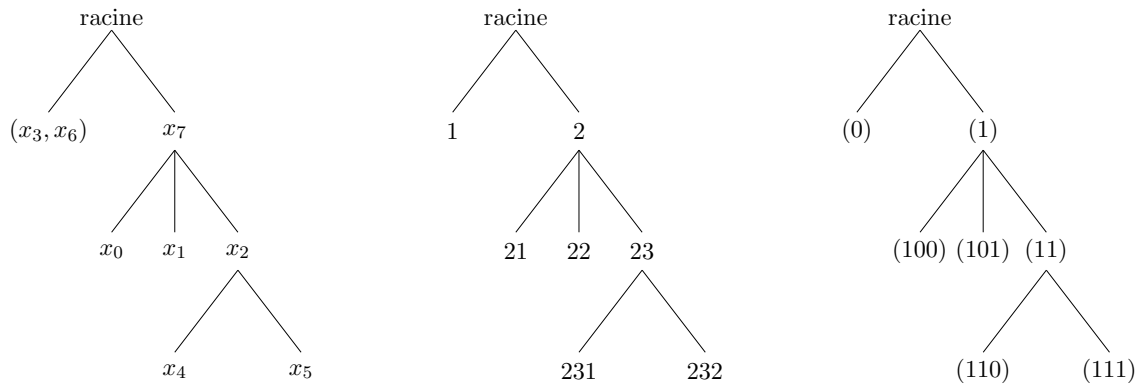


FIGURE 2.8 – Arbre de partition (gauche), étiquettes (centre) et mots de code associés (droite).



Une fois l'arbre de partition construit, il s'agit maintenant de déterminer le code à longueur variable optimal. Un code  $\gamma_Y$  est dit sans perte pour  $p(x, y)$  s'il respecte les conditions suivantes : Soit  $\mathcal{P}$  une partition de  $\mathcal{Y}$  pour  $p(x, y)$ . Soient  $\mathcal{T}(\mathbf{n}) \in \mathcal{T}(\mathcal{P})$  et  $y_1, y_2 \in \mathbf{n}$  et  $y_3 \in \mathbf{nk}, k \in \{1, \dots, K(\mathbf{n})\}$ .

- $\gamma_Y(y_1) = \gamma_Y(y_2)$ ;
- $\gamma_Y(y_1)$  est un préfixe de  $\gamma_Y(y_3)$ ;
- $\{\gamma_Y(\mathbf{nk}) : k \in \{1, \dots, K(\mathbf{n})\}\}$  ne possède pas de préfixe.

La figure 2.8 (droite) montre un exemple de code sans perte adapté à la partition  $\{(x_3, x_6), \mathcal{G}(3)\}$ .

Dans [86], Zhao et Effros donnent un algorithme permettant de construire un code optimal sans perte pour une partition donnée, ainsi qu'un algorithme de construction d'une partition optimale. On pourra se référer à [87] pour avoir une description complète de ces algorithmes.

### 2.3.2 Solutions basées sur le codage de canal

Le problème de codage d'une source avec information adjacente est en fait un problème de codage canal. En effet, la corrélation entre la source  $X$  et l'information adjacente  $Y$  peut être modélisée par un «canal virtuel» comme le montre la figure 2.9 pour un modèle de canal binaire symétrique (à gauche) et un modèle de canal AWGN (à droite). L'entrée du canal est  $X$  et la sortie est  $Y$ .  $Y$  est considérée comme étant une version bruitée de  $X$ .

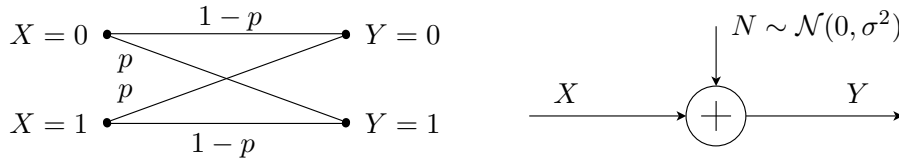


FIGURE 2.9 – L'information adjacente  $Y$  peut être vue comme une observation bruitée de  $X$ . Exemple d'un canal binaire symétrique (gauche) et d'un canal AWGN (droite).

Un canal binaire symétrique est caractérisé par sa probabilité de transition  $p$ . Ce paramètre correspond à la probabilité que le symbole reçu ne soit pas le même que celui transmis. La capacité du canal est  $1 - H(p)$  bits. Pour des sources binaires,  $H(p) = H(X|Y) = H(Y|X) = -p \log_2(p) - (1-p) \log_2(1-p)$ . Soit un code de canal  $(n, k)$ , qui pour  $n$  bits en entrée produit  $n - k$  bits en sortie. Si ce code permet d'atteindre la capacité du canal, alors  $\frac{k}{n} = 1 - H(p)$ . Le canal AWGN correspond à des sources à valeurs continues où  $Y = X + N$ ,  $N$  étant un bruit gaussien d'espérance nulle et de variance  $\sigma^2$ . La capacité d'un canal AWGN est :

$$C = \frac{1}{2} \log \left( 1 + \frac{P}{\sigma^2} \right) \quad (2.15)$$

où  $P$  est une contrainte d'énergie définie par :

$$P \geq \frac{1}{n} \sum_{i=1}^n x_i^2 \quad (2.16)$$

où  $(x_1, x_2, \dots, x_n)$  sont les mots de code transmis sur le canal. Les codes correcteur d'erreurs qui permettent de s'approcher au mieux de la capacité du canal, tels que les turbo codes ou les codes LDPC, sont également de bons candidats pour approcher la limite de Slepian-Wolf. Il est donc naturel d'envisager leur utilisation dans un codeur de Slepian-Wolf.

### 2.3.2.1 Codage par cosets : DISCUS

La première solution utilisant les codes de canal est le codage de source distribué basé syndromes (DIstributed Source Coding Using Syndromes, DISCUS) introduit en 1999 par Pradhan et Ramchandran [80]. Nous allons commencer par décrire cette approche avec un exemple simple avant de décrire son utilisation sous la forme d'un code en blocs.

Considérons deux mots binaires équiprobables  $X$  et  $Y$  représentés sur 3 bits ( $H(X) = H(Y) = 3$  bits). Ces deux mots sont corrélés dans le sens où la distance de Hamming entre  $X$  et  $Y$  est au plus 1 ( $d_H(X, Y) \leq 1$ ). Si  $Y$  est disponible à l'encodeur et au décodeur, comme il n'existe que 2 bits d'incertitude entre  $X$  et  $Y$ , on peut décrire  $X$  avec  $H(X|Y) = 2$  bits. Pour cet exemple, il est possible de montrer que, même si  $Y$  n'est connue qu'au décodeur, 2 bits suffisent pour décrire  $X$ . Pour cela, on commence par réunir les 8 valeurs possibles de  $X$  en 4 paires  $\delta_i (i = 0, 1, 2, 3)$ , de manière à ce que la distance de Hamming entre les éléments de chaque paire soit la plus grande possible :  $\delta_0 = \{000, 111\}$ ,  $\delta_1 = \{001, 110\}$ ,  $\delta_2 = \{010, 101\}$ ,  $\delta_3 = \{011, 100\}$ . Ici, la distance de Hamming est 3 pour toutes les paires. Chaque paire  $\delta_i$  est représentée par un coset  $i$  codé sur 2 bits comme le montre la figure 2.10. Ainsi, au lieu de décrire  $X$  avec 3 bits, chaque mot de code est associé au coset correspondant codé sur seulement 2 bits. Prenons comme exemple les mots de code suivants,  $X = \{010\}$  et  $Y = \{110\}$ . L'encodeur de  $X$  ne connaît pas la valeur de  $Y$ , il envoie le coset  $s$  correspondant à  $X$ , c'est-à-dire  $\{10\}$ . Le décodeur utilise sa connaissance de  $Y$  et de la corrélation entre  $X$  et  $Y$  pour trouver 4 choix équiprobables pour  $X$  :  $\{110\}$ ,  $\{100\}$ ,  $\{010\}$  ou  $\{111\}$ . Parmi ces choix, un seul appartient à la paire désignée par le coset, il s'agit de la valeur de  $X$ . Dans cet exemple, 5 bits suffisent pour comprimer séparément les deux sources  $X$  et  $Y$ . Ce débit étant égal à l'entropie conjointe,  $H(X, Y)$ , il n'est pas possible de faire mieux.

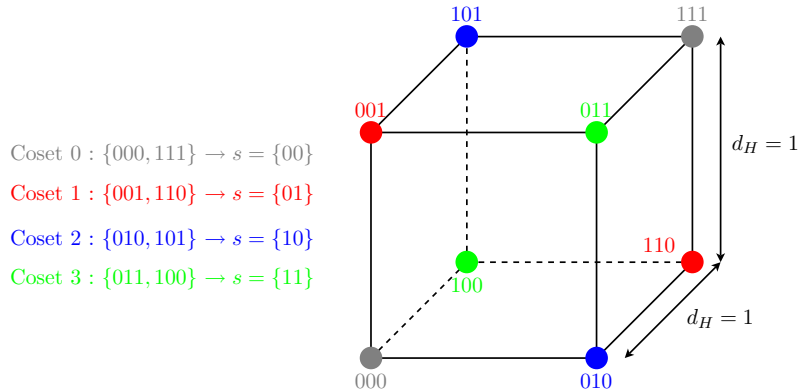


FIGURE 2.10 – Codage des cosets dans DISCUS. Les sommets de la même couleur appartiennent au même coset.

Le même exemple peut être traité avec un code en blocs  $(n, k, d)$ , où  $n$  est la longueur du code,  $k$  est la longueur du message et  $d$  est la distance minimale entre deux mots de code. Considérons pour cela la matrice de parité  $\mathbf{H}$  du code en blocs  $(3, 1, 3)$  définie par :

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad (2.17)$$

Pour chaque mot de code  $\mathbf{x}$ , un syndrome  $\mathbf{s} = \mathbf{H}\mathbf{x}$  représenté sur deux bits est transmis au décodeur. Les syndromes  $\{00\}$ ,  $\{01\}$ ,  $\{10\}$  et  $\{11\}$  correspondent aux cosets  $\{000, 111\}$ ,

$\{001, 110\}$ ,  $\{010, 101\}$  et  $\{011, 100\}$ , respectivement. Le décodeur reçoit l'indice du coset (par exemple  $k$ ) et décode comme valeur de  $X$  le vecteur dans le coset  $k$  dont la distance de Hamming à  $Y$  est la plus faible.

D'une manière plus générale, considérons deux sources  $X$  et  $Y$  telles que  $d_H(X, Y) \leq t$ . Soit  $\mathbf{H}$ , la matrice de parité d'un codeur en blocs  $(n, k, 2t + 1)$  dans sa forme systématique, c'est-à-dire  $\mathbf{H} = [\mathbf{A}|\mathbf{I}]$ . Soit  $\mathbf{s} = \mathbf{H}\mathbf{x}$  le syndrome de  $X$  lorsque la réalisation de  $X$  est  $\mathbf{x}$ . L'encodeur transmet le syndrome au décodeur. La tâche du décodeur est de trouver le vecteur le plus proche de  $\mathbf{y}$ , la réalisation de  $Y$ , dans le coset désigné par le syndrome  $\mathbf{s}$ . Il commence par calculer  $\mathbf{y}' = \mathbf{y} + \mathbf{a}$  (modulo 2), où  $\mathbf{a}$  est n'importe quel vecteur dans le coset dont le syndrome est  $\mathbf{s}$ . Soit  $\mathbf{x}'$  un des vecteurs les plus proches de  $\mathbf{y}'$  dans le coset dont le syndrome est un vecteur nul. On peut alors facilement voir que  $\mathbf{x} = \mathbf{x}' + \mathbf{a}$ . Dans ce cas, le taux de compression de  $X$  est  $\frac{n}{n-k}$  avec  $H(X|Y) = n - k$ .

### 2.3.2.2 Solutions basées sur les turbo codes

La première mise en oeuvre des turbo codes en codage de source distribué est attribuée à Garcia-Frias et Zhao [88, 89, 90] (voir l'annexe A pour plus de détails sur les turbo codes). Ils considèrent le cas symétrique où deux sources binaires corrélés  $X$  et  $Y$  i.i.d. sont codées séparément et décodées conjointement. Chaque source est codée par un turbo code poinçonné. Seule la moitié des bits systématiques à la sortie de chaque turbo code est envoyée au décodeur, les bits de parité poinçonnés sont, eux, placés dans une mémoire tampon. Le décodage est effectué par deux turbo décodeurs connectés en série. Après chaque itération dans un turbo décodeur, son information extrinsèque est utilisé comme information *à priori* par l'autre turbo décodeur. La conversion de l'information d'un décodeur à l'autre tient compte de la corrélation entre  $X$  et  $Y$  et de l'entrelaceur supplémentaire pour la source  $Y$ . Si cette corrélation n'est pas connue au décodeur, les auteurs de [88, 89] montrent qu'il est possible de l'estimer à partir de l'information extrinsèque de chaque turbo décodeur avec très peu d'impact sur les performances.

Dans [91], Bajcsy et Mitran considèrent le codage d'une source  $X$  avec une information adjacente  $Y$  uniquement connue au décodeur.  $X$  est codé par un turbo code qui consiste en une concaténation en parallèle de machines à états finis séparées par un entrelaceur. Le nombre d'états de chaque code élémentaire est  $2^k$  et leur taux de codage est  $\frac{k}{n}$  avec  $k > n$ . Le taux de codage global du turbo code est  $\frac{k}{2n}$ . Un exemple de machine à états finis avec  $k = 2$  et  $n = 1$  est illustré sur la figure 2.11. Les états sont représentés par les cercles. Sur chaque branche, les bits à gauche représentent les bits en entrée et le bit à droite celui en sortie. Le décodage s'effectue de manière itérative à l'aide d'un algorithme BCJR [92]. A chaque décodeur, les probabilités *a posteriori* sont estimées à partir du codeur correspondant, de  $Y$ , et de l'information extrinsèque provenant de l'autre décodeur. Les informations extrinsèques véhiculées par les décodeurs sont considérées comme étant des variables gaussiennes de variance unité et d'espérance 0.5.

Aaron et Girod [93] proposent un système de codage d'une source  $X$  avec information adjacente  $Y$  similaire à celui de Bajcsy et Mitran mais utilisant des codes RSC pour les codes élémentaires constitutifs du turbo code. Le schéma de codage est présenté sur la figure 2.12. L'information adjacente est transmise à un taux proche de son entropie, sans erreur, au décodeur. Le codage de  $X$  est effectué par un turbo code dont les codes RSC, séparés par un entrelaceur, ont un taux de codage de  $\frac{n-1}{n}$  (pour chaque séquence en entrée de taille  $n-1$ , ce type de code produit  $n$  mots de code comprenant les bits en entrée, appelés «bits systématiques», et un bit de parité). A la sortie des deux codes RSC, seuls les bits de

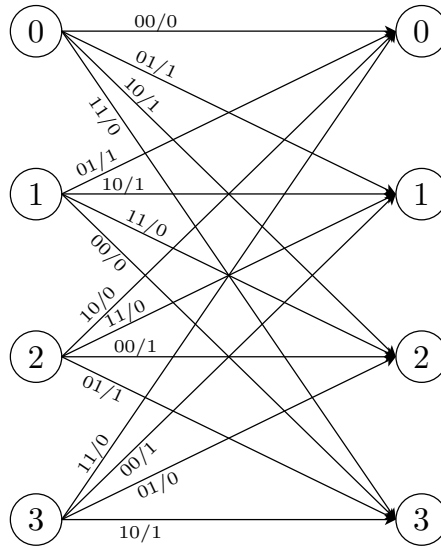


FIGURE 2.11 – Exemple d’une machine à états finis avec  $k = 2$  et  $n = 1$ .

parité,  $X_P^1$  et  $X_P^2$ , sont transmis au décodeur. Par conséquent, le taux de compression de  $X$  est  $\frac{2}{n-1}$  bits pour  $n$  bits en entrée. La destruction des bits systématiques est rendue possible par la présence de l’information adjacente  $Y$  au décodeur, qui peut être considérée comme une version bruitée de  $X$ . Le décodage est effectuée par un algorithme BCJR [92]. La différence avec un turbo décodage classique réside dans le fait que les probabilités de canal à l’entrée de chaque décodeur SISO (en anglais Single Input Single Output) sont calculées à partir des bits de parité,  $X_P^1$  et  $X_P^2$ , de  $Y$  et de la fonction de densité de probabilité  $p(y|x)$ .  $p(y|x)$  représente ici le modèle de corrélation entre  $X$  et  $Y$ . Elle remplace la probabilité de transition du canal dans l’algorithme de décodage MAP.

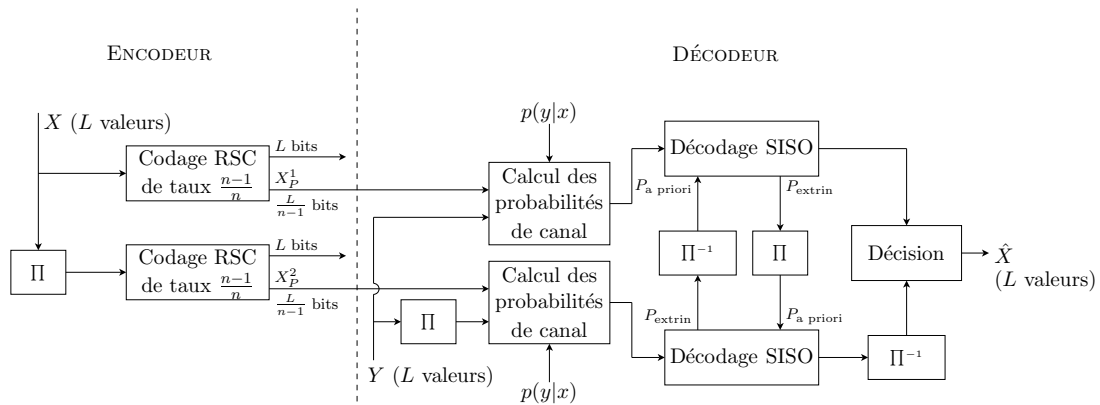


FIGURE 2.12 – Schéma de codage de Slepian-Wolf présenté par Aaron et Girod [93].

### 2.3.2.3 Solutions basées sur les codes LDPC

Pour mieux s’approcher de la borne théorique de Slepian-Wolf, Liveris *et al.* [94] ont proposé l’utilisation des codes LDPC (voir l’annexe B pour plus de détails sur les codes LDPC). Ils considèrent le problème de codage d’une source  $\mathbf{X} = [X_1, X_2, \dots, X_n]$  avec

information adjacente  $\mathbf{Y} = [Y_1, Y_2, \dots, Y_n]$ , où  $\mathbf{Y}$  est transmise à un taux proche de son entropie  $H(\mathbf{Y})$  et est uniquement disponible au décodeur, et où le codage de  $\mathbf{X}$  est réalisé par un code LDPC irrégulier.

Le code LDPC utilisé est un code  $(n, k)$  pouvant générer  $2^{n-k}$  syndromes différents. Le taux de compression est  $\frac{n}{n-k}$ . Soit  $\mathbf{H}$  la matrice de parité du code LDPC. A l'encodeur, la séquence de syndromes  $\mathbf{Z} = [Z_1, Z_2, \dots, Z_{n-k}]$  est obtenue par multiplication de  $\mathbf{X}$  avec la matrice de parité.

Le décodage de  $\mathbf{X}$  à partir de  $\mathbf{Y}$  et  $\mathbf{Z}$  est effectué par un algorithme itératif de propagation de croyance. Contrairement à l'annexe B, nous décrivons cet algorithme dans le domaine logarithmique. Commençons par donner quelques notations :

- $x_i, y_i \in \{0, 1\}$ ,  $i = 1, 2, \dots, n$  sont respectivement les valeurs courantes de  $X_i$  et  $Y_i$ , correspondant au  $i$ -ème nœud de variable  $c_i$  ;
- $l_i \in \{2, 3, \dots\}$ ,  $i = 1, 2, \dots, n$ , est le degré de  $c_i$ , c'est-à-dire le nombre de nœuds de contrôle auxquels  $c_i$  est connecté dans le graphe biparti ;
- $q_{i,m}^{\text{out}}(q_{i,m}^{\text{in}}) \in \mathbb{R}$ ,  $i = 1, 2, \dots, n$ ,  $m = 1, 2, \dots, l_i$ , est le rapport de log-vraisemblance (LLR) envoyé le long de la  $m$ -ème branche de (vers)  $c_i$  ;
- $s_j \in \{0, 1\}$ ,  $j = 1, 2, \dots, n - k$  est la valeur du syndrome  $Z_j$  correspondant au  $j$ -ème nœud de contrôle  $f_j$  ;
- $t_j \in \{2, 3, \dots\}$ ,  $j = 1, 2, \dots, n - k$ , est le degré de  $f_j$ , c'est-à-dire le nombre de nœuds de variable auxquels  $f_j$  est connecté dans le graphe biparti ;
- $r_{j,m}^{\text{out}}(r_{j,m}^{\text{in}}) \in \mathbb{R}$ ,  $j = 1, 2, \dots, n - k$ ,  $m = 1, 2, \dots, t_j$ , est le LLR envoyé le long de la  $m$ -ème branche de (vers)  $f_j$ .

Soit

$$q_{i,0} = \log \frac{\Pr[x_i = 0|y_i]}{\Pr[x_i = 1|y_i]} = (1 - 2y_i) \log \frac{1-p}{p} \quad (2.18)$$

où  $i = 1, 2, \dots, n$ ,  $p = \Pr[x_i \neq y_i] < 0.5$ . Alors, le LLR envoyé à partir de  $c_i$  le long de la  $m$ -ième branche est

$$q_{i,m}^{\text{out}} = q_{i,0} + \sum_{j=1, j \neq m}^{l_i} q_{i,j}^{\text{in}} \quad (2.19)$$

où  $m = 1, 2, \dots, l_i$ ,  $i = 1, 2, \dots, n$  et où, à l'initialisation,  $q_{i,j}^{\text{in}} = 0$ .

Les valeurs  $q_{i,m}^{\text{out}}$  sont assignées aux valeurs correspondantes  $r_{j,\pi(i,m,j)}^{\text{in}}$  selon les connexions dans le graphe biparti. A partir de la règle de la tangente hyperbolique et l'information de syndrome, le LLR envoyé à partir du  $j$ -ème nœud de contrôle le long de la  $m$ -ème branche est

$$\tanh \left( \frac{r_{j,m}^{\text{out}}}{2} \right) = (1 - 2s_j) \prod_{i=1, i \neq m}^{t_j} \tanh \left( \frac{r_{i,m}^{\text{in}}}{2} \right) \quad (2.20)$$

où  $m = 1, 2, \dots, t_j$  et  $j = 1, 2, \dots, n - k$ . Ensuite, les valeurs  $r_{j,\pi(i,m,j)}^{\text{out}}$  sont assignées aux valeurs correspondantes  $q_{i,m}^{\text{in}}$  qui peuvent être utilisées pour commencer une nouvelle itération et estimer  $x_i$  à partir de

$$\hat{x}_i = \begin{cases} 0, & \text{si } q_{i,0} + \sum_{m=1}^{l_i} q_{i,m}^{\text{in}} \geq 0 \\ 1, & \text{sinon.} \end{cases} \quad (2.21)$$

Les résultats de simulation montrent que des codes LDPC réguliers peuvent être plus performants que les turbo codes de Garcia-Frias et Zhao [88] pour des séquences de taille 16384. Ils donnent également de meilleurs résultats que les turbo codes de Aaron et Girod [93] pour des séquences de taille  $10^4$  et  $10^5$  mais seulement à de faible taux de compression. Pour des taux de compression plus élevés, de l'ordre de 4 : 1 et 8 : 1, seuls les codes LDPC irréguliers sont capable de faire mieux. Une extension de cette approche dans le cas où les sources binaires sont corrélées par un champ de Markov caché est présentée dans [95].

## 2.4 Techniques de codage de Wyner-Ziv

Dans cette section, nous allons présenter le codage de source distribué avec pertes (codage de Wyner-Ziv). On considère deux sources corrélées  $X$  et  $Y$  à valeurs continues (en général,  $X$  et  $Y$  sont des sources gaussiennes i.i.d., telles que  $X = Y + N$  où  $X$  et  $N$  sont indépendantes, d'espérances nulles et de variances respectives  $\sigma_X^2$  et  $\sigma_N^2$ ).

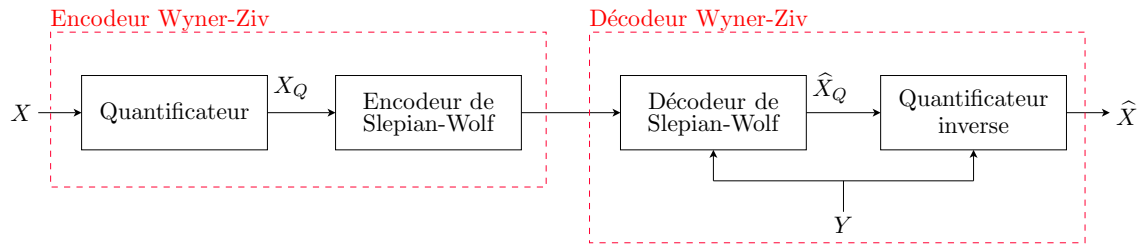


FIGURE 2.13 – Schéma d'un codeur de Wyner-Ziv.

D'un point de vue pratique, un codeur de Wyner-Ziv est constitué d'un quantificateur suivi d'un codeur de Slepian-Wolf comme le montre la figure 2.13. Les codeurs de Wyner-Ziv que nous allons décrire sont basés sur un codage de Slepian-Wolf utilisant des techniques de codage de canal.

### 2.4.1 Solution basée sur DISCUS

Dans [80, 96], Pradhan et Ramchandran présentent un système de codage de Wyner-Ziv. Le schéma du codeur est présenté sur la figure 2.14. Soit  $X$  un bloc de taille  $L$ . Le but est de décoder la meilleure approximation  $\hat{X}$  de  $X$  pour un débit de  $R$  bps. Le problème est posé comme une minimisation de  $R$  sous la contrainte  $\mathbb{E}[\rho(X, \hat{X})] < D$ , où  $\rho(\cdot)$  est une mesure de distorsion quadratique et  $D$  est la distorsion maximum autorisée. Le principe du codeur est le suivant :  $X$  est quantifié, et l'indice du coset contenant le mot de code quantifié est transmis au décodeur. Le décodeur détermine le mot de code  $\hat{X}$  en décodant l'information adjacente  $Y$  dans le coset reçu. Le système de codage est composé de cinq fonctions  $\{M_i\}_{i=1}^5$  comme le montre la figure 2.14 :

- Codage de source ( $M_1, M_2$ ) : L'espace  $\mathbb{R}^L$  est partitionné en  $2^{LR_s}$  régions disjointes, où  $R_s$  est le débit du codage de source :

$$M_1 : \mathbb{R}^L \rightarrow \{1, 2, \dots, 2^{LR_s}\}. \quad (2.22)$$

Soit  $\mathcal{T} = \{\Gamma_1, \Gamma_2, \dots, \Gamma_{2^{LR_s}}\}$  l'ensemble des régions disjointes de  $2^{LR_s}$ . A chaque région est associé un mot de code. La fonction  $M_2$  réalise la correspondance entre les

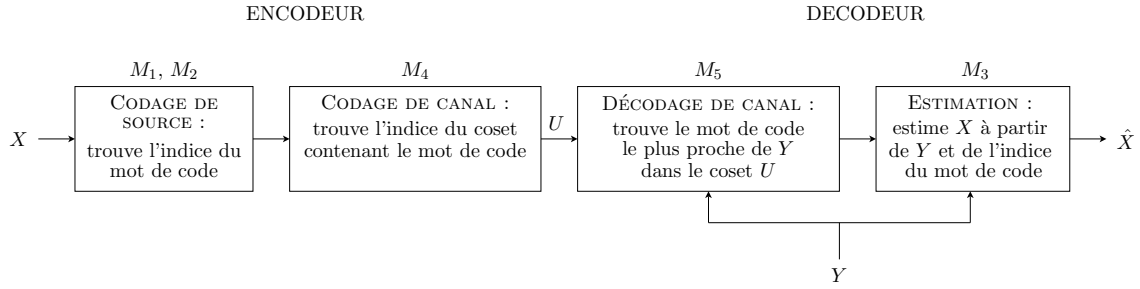


FIGURE 2.14 – Schéma de codage DISCUS.

mots de code et les régions :

$$M_2 : \{1, 2, \dots, 2^{LR_s}\} \rightarrow \mathbb{R}^L. \quad (2.23)$$

- Estimation de  $X$  ( $M_3$ ) : Le décodeur reconstruit le meilleur estimé de  $X$  à partir de l'information adjacente et de la région  $\Gamma_i \in \mathcal{T}$  contenant  $X$  :

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{a} \in \mathbb{R}^L} E [\rho(\mathbf{X}, \mathbf{a}) | \mathbf{X} \in \Gamma_i, \mathbf{Y} = \mathbf{y}] \quad (2.24)$$

Cette estimation est le rôle de la fonction  $M_3$  :

$$M_3 : \mathbb{R}^L \times \{1, 2, \dots, 2^{LR_s}\} \rightarrow \mathbb{R}^L. \quad (2.25)$$

- Codage et décodage de canal ( $M_4, M_5$ ) : A ce stade, le système peut effectuer une transmission de  $R_s$  bps pour garantir une distorsion inférieure à  $D$ . Pour réduire encore davantage le débit, le système propose d'exploiter la corrélation entre  $X$  et  $Y$  au décodeur. Soit  $W$  la version quantifiée de  $X$ . La corrélation entre  $W$  et  $Y$ , caractérisée par la distribution conditionnelle  $P(Y|W)$ , est connue au décodeur. Avec la présence de  $Y$  au décodeur, la débit minimum à transmettre pour reconstruire  $W$  est

$$H(W|Y) = H(W) - I(W; Y). \quad (2.26)$$

Le rôle du codage de canal est de supprimer l'information mutuelle pour ne transmettre que  $H(W|Y)$ . Pour ce faire, on utilise un constructeur de cosets tel que celui présenté dans la section 2.3.2.1.  $M_4$  détermine les indices des cosets contenant les mots de codes :

$$M_4 : \{1, 2, \dots, 2^{LR_s}\} \rightarrow \{1, 2, \dots, 2^{LR}\} \quad (2.27)$$

et transmet cette information à un débit  $R = R_s - R_c$  bps, où  $R_c$  est le débit du codage de canal. Le décodeur retrouve le mot de code à partir de l'indice du coset et de l'information adjacente :

$$M_5 : \mathbb{R}^L \times \{1, 2, \dots, 2^{LR}\} \rightarrow \{1, 2, \dots, 2^{LR_s}\}. \quad (2.28)$$

Dans cette approche, la probabilité d'avoir une erreur de décodage est non nulle, elle dépend de l'efficacité du code de canal utilisé.

Le débit  $R$  et la distorsion  $D$  dépendent de la construction des dictionnaires de codage de source et de codage de canal. Pour une distorsion  $D$  donnée, l'objectif est de minimiser  $R_s$  et maximiser  $R_c$ . On peut résumer le rôle des différentes fonctions ainsi :

- $M_1$  et  $M_3$  : minimiser  $R_s$  sous la contrainte que la distorsion soit inférieure à une certaine valeur.
- $M_2$  : maximiser l'information mutuelle entre  $W$  et  $Y$ ,  $I(W; Y)$ . On choisit comme représentation du mot de code de  $\Gamma_i$ , le vecteur dont  $\Gamma_i$  est la région de Voronoi (la même contrainte est appliquée sur  $M_1$ ).
- $M_4$  : maximiser le débit  $R_c$  du code de canal (et chacun de ses cosets) tel que la probabilité d'erreur soit inférieure à un certain seuil.
- $M_5$  : minimiser la complexité du décodeur.

Ainsi, le vecteur reconstruit  $\hat{\mathbf{X}}$  est obtenu par l'équation suivante :

$$\hat{\mathbf{X}} = M_3[\mathbf{Y}, M_5\{\mathbf{Y}, M_4(M_1(\mathbf{X}))\}] \quad (2.29)$$

pour des vecteurs de source et d'information adjacente  $\mathbf{X}$ ,  $\mathbf{Y}$  respectivement.

Dans [96], le codage de source et le codage de canal sont d'abord étudiés avec des codes sans mémoire, puis avec mémoire. Les meilleures performances sont obtenus avec des codes avec mémoire, au prix d'une plus grande complexité.

## 2.4.2 Solutions basées sur les turbo codes

Dans [93], le système de codage sans perte vu dans la section 2.3.2.2 est proposé comme structure de base d'un codeur de Wyner-Ziv.

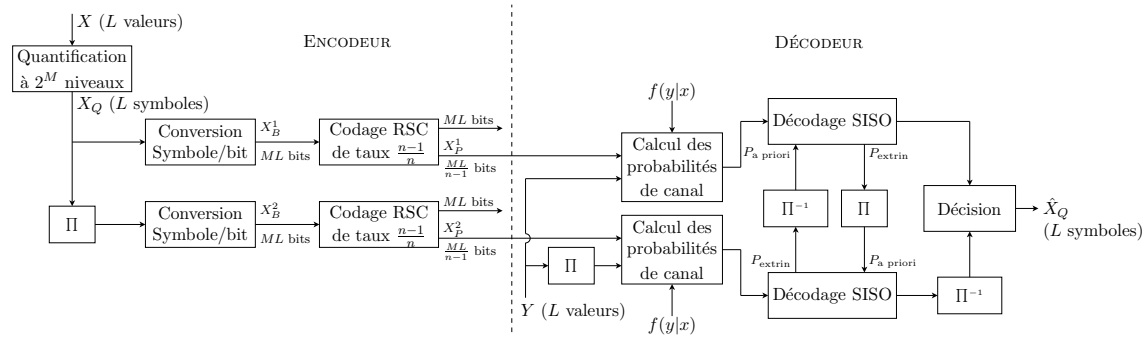


FIGURE 2.15 – Schéma de codage de Wyner-Ziv présenté par Aaron et Girod [93].

Le schéma de codage est présenté sur la figure 2.15. Soit  $X$  une séquence de variables aléatoires continues i.i.d. de taille  $L$  et de densité de probabilité  $f(x)$ . Soit  $Y$  une version bruitée de  $X$  tel que  $Y = X + N$  où  $N$  est une séquence de variables i.i.d. et indépendantes de  $X$ . La corrélation entre  $X$  et  $Y$  est représentée par la densité de probabilité conditionnelle  $f(y|x)$ . Avant le codage de Slepian-Wolf, chaque valeur de  $X$  est quantifiée sur  $2^M$  niveaux, produisant ainsi une séquence de valeurs discrètes  $X_Q$  de taille  $L$ . Seuls les bits de parité sont poinçonnés et transmis au décodeur. Le turbo décodage s'effectue de la même manière que dans le codeur de Slepian-Wolf de la section 2.3.2.2, à l'exception du calcul des probabilités de canal qui désormais représentent une probabilité de symbole (au lieu de bit dans le codeur de Slepian-Wolf). Ici, ce calcul est fait à partir d'une fonction de densité de probabilité gaussienne  $f(y|x)$  et des valeurs extrinsèques qui circulent entre les codeurs SISO.



### 2.4.3 Solutions basées sur les codes LDPC

Dans le schéma de la figure 2.15, les codes de canal utilisés sont des turbo codes poinçonnés. Le poinçonnage s'effectue sur les bits de parité et permet de contrôler le débit au décodeur. Le poinçonnage peut également être appliqué sur les syndromes à la sortie d'un code LDPC mais résulte en de mauvaises performances débit-distorsion. Dans [93], deux méthodes sont proposées pour construire des codes LDPC permettant un meilleur contrôle du débit.

Le premier code proposé est un code LDPC avec accumulation de syndromes (en anglais LDPC Accumulate, LDPCA). Il est constitué d'un code LDPC concaténé avec un accumulateur. Un exemple de code LDPCA est présenté sur la figure 2.16. A chaque noeud de contrôle, les syndromes sont accumulés (modulo 2), produisant des syndromes accumulés. Ces syndromes accumulés sont placés dans une mémoire tampon, en attente d'être expédiés au décodeur. Le décodeur modifie la structure de son graphe biparti chaque fois qu'il reçoit un nouvel incrément de syndromes accumulés.

Par exemple, si le décodeur reçoit tous les syndromes accumulés, il lui suffit de calculer la différence de ces symboles (modulo 2) pour retrouver les syndromes originaux. La modification du graphe n'intervient qu'à des taux de compression plus élevés. Considérons, par exemple, un taux de compression  $\frac{1}{2}$ . Dans l'exemple de la figure 2.16, cela revient à transmettre seulement les syndromes accumulés d'indices paires ( $a_2, a_4, a_6, a_8$ ). La différence (modulo 2) de ces syndromes produit  $(s_1 + s_2, s_3 + s_4, s_5 + s_6, s_7 + s_8)$  à partir de  $(x_1, \dots, x_8)$ . Le graphe sous-jacent maintient les degrés de tous ses noeuds de variable et est ainsi parfaitement adapté à un décodage par propagation de croyance.

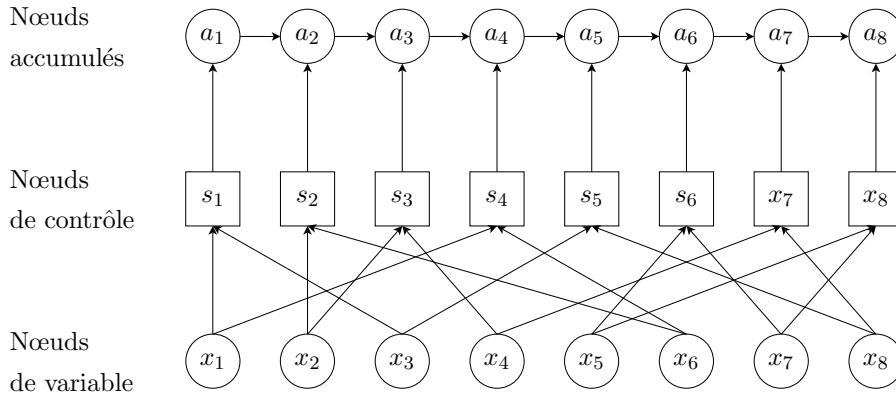


FIGURE 2.16 – Graphe biparti d'un code LDPC avec accumulation de syndromes (LDPCA).

Le deuxième code proposé dans [93] pour contrôler le débit est le code LDPC sommé avec accumulation de syndromes (en anglais Sum LDPC Accumulate, SLDPCA). Il consiste en un additionneur (l'inverse d'un accumulateur) concaténé à un code LDPCA. Si l'algorithme de décodage n'est pas tout à fait le même que pour un code LDPCA, la manière de contrôler le débit est, elle, strictement identique.

Les résultats expérimentaux montrent que les codes LDPCA et SLDPCA irréguliers permettent de mieux s'approcher de la limite de Slepian-Wolf que les turbo codes poinçonnés.

---

## 2.5 Codage vidéo distribué

Ces dernières décennies, la recherche en codage vidéo s'est concentrée sur des systèmes où c'est la tâche de l'encodeur que d'exploiter les statistiques de la source pour la comprimer le plus efficacement possible avant de la transmettre au décodeur. Cette tâche relève du rôle de la décorrélation spatiale à l'intérieur des trames et de la prédiction de mouvement entre les trames. Alors que la première opération est relativement simple, la deuxième augmente de façon considérable la complexité de l'encodeur. Le codage vidéo distribué adopte un paradigme de codage complètement différent, en donnant au décodeur la tâche d'exploiter, partiellement ou totalement, les statistiques de la source pour la comprimer efficacement. Ce changement de paradigme déplace la complexité de l'encodeur vers le décodeur, permettant des applications nouvelles.

La recherche dans le domaine du codage vidéo distribué est motivée par des applications telles que des réseaux de caméra de surveillance sans fil à faible consommation, des réseaux de capteurs, l'acquisition d'images multi-vues, etc, c'est-à-dire des applications dans lesquelles une faible complexité de l'encodeur est nécessaire pour des raisons qui peuvent être une faible capacité de mémoire, une faible puissance de calcul, ou un contrôle strict de la consommation d'énergie.

Afin d'exploiter la corrélation contenue dans les séquences vidéo au décodeur, les codeurs vidéo distribué utilisent des méthodes de codage de source distribué telles que celles que nous avons vues jusqu'ici. Le principe général consiste à coder séparément les données vidéo corrélées et à les décoder conjointement. Par exemple, les trames successives d'une séquence vidéo peuvent être codées séparément et décoder conjointement, le décodeur se chargeant alors d'effectuer l'estimation de mouvement entre les trames. Pour atteindre des performances comparables à celle d'un codeur vidéo traditionnel, il est nécessaire pour le décodeur d'utiliser des techniques d'estimation de mouvement complexe, ce qui suppose que le récepteur possède une puissance suffisante.

Witsenhausen et Wyner sont les premiers à avoir évoqué le principe de codage vidéo distribué en 1978 [97]. Cependant, le système qu'ils proposent à l'époque (proche de celui de Stanford que nous verrons dans la section 2.5.2) n'a jamais été implémenté. C'est seulement plus de vingt ans plus tard qu'apparaît le premier schéma pratique de codage vidéo distribué. Les deux architectures principales de codage vidéo distribué proposées dans la littérature sont PRISM développée par l'université de Berkeley [98] et celle développée par l'université de Stanford [99]. Les deux approches sont foncièrement différentes, la première reposant sur un codage à bloc sans voie de retour, la deuxième sur un codage par trame avec voie de retour. Elles sont décrites dans les prochaines sections.

### 2.5.1 Codage vidéo distribué sans voie de retour : PRISM

Le premier codeur vidéo distribué proposé dans la littérature est PRISM. PRISM est l'acronyme de Power-efficient, Robust, hIgh-compression, Syndrome-based Multimedia. Il a été présenté pour la première fois en 1999 par Puri et Ramchandran [98]. Ce codeur est caractérisé par le fait qu'il ne nécessite pas de voie de retour entre le décodeur et l'encodeur. Le codage s'effectue par bloc. Ainsi, chaque trame de la séquence vidéo est divisée en blocs de taille  $8 \times 8$  disjoints (les blocs ne se chevauchent pas). Un classificateur choisit le mode d'encodage de chaque bloc : un bloc peut ne pas être codé (mode `skip`), être codé en Intra (mode `intra`) ou être codé en Wyner-Ziv (mode `inter`). La version la plus récente du codeur est celle décrite dans [100]. C'est cette dernière que nous allons présenter.

---

### 2.5.1.1 Architecture de l'encodeur

On commence par décrire l'architecture de l'encodeur (présentée sur la figure 2.17). Les étapes d'encodage sont les suivantes :

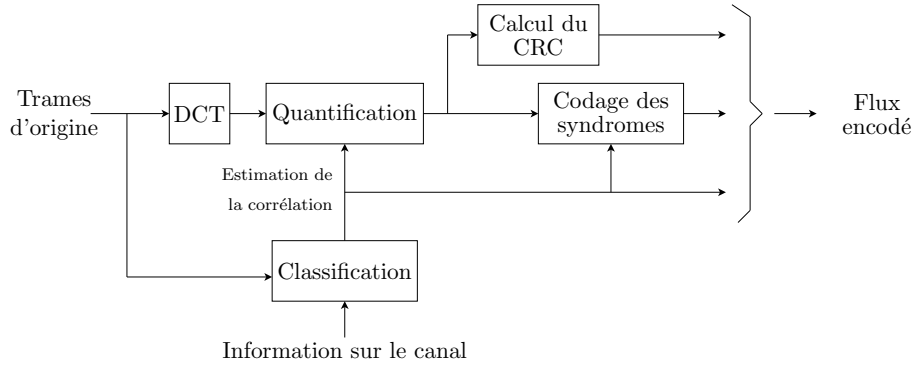


FIGURE 2.17 – Architecture de l'encodeur PRISM [100].

- A. Transformée spatiale :** On commence par appliquer une DTC sur le bloc source de façon à approximer une transformée de Karhunen-Loeve (KL) de l'innovation représentée par le bruit de corrélation entre le vecteur source et son information adjacente. Les coefficients transformés  $\mathbf{X}$  sont alors ordonnés sur un plan unidimensionnel (de 64 coefficients) en effectuant un balayage en zig-zag sur le bloc 2-D (de taille  $8 \times 8$ ).
- B. Quantification :** Les coefficients DCT sont quantifiés par un quantificateur scalaire. Le pas de quantification est choisi en fonction de la qualité de la reconstruction désirée.
- C. Classification :** Chaque bloc est décomposé en plans de bits comme le montre la figure 2.18 pour un bloc de 64 coefficients. Dans cette figure, les bits correspondants à des cases de couleur blanche peuvent être prédits à partir de l'information adjacente au décodeur, ils ne sont donc pas codés. Les bits restants (en gris et en noir sur la figure) ne peuvent pas être prédits au décodeur et nécessitent d'être encodés. Pour choisir quels bits seront codés, on utilise un classificateur pour estimer le bruit de quantification entre chaque bloc et son prédicteur temporel. Ce dernier est obtenu en mesurant le nombre de bits les plus significatifs qu'il est possible de prédire au décodeur pour chaque coefficient quantifié. Le choix du classificateur dépend du degré de complexité autorisé au décodeur et des conditions du canal de transmission :
  - Si aucune estimation de mouvement n'est possible, l'encodeur utilise le résidu entre le bloc courant (dans le domaine pixel) et le bloc correspondant dans la trame précédente pour évaluer le bruit de corrélation. La classe du bloc est déterminée par un seuillage de l'erreur quadratique entre le bloc courant et le bloc de la trame précédente. Selon la valeur du seuil, le bloc peut appartenir à 16 classes différentes numérotées de 0 à 15. A chaque classe est associée un bruit de corrélation dont la statistique a été obtenue préalablement par un entraînement de l'algorithme. Le bruit de corrélation détermine le nombre de plans de bits les moins significatifs pour chaque coefficient. Le bruit de corrélation étant sur-estimé (du fait de l'absence d'estimation de mouvement), le nombre de plans de bits à transmettre l'est aussi. C'est pourquoi, les plans de bits les plus significatifs sont codés par un code de canal basé sur les cosets (les plans de bits de couleur grise sur la figure 2.18), tandis que les bits

restants sont codés par un codeur entropique sans prédicteur (les plans de bits de couleur noire sur la figure 2.18).

- Si au contraire, une estimation de mouvement complexe est permise à l’encodeur, le classificateur utilise le bruit de corrélation entre le bloc courant et son meilleur prédicteur temporel compensé en mouvement pour déterminer le nombre de bits les moins significatifs qu’il sera nécessaire de transmettre au décodeur. Ces plans de bits sont codés par un codeur entropique. En utilisant une estimation de mouvement complexe, on élimine toutes les incertitudes sur le nombre de bits qu’il sera possible de prédire au décodeur, c’est pourquoi aucun bit n’est codé par le code de canal (pas de plans de bits de couleur grise sur la figure 2.18).

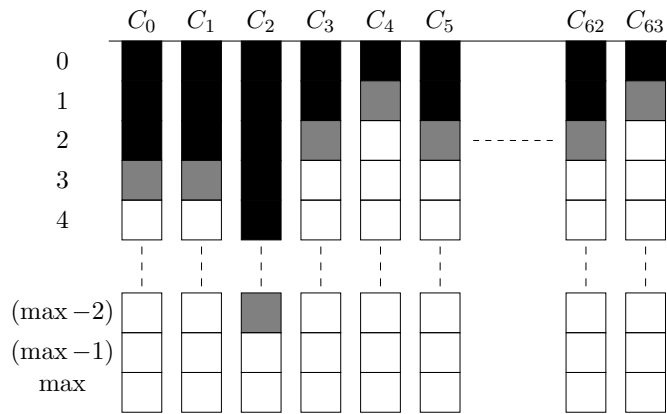


FIGURE 2.18 – Plans de bit d’un bloc de 64 coefficients. Le plan de bit 0 correspond au bit le moins significatif. Chaque couleur correspond à un codage différent : pas de codage (blanc), codage de Wyner-Ziv (gris), codage entropique (noir).

**D. Codage de syndromes :** Les bits les moins significatifs de chaque bloc, déterminés par le classificateur, sont codés suivant leur couleur : les blocs noirs dans la figure 2.18 sont codés par un codeur entropique, les blocs gris sont codés par un code de canal basé sur les cosets. Les deux types de codage sont les suivants :

- **Codage entropique :** La première étape est la construction d’un alphabet de codage entropique qui consiste en un quadruplet (**run, depth, path, last**). **run** indique le nombre de coefficients précédent le coefficient courant pour lesquels aucun plan de bits n’a été encodé, **depth** indique le nombre de plans de bits non significatifs encodés pour le coefficient courant, **path** indique le chemin dans l’arbre binaire qui spécifie le coset contenant le coefficient courant. Enfin, **last** indique si le coefficient non nul est le dernier dans le bloc. Une fois l’alphabet construit, on utilise un codeur arithmétique pour coder l’information de syndrome.
- **Code de canal basé sur les cosets :** Le syndrome est calculé à partir d’une matrice de parité d’un code correcteur d’erreur linéaire  $(n, k_i)$  :  $\mathbf{s}_i = H_i \mathbf{b}_i$  où  $\mathbf{s}_i$  et  $H_i$  représentent le syndrome et la matrice de parité et  $\mathbf{b}_i$  représente les bits à coder. Le taux d’encodage est  $\frac{n-k_i}{n}$  bits par coefficient. Comme la taille des blocs est petite (64 échantillons pour un bloc  $8 \times 8$ ), plutôt que d’utiliser des codes LDPC ou des turbo codes, le code utilisé est un code BCH (Bose, Ray-Chaudhuri, Hocquenghem) [101]. Le paramètre  $k_i$  associé au  $i$ -ème code de canal dépend de la classe  $i$  à laquelle le bloc appartient.

**E. Génération d'un CRC :** Un contrôle de redondance cyclique (en anglais Cyclic Redundancy Check, CRC) est utilisé pour générer une signature, un CRC, pour chaque séquence de mots de code quantifiés. Ce CRC permet au décodeur de vérifier si la séquence a bien été décodée.

### 2.5.1.2 Architecture du décodeur

L'architecture du décodeur de PRISM est présentée sur la figure 2.19. Les différentes étapes de décodage sont les suivantes :

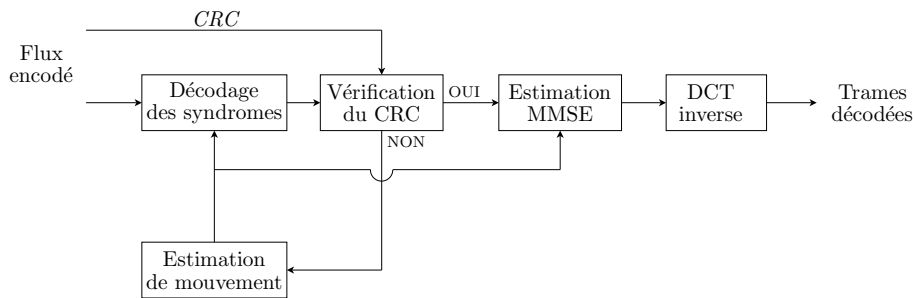


FIGURE 2.19 – Architecture du décodeur de PRISM [100].

- A. Génération de l'information adjacente :** Le décodeur réalise une estimation de mouvement pour générer les prédicteurs candidats au décodage des mots de code quantifiés représentés par leur syndrome.
- B. Décodage des syndromes :** L'information adjacente  $Y$  est un prédicteur candidat obtenu par le générateur d'information adjacente. Le décodage du syndrome se fait en deux étapes. Dans une première étape, les bits codés entropiquement (les plans de bits de couleur noir dans la figure 2.18) sont décodés. Si aucun bit n'a été codé par le code de canal (les plans de bits de couleur grise dans la figure 2.18), alors les bits décodés entropiquement identifient de façon unique le coset dans lequel l'information adjacente  $Y$  doit être décodée. Si, au contraire, un plan de bit a été codé par le code de canal, c'est la combinaison du coset de ce plan de bit et des bits décodées entropiquement qui spécifie le coset dans lequel  $Y$  doit être décodée. Une fois l'identification du coset effectuée, la deuxième étape est le décodage qui consiste à trouver le mot de code le plus proche de  $Y$  dans un coset spécifique. Ce décodage souple est basée sur la méthode de [102].
- C. Vérification du CRC :** Une fois le décodage des syndromes effectué, un CRC est calculé sur le bloc décodé. Si le CRC calculé ne correspond pas à celui transmis par l'encodeur, l'algorithme revient à l'étape de génération de l'information adjacente et un nouveau prédicteur candidat est utilisé.
- D. Estimation et reconstruction :** Une fois la séquence de mots de code décodée, la séquence de coefficients transformés est reconstruite en choisissant l'estimé qui minimise l'erreur quadratique à partir du prédicteur et du mot de code quantifié.
- E. Transformée inverse :** Une fois tous les coefficients reconstruits, l'opération de balayage en zig-zag faite à l'encodeur est inversée pour obtenir un bloc 2-D de coefficients reconstruits. Les coefficients transformés sont ensuite inversés par la transformée DCT inverse pour obtenir les valeurs des pixels du bloc décodé.

Comme on a pu le voir, PRISM a la particularité de proposer une architecture flexible qui lui permet de contrôler le compromis entre la complexité de l'encodeur et celle du décodeur. Cette flexibilité permet notamment d'envisager l'utilisation à l'encodeur de méthodes d'estimation de mouvement rapide telle que celle proposée dans [103].

## 2.5.2 Codage vidéo distribué avec voie de retour : le codeur de Stanford

Alors que le codeur PRISM venait tout juste d'être présenté par l'université de Berkeley, une autre architecture de codage vidéo distribué, développée cette fois par l'université de Stanford fut proposée par Aaron, Zhang et Girod [99]. A l'époque, le codage était fait dans le domaine pixel [99, 104]. Puis sont apparus les premières implémentations dans le domaine transformé [105, 106]. L'utilisation d'une transformée spatiale à l'encodeur permet de tirer au mieux parti de la corrélation spatiale à l'intérieur des trames. Nous allons seulement décrire le codeur dans le domaine transformé, qui est devenu la référence dans la littérature.

A la différence de PRISM, l'architecture de Stanford nécessite l'utilisation d'une voie de retour entre le décodeur et l'encodeur. La voie de retour permet de mieux contrôler le débit et donc d'offrir de meilleures performances débit-distorsion. En contrepartie, elle augmente la complexité du décodeur car elle oblige ce dernier à effectuer un décodage complet après chaque nouvelle requête. Elle restreint également l'utilisation du codeur à des applications qui ne nécessitent pas de décodage en temps réel.

L'architecture du codeur de Stanford est présentée sur la figure 2.20. Le codage est fait par trame et non par bloc comme c'est le cas dans PRISM.

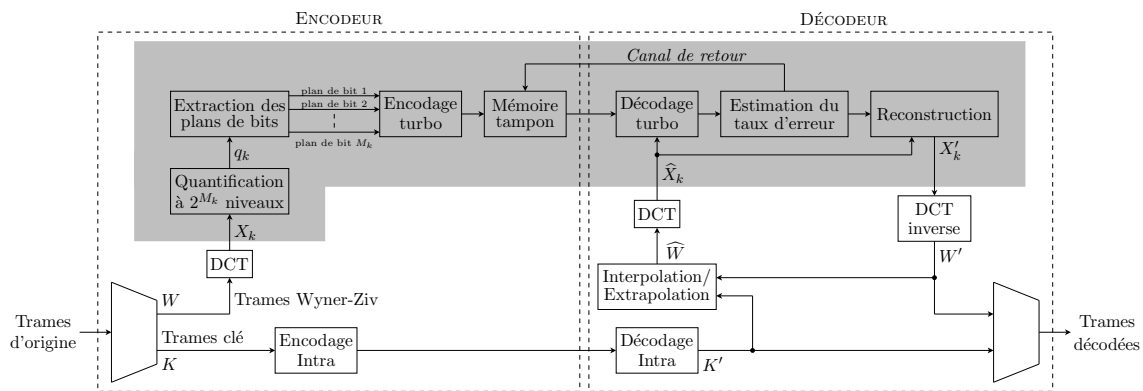


FIGURE 2.20 – Architecture du codeur de Stanford [105]. La partie grisée correspond à un codage/décodage par bande de fréquence  $k$ .

### 2.5.2.1 Architecture de l'encodeur

Le codeur de Stanford comprend les étapes suivantes :

- A. **Classification des trames** : Les trames sont organisées par GOP (de l'anglais Group Of Pictures). La première trame de chaque GOP est une trame codée par un codeur conventionnel en mode Intra et le reste des trames sont codées en utilisant un codeur de Wyner-Ziv. Les trames codées en mode Intra sont appelées les trames clé (K) (en anglais key frame), elles sont codées et décodées séparément. Les trames codées en Wyner-Ziv (W) (en anglais Wyner-Ziv frame) sont, elles, codées séparément et décodées à l'aide d'une information adjacente obtenue par interpolation ou extrapolation de ses trames

voisines. Par exemple, un GOP de taille 2 comprendra une trame K et une trame W. Dans ce cas, une trame sur deux, par exemple les trames paires seront des trames clé et le reste des trames (les trames impaires) seront des trames codées en Wyner-Ziv.

- B. Codage Intra :** Les trames clé de chaque GOP sont codées par un codeur conventionnel en mode Intra, tels qu'H.263 ou H.264/AVC (Advanced Video Coding). Le codage Intra se veut relativement simple pour éviter d'augmenter la complexité de l'encodeur. Ce flux d'information est directement envoyé au décodeur.
- C. Transformée spatiale :** Les trames codées en Wyner-Ziv sont transformées spatialement par une DCT  $4 \times 4$ . Les coefficients sont organisés par bande. La bande 0, dont les coefficients sont situés dans le coin gauche de chaque bloc  $4 \times 4$ , est appelée la bande DC. Elle contient l'information de plus basse fréquence. Les autres bandes sont appelées les bandes AC, elles contiennent les informations de plus haute fréquence. Prenons l'exemple d'une séquence QCIF, c'est-à-dire une séquence dont la résolution de chaque trame est  $176 \times 144$ . Dans cet exemple, la DCT  $4 \times 4$  transforme une trame QCIF en 16 bandes de coefficients de taille 1584.
- D. Quantification :** Les coefficients de chaque bande sont quantifiés par un quantificateur uniforme à  $2^{M_k}$  niveaux,  $k = 1, \dots, 16$ . Le quantificateur utilisé dépend de la qualité désirée, il est choisi de façon à ce que la qualité des trames codées en Wyner-Ziv et des trames clé soit sensiblement la même afin d'éviter tout effet de scintillement.
- E. Extraction des plans de bits :** Les coefficients quantifiés  $q_k$  sont convertis en mots de code binaire à longueur fixe, et les bits de même niveau sont regroupés pour former  $M_k$  plans de bits. Chaque plan de bit est ensuite envoyé à un codeur de Slepian-Wolf.
- F. Turbo codage :** Chaque plan de bits est codé par un turbo code poinçonné. Seuls les bits de parité sont placés dans une mémoire tampon, en attente d'être demandés par le décodeur, les bits systématiques sont simplement ignorés.

### 2.5.2.2 Architecture du décodeur

- A. Décodage Intra :** Les trames clé sont décodées séparément et mises de côté, en attente d'être utilisées pour générer l'information adjacente.
- B. Génération de l'information adjacente :** Le décodeur utilise les trames voisines précédemment décodées de  $W$  pour générer une information adjacente  $\hat{W}$ , qui est un estimé de  $W$ . Deux approches pour générer cette information sont envisagées. La première consiste à extrapoler l'information à partir des deux trames qui précèdent la trame  $W$ . Par exemple, pour un GOP de taille 2, si la trame  $W$  que l'on cherche à décoder se situe au temps  $t$ , l'estimation de mouvement sera effectuée entre la trame  $W$  au temps  $t - 2$  et la trame  $K$  au temps  $t - 1$ . La deuxième approche consiste à utiliser la trame clé la plus proche à gauche et celle la plus proche à droite et faire une interpolation compensée en mouvement entre les deux. Par exemple, pour un GOP de taille 2, l'information adjacente pour une trame  $W$  située au temps  $t$  sera obtenue par interpolation des trames clé situées au temps  $t - 1$  et  $t + 1$ . La première approche a l'avantage de proposer un décodage séquentiel, cependant c'est la deuxième qui est le plus souvent envisagée car l'information adjacente ainsi générée ressemble davantage à la trame  $W$ . Parmi toutes les techniques d'interpolation, la plus populaire dans les codeurs vidéo distribué de ce type est l'algorithme de mise en correspondance par blocs (en anglais block-matching) compensé en mouvement. La même transformée DCT que celle utilisée à l'encodeur sur les trames codées en Wyner-Ziv est appliquée sur  $\hat{W}$ .

- C. Turbo décodage :** Pour une bande donnée, le turbo décodeur décode successivement chaque plan de bits en commençant par le plan de bits le plus significatif. Le décodage s'effectue avec les bits de parité envoyés par le codeur et l'information adjacente  $\hat{X}_k$ . Si le nombre de bits de parité n'est pas suffisant pour décoder le plan de bits, le décodeur demande à l'encodeur de lui en envoyer davantage. Après chaque nouvelle requête, un turbo décodage complet est effectué. Le processus de «décodage-requête» se répète jusqu'à ce que la probabilité d'erreur soit inférieure à un certain seuil (par exemple,  $10^{-3}$ ). L'estimation du taux d'erreur dans [105] s'effectue directement sur les bits de la trame d'origine, ce qui n'est pas possible en pratique. Pour utiliser efficacement l'information adjacente, le décodeur doit connaître le bruit de corrélation entre les bandes de la trame codée en Wyner-Ziv et celles de l'information adjacente ( $d_k = X_k - \hat{X}_k$ ). Ce bruit de corrélation correspond au résidu entre les bandes de coefficients DCT. Dans le codeur de Stanford [105], la distribution du résidu  $d_k$  est approximée par une Laplacienne :  $f(d_k) = \frac{\alpha}{2} e^{-\alpha|d_k|}$ . Les paramètres  $\alpha$  sont approximés hors ligne à partir des histogrammes de plusieurs séquences vidéo.
- D. Reconstruction :** Lorsque tous les plans de bits d'une bande sont décodés, ils sont regroupés en un ensemble de symboles quantifiés  $q'_k$ . La bande de coefficients reconstruits  $X'_k$  est calculée à partir de  $q'_k$  et de l'information adjacente  $\hat{X}_k$  :  $X'_k = \mathbb{E}[X_k|q'_k, \hat{X}_k]$  en supposant  $q'_k$  décodé sans erreur. Si, aucun bit n'a été envoyé pour la bande  $k$  ( $2^{M_k} = 0$ ),  $X'_k$  est remplacé par l'information adjacente  $\hat{X}_k$ .
- E. Transformée inverse :** Une fois toutes les bandes décodées, une transformée DCT inverse est appliquée sur les coefficients pour générer la trame Wyner-Ziv décodée  $W'$ .
- F. Regroupement des trames :** Les trames clé précédemment décodées  $K'$  et les trames Wyner-Ziv fraîchement décodées  $W'$  sont regroupées dans une même séquence pour former la séquence vidéo décodée.

Le codeur de Stanford est à la base de nombreux codeurs vidéo distribué, notamment celui du projet DISCOVER [107] qui a servi d'architecture de base pour les contributions de cette thèse.

### 2.5.3 Derniers développements

L'architecture de Stanford étant celle que nous avons retenue pour le travail de cette thèse, nous ne décrivons que les améliorations les plus récentes concernant ce codeur.

Comme nous l'avons dit dans la section 2.5.2, l'estimation du taux d'erreur à la sortie du turbo décodeur dans le codeur de Stanford s'effectue en comparant les bits décodés avec ceux que l'on cherche à décoder. Cette situation est évidemment impossible en réalité, car si les bits étaient déjà connus aux décodeurs, on n'aurait pas besoin de les décoder. C'est pourquoi plusieurs solutions ont été proposées pour résoudre ce problème [108, 109]. Dans [108], la mesure de confiance est basée sur le rapport de vraisemblance défini par :

$$\Lambda_i = \log \frac{Pr(X_i = 1|S_i)}{Pr(X_i = 0|S_i)} \quad (2.30)$$

où  $X_i$  est le bit à décoder à l'instant  $i$  et  $S_i$  est l'information adjacente. Pour chaque instant  $i$ , si la valeur absolue du rapport de vraisemblance ( $|\Lambda_i|$ ) est inférieure à un seuil de confiance (fixé à 4.6), alors le bit  $X_i$  est considéré comme étant incorrectement décodé. A la fin du décodage, la probabilité d'erreur (en anglais Bit Error Rate, BER) est estimée par le nombre de bits incorrectement décodés divisé par le nombre de bits total. Si le BER



est supérieur à un seuil (fixé à  $10^{-3}$ ), le décodage est considéré comme étant un échec et davantage de bits de parité sont demandés à l'encodeur. C'est cette approche d'estimation du taux d'erreur qui a été retenue dans le codeur du projet DISCOVER.

L'utilisation d'une voie de retour est une contrainte importante du codeur de Stanford, sa présence implique une utilisation abusive de la bande passante, de la latence et un accroissement de la complexité au décodeur. Des efforts pour réduire l'utilisation de la voie de retour ont été faits [110, 111]. Dans [110], les auteurs proposent une solution permettant de supprimer complètement la voie de retour en la remplaçant par un contrôle de débit au niveau de l'encodeur basé sur une estimation de la corrélation entre les trames codées en Wyner-Ziv et l'information adjacente. L'estimation utilise un algorithme d'estimation de mouvement simplifié pour éviter d'augmenter la complexité de l'encodeur.

Le choix du modèle de corrélation du résidu entre les bandes de coefficients DCT des trames codées en Wyner-Ziv et celles de l'information adjacente est très important en codage vidéo distribué. Si le modèle utilisé ne correspond pas à la distribution du résidu, le codage sera moins efficace. Dans le contexte de la figure 2.20, cela se traduit par une plus faible précision à l'entrée du turbo décodeur, et oblige ce dernier à réclamer davantage de bits de parité. Dans le codeur de Stanford, l'estimation s'effectue «hors ligne», en supposant que l'information adjacente est disponible à l'encodeur ou que la trame codée en Wyner-Ziv l'est au décodeur. Dans [112], les auteurs proposent une méthode plus réaliste pour effectuer une estimation du modèle du bruit de corrélation «en ligne», au moment du décodage. L'approche proposée consiste à estimer la variance  $\sigma_b^2$  de la distribution laplacienne du bruit de corrélation pour la bande  $b$  d'une trame codée en Wyner-Ziv située au temps  $t$  à partir des trames clé utilisées pour interpoler l'information adjacente. La première étape consiste à générer une trame résiduelle :

$$R(x, y) = \frac{X_F(x + dx_f, y + dy_f) - X_B(x + dx_b, y + dy_b)}{2} \quad (2.31)$$

où  $X_B(x + dx_b, y + dy_b)$  et  $X_F(x + dx_f, y + dy_f)$  représentent respectivement les trames compensées aux instants  $(t - 1)$  et  $(t + 1)$ .  $(x, y)$  correspond à la position des pixels dans la trame  $R$ .  $(dx_f, dy_f)$  et  $(dx_b, dy_b)$  représentent les vecteurs de mouvement pour les trames  $X_F$  et  $X_B$ . On applique ensuite une DCT  $4 \times 4$  sur la trame résiduelle pour obtenir une trame de coefficients DCT,  $T$ . Enfin, la variance  $\hat{\sigma}_b^2$  est calculée à partir de  $|T|$ , les valeurs absolues des coefficients de la trame  $T$  :

$$\hat{\sigma}_b^2 = E[T^2] - (E[|T|])^2 \quad (2.32)$$

$\hat{\sigma}_b^2$  est utilisée pour estimer la variance du résidu entre les bandes des trames codées en Wyner-Ziv et celles de l'information adjacente. Pour améliorer les performances, une plus fine granularité peut être obtenue en calculant un paramètre pour chaque coefficient DCT.

La prédiction joue un rôle majeur en codage vidéo traditionnel. Dans un tel système, le nombre de trames prédites à partir des trames voisines est obtenu par estimation du mouvement dans la séquence. Cette opération, très complexe, ne peut pas être utilisée dans un codeur vidéo distribué. C'est pourquoi, dans le codeur de Stanford, la taille du GOP est choisi manuellement et reste la même pour toutes les trames de la séquence. Dans [113], une méthode permettant d'ajuster la taille du GOP est proposée. Le choix du type d'encodage pour une trame est obtenu à partir de métriques simples calculés à partir de la trame courante et de la trame précédente. L'idée générale consiste à mesurer la corrélation entre les trames pour en déduire la quantité de mouvement : plus les trames sont corrélées, plus la quantité de mouvement est considérée faible. Les résultats expérimentaux dans [113]

montrent des gains en PSNR pouvant atteindre jusqu'à 0.4 dB en exploitant la corrélation temporelle à l'encodeur. Une alternative à cette méthode est présentée dans [114] dans le domaine pixel. Ici, l'encodeur choisit la taille de GOP qui maximise le rapport débit-distorsion moyen. Le rapport débit-distorsion est calculé pour chaque trame du GOP en utilisant une approximation de son information adjacente obtenue par interpolation sans estimation de mouvement. L'estimation du débit de chaque trame est donnée par la multiplication de la borne inférieure théorique par un paramètre qui dépend du niveau de quantification choisi. Cet algorithme permet également de se passer de la voie de retour en faisant le choix d'utiliser le débit estimé pour transmettre les trames codées en Wyner-Ziv.

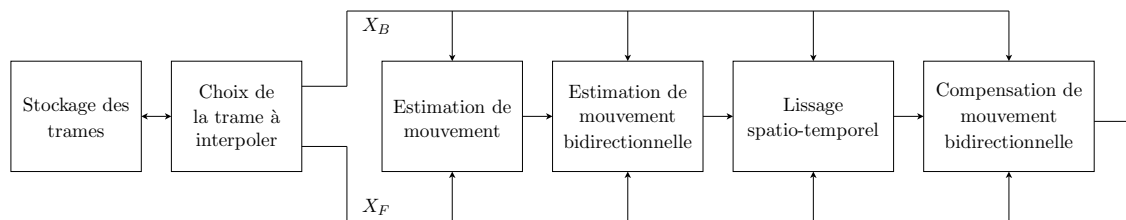


FIGURE 2.21 – Architecture de l'interpolateur de trame proposé dans (Extrait de [113]).

Dans le codeur de Stanford, un algorithme d'interpolation est utilisé au décodeur pour générer l'information adjacente. Le choix de la technique d'interpolation peut avoir une grande influence sur les performances du système. Plus la corrélation entre l'information adjacente et la trame à décoder est grande, moins le décodeur aura à demander de bits de parité supplémentaires à l'encodeur pour atteindre une certaine qualité de reconstruction. Une méthode de génération d'information adjacente basée sur une interpolation compensée en mouvement est proposée dans [115] pour un GOP de taille 2. Cette approche est étendue dans [113] aux GOP de taille supérieure à 2. L'architecture d'interpolation proposée dans [113] est illustrée sur la figure 2.21. Les modules sont classés dans l'ordre de leur utilisation dans l'algorithme :

1. Choix de la trame à interpoler : Ce module choisit la prochaine trame à interpoler parmi toutes les trames non décodées dans le GOP courant. Il choisit deux trames,  $X_B$  et  $X_F$ , déjà décodées telles que l'intervalle de temps entre les deux ne contiennent que des trames non décodées et telles que la distance qui les sépare est la plus grande. La trame à interpoler est simplement celle qui se trouve à la moitié de la distance entre  $X_B$  et  $X_F$ . Une fois cette trame décodée, le module désigne une nouvelle trame à interpoler jusqu'à ce que toutes les trames soient décodées.
2. Estimation de mouvement : Un algorithme de mise en correspondance par blocs est utilisé pour estimer le mouvement entre les trames  $X_B$  et  $X_F$ . Après obtention du champ de mouvement entre les deux trames, le module choisit pour chaque bloc interpolé, le vecteur de mouvement dont l'intersection avec le bloc est la plus proche du centre.
3. Estimation de mouvement bidirectionnelle : Cette étape a pour but d'augmenter la précision des vecteurs de mouvement obtenus par le module d'estimation de mouvement en utilisant une estimation de mouvement bidirectionnelle.
4. Lissage spatio-temporel : Pour supprimer les incohérences spatiales des vecteurs de mouvement, ce module supprime les faux vecteurs de mouvement, c'est-à-dire ceux qui ne correspondent pas au vrai champ de mouvement. Pour cela, il utilise des filtres médian pondérés [116] sur les vecteurs. Ces filtres maintiennent la cohérence spatiale

dans le champ de mouvement en cherchant, pour chaque bloc, des vecteurs candidats dans les blocs voisins. Le choix de substituer un vecteur de mouvement par un vecteur voisin se fait suivant l'erreur de prédiction et les propriétés spatiales du champ de mouvement.

5. Compensation de mouvement bidirectionnelle : Enfin, la trame interpolée est générée par compensation de mouvement bidirectionnelle. L'interpolation linéaire est pondérée par la distance entre les trames de référence et la trame interpolée.

## 2.6 Conclusion

Dans ce chapitre, nous avons présenté le problème de codage de source distribué. Après une description du problème et des limites théoriques, nous nous sommes focalisés sur le problème de codage de source avec information adjacente. Nous avons présenté des solutions pour le codage sans perte, certaines reposant sur un codage de source, d'autres sur un codage de canal. Les solutions de codage avec pertes ont également été étudiées.

Enfin, nous avons décrit une application du codage de source distribué : le codage vidéo distribué mono-vue. Les deux principales architectures de codage vidéo distribué ont été présentées en détail. La première, PRISM, utilise un codage en bloc où un classificateur détermine le codage de chaque bloc. Ce système a l'avantage de ne pas utiliser de voie de retour et permet un contrôle flexible du compromis entre la complexité de l'encodeur et celle du décodeur. La deuxième architecture de référence est celle de Stanford. Contrairement à PRISM, le codage ici se fait par trame. Certaines trames de la séquence sont codées par un codeur conventionnel en mode Intra, les autres étant codées par un codeur de Wyner-Ziv. Cette architecture nécessite une voie de retour pour contrôler le débit des trames codées en Wyner-Ziv. Cette voie de retour introduit une latence et une utilisation plus élevée de la bande passante mais permet d'atteindre de meilleures performance débit-distorsion.

Les performances des schémas de codage vidéo distribué sont supérieures à celles des codeurs conventionnels en mode Intra, tels qu'H.264/AVC en mode Intra lorsque le mouvement dans la séquence est faible (voir par exemple la figure 2.22). Par contre, si la séquence contient beaucoup de mouvement, les schémas de codage vidéo distribué peuvent s'avérer moins bons qu'un codeur Intra (voir par exemple la figure 2.23). Comme on peut s'y attendre, les codeurs vidéo distribué sont systématiquement moins performants que les codeurs Inter, du fait de la connaissance parfaite à l'encodeur de ces derniers des vecteurs de mouvement entre les trames, ce qui n'est pas le cas du décodeur en codage vidéo distribué qui ne fait qu'estimer le champ de mouvement. Il reste donc encore beaucoup de recherche à effectuer dans ce domaine pour que le codage vidéo distribué rattrape les performances des codeurs inter. Cependant, cette technologie étant encore à ses débuts, on peut penser qu'elle arrivera à combler une partie de son retard dans les années à venir.

Une propriété du codage vidéo distribué qui n'a pas été évoquée jusqu'ici est sa résistance naturelle face aux erreurs de transmission. En effet, les pertes induites par le bruit de canal peuvent être corrigées par le code de canal utilisé pour transmettre l'information à condition que ce bruit (comprenant le bruit de corrélation et le bruit de transmission) ne dépasse pas le pouvoir de correction du code. Ceci dit, cette résistance a tout de même un coût qui correspond à l'augmentation du débit dans le cas de pertes. La robustesse en codage vidéo distribué est un des sujets de la prochaine partie.

---

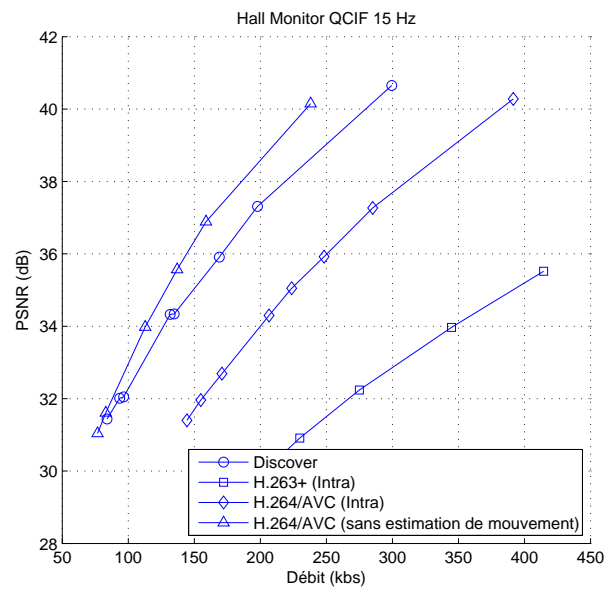


FIGURE 2.22 – Comparaison des performances débit-distorsion du codeur vidéo distribué du projet DISCOVER [107], de H.263+, de H.264/AVC en mode Intra et de H.264/AVC sans estimation de mouvement (Hall Monitor, QCIF 15 fps) (Extrait de [117]).

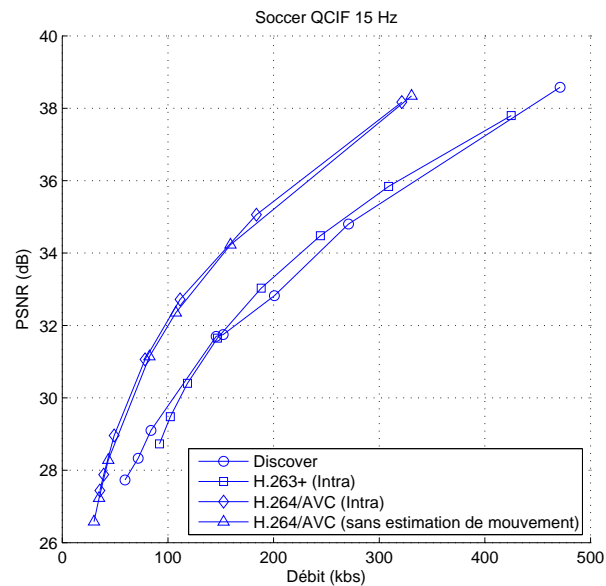


FIGURE 2.23 – Comparaison des performances débit-distorsion du codeur vidéo distribué du projet DISCOVER [107], de H.263+, de H.264/AVC en mode Intra et de H.264/AVC sans estimation de mouvement (Soccer, QCIF 15 fps) (Extrait de [117]).



Deuxième partie  
Contributions

---



## Chapitre 3

# Avant-propos

*Dans ce chapitre, nous allons explorer les champs d'investigation en codage vidéo robuste à base de codage de source distribué et parfois de codage par descriptions multiples. Nous commencerons par présenter les techniques utilisant le codage de Wyner-Ziv pour protéger une transmission vidéo sur des canaux bruités. Ensuite, nous verrons le codage de Wyner-Ziv par couches qui permet d'atténuer le problème de propagation d'erreurs dans les codeurs DPCM. Puis, nous introduirons le problème de codage de source distribué robuste né de la greffe du problème de codage par descriptions multiples au problème de codage de source distribué. Dans cette section, nous nous intéresserons plus particulièrement au problème de codage par descriptions multiples avec information adjacente qui peut être vu comme un problème de codage de source distribué robuste asymétrique.*

### 3.1 Codage de Wyner-Ziv pour une transmission vidéo sur des réseaux bruités

Le codage vidéo prédictif est particulièrement sensible aux erreurs de transmission : des bits erronés ou des pertes de paquets engendrent une erreur au décodage qui peut se propager à travers la séquence du fait de l'usage de la prédiction [55]. Les décodeurs prédictifs, lorsqu'ils sont utilisés dans un environnement où le bruit de transmission introduit des erreurs, sont souvent suivis d'un post-traitement connu sous le nom de masquage d'erreurs pour limiter les effets catastrophiques du drift et de la propagation des erreurs. Malheureusement, même après ce post-traitement, la qualité du signal reconstruit reste très mauvaise.

En codage vidéo distribué, en présence d'erreurs, la qualité de l'information adjacente va également se dégrader, résultant, comme en codage prédictif, en un effet de drift au décodeur. L'information adjacente  $Y$  peut alors seulement être partiellement reconstruite et sera dénotée  $\tilde{Y}$ . Le canal virtuel du problème de codage de Wyner-Ziv est alors défini par la distribution de  $X - \tilde{Y}$  au lieu de  $X - Y$ . Les erreurs correspondantes seront corrigées si elles restent dans la limite du pouvoir de correction du code de Slepian-Wolf, qui opère alors comme un code conjoint source-canal. On notera par ailleurs que les architectures dans lesquelles le décodeur cherche, avec des méthodes proches de l'estimation de mouvement, la meilleure information adjacente sont plus à même de réduire le bruit sur le canal de corrélation dégradé. Les propriétés de résistance aux erreurs de l'architecture de codage du projet DISCOVER [107] ont été récemment évaluées dans [118, 119]. Dans [118], les trames clé sont encodées par un codeur H.264/AVC utilisant un ordonnancement flexible

---



des macroblocs (en anglais Flexible Macroblock Ordering, FMO) et des techniques de masquage d'erreurs. Une redondance de type codage de canal est ajoutée dans le flux des trames clé par des codes de Reed-Solomon, tandis que les bits de parité des trames codées en Wyner-Ziv sont considérés comme étant non protégés. Les résultats montrent que, à des taux de pertes de paquets faibles ou moyens ( $\leq 10\%$ ), un flux H.264/AVC protégé par des codes de Reed-Solomon est mieux protégé qu'un qu'un flux codé en Wyner-Ziv. Par contre, pour des taux de pertes plus élevés ( $\geq 15\%$ ), le codec Wyner-Ziv obtient de meilleures performances, en particulier à hauts débits. Les auteurs de [119] considèrent des pertes de paquets affectant à la fois les trames clé et les trames codées en Wyner-Ziv. Les résultats expérimentaux montrent que le codec de [107] est très sensible aux erreurs dans les trames clé, mais qu'il permet une dégradation progressive de la qualité lorsque des paquets de données Wyner-Ziv sont perdus.

Pour augmenter la résistance aux erreurs, le débit du codeur de Slepian-Wolf peut être choisi de manière à s'adapter au bruit sur un canal de corrélation dégradé. Par exemple, dans [120], une approche de codage conjoint source-canal est présentée. Les auteurs proposent d'effectuer le codage de Slepian-Wolf par un code Raptor [121, 122]. Les codes Raptor sont une extension des codes LT [122]. Les codes LT appartiennent à la famille des codes fontaine (en anglais fountain codes) [123], sans rendement, capables d'atteindre asymptotiquement la capacité du canal. Les codes Raptor sont construits en concaténant un code LT et un code correcteur d'erreurs appelé «précode». Celui-ci permet de relâcher des contraintes pour l'optimisation du code LT, et ainsi de se ramener à une complexité d'encodage et de décodage qui augmente linéairement avec la taille du message. Les codes Raptor encodent un message constitué de  $k$  symboles en une séquence au nombre potentiellement illimité de symboles, de façon à ce que la réception de n'importe quel sous-ensemble d'au moins  $k$  symboles codés soient suffisante pour décoder le message d'origine sans erreur. Ce type de code est particulièrement adapté à la transmission sur des réseaux sans fil sensibles aux erreurs de transmission.

D'un autre côté, le codage de Wyner-Ziv peut également être utilisé au sein d'un codeur vidéo prédictif pour augmenter la robustesse en ajoutant de l'information redondante encodée suivant les principes de codage vidéo distribué. Dans ce cas, l'information supplémentaire peut être envoyée sur un canal auxiliaire pour servir à atténuer l'effet de drift. Cette idée a été initialement suggérée par Shamai *et al.* [124] pour améliorer la transmission analogique avec une information numérique codée en Wyner-Ziv. La version analogique servant d'information adjacente pour décoder la sortie du canal numérique. Ce principe est appliqué dans [125, 126, 127] au problème de transmission numérique robuste de vidéo.

Dans [127], pour certaines trames appelées trames peg, les indices des cosets dans lesquels se trouvent les symboles transformés sont transmis en plus du résidu de la prédiction temporelle effectuée par le codeur conventionnel. La propagation d'erreurs reste alors confinée entre deux trames peg. L'idée est alors d'avoir la même robustesse que si l'on transmettait régulièrement des trames codées en mode Intra pour un coût en débit plus faible.

Dans [125], la séquence vidéo  $S$  est dans un premier temps codée conventionnellement, par un codeur MPEG-2. En parallèle,  $S$  est codée en Wyner-Ziv, mais avec une quantification plus grossière comme le montre la figure 3.1. En présence d'erreurs, des techniques de masquage d'erreurs sont appliquées sur la version décodée  $S'$  permettant ainsi d'obtenir  $S^*$ .  $S^*$  est ensuite utilisée comme information adjacente pour décoder le flux Wyner-Ziv et ainsi obtenir  $S^{**}$ . Cette architecture de codage, appelée SLEP (en anglais Systematic Lossy Error Protection), n'empêche pas totalement l'effet de drift, puisqu'elle ne permet que de

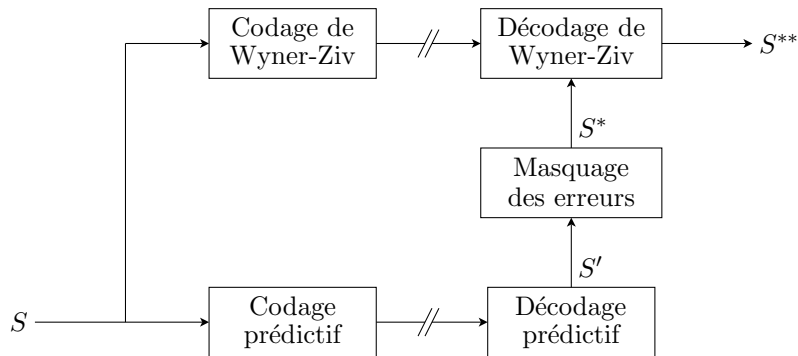


FIGURE 3.1 – Architecture de codage SLEP [125].

décoder au mieux une version grossière de  $S$ . Ce système a plus tard été implémenté dans le standard de codage vidéo H.264/AVC [128]. Les tranches redondantes sont alors utilisées pour générer les descriptions vidéo redondantes sur lesquelles un codage de Slepian-Wolf avec des codes de Reed-Solomon est appliqué. En utilisation FMO, une région d'intérêt (en anglais Region Of Interest, ROI) peut être sélectionnée dans chaque trame vidéo. La technique de [125] peut alors être appliquée de préférence sur la ROI, ce qui ne laisse que les parties les moins importantes d'arrière-plan à protéger par des techniques de masquage d'erreurs au décodeur.

Dans [126], la séquence vidéo est codée par un codeur H.263+ et une information supplémentaire codée en Wyner-Ziv est envoyée sur un canal auxiliaire. Une modélisation du bruit de corrélation du canal global (prenant en compte la corrélation avec l'information adjacente et le bruit du canal) détermine le nombre de coefficients transformés à coder en Wyner-Ziv, ainsi que leur niveau de quantification (nombre de cosets). Seuls les indices des cosets sont transmis par le codeur auxiliaire. L'algorithme utilisé pour estimer le bruit de corrélation est basé sur une version modifiée de l'algorithme ROPE (en anglais Recursive Optimal Per-pixel Estimate) [129] pour agir directement dans le domaine transformé. Dans [130], cet algorithme est remplacé par l'algorithme EDDD (en anglais Expected Distorsion of DCT coefficients) mieux adapté à une estimation dans le domaine transformé.

Dans les trois approches citées ci-dessus, l'information de prédiction constitue la partie systématique des données qui peut être protégée par un mécanisme de type FEC. Le flux codé en Wyner-Ziv est une description plus grossière de la séquence vidéo, et est redondant s'il n'y a aucune erreur de transmission. Ceci peut être vu comme un système de codage MD déséquilibré. En évitant l'effet de cliff, il a été montré que la protection contre les erreurs reposant sur un codage de Wyner-Ziv permet une dégradation plus progressive de la qualité de la vidéo reconstruite que les mécanismes de type FEC [131].

Dans le chapitre 4, nous proposons des schémas de codage MD avec filtrage temporel compensé en mouvement où les informations manquantes dans une description sont codées en Wyner-Ziv. Dans ces schémas, le flux de données Wyner-Ziv sert à atténuer l'impact dû à la perte d'une description.

### 3.2 Codage de Wyner-Ziv par couches

Le codage vidéo scalable est intéressant pour des applications telles que le streaming sur des réseaux hétérogènes et/ou vers des terminaux ayant différentes résolutions et capacités. Les solutions de codage vidéo scalable se basent souvent sur des représentations du signal par couches en incluant une boucle de prédiction inter-couches. Le problème du codage par couches prédictif, tout comme le codage prédictif dans le domaine temporel, peut être reformulé comme un problème de codage de source distribué, avec des caractéristiques communes en termes de répartition de la charge entre le codage et le décodage et de résistance aux erreurs. Tandis qu'en codage par couches, les signaux d'amélioration sont calculés à partir de réalisations codées et décodées des couches les plus basses, en codage de Wyner-Ziv, seul le modèle de corrélation entre les données Wyner-Ziv (à l'intérieur d'une couche) et l'information adjacente reconstruite à partir des couches les plus basses a besoin d'être connue.

L'encodage des signaux d'amélioration devient alors dans une certaine mesure indépendant du codec utilisée dans les couches les plus basses, la seule contrainte étant que le bruit de corrélation entre l'information adjacente reconstruite à partir des couches les plus basses et les données Wyner-Ziv reste dans le pouvoir de correction du code de Slepian-Wolf. Les conditions théoriques pour que le raffinement successif dans un codeur Wyner-Ziv puisse atteindre asymptotiquement la fonction débit-distorsion dans chaque couche, c'est-à-dire  $R_1 + \Delta R = R_{X|Y_2}^*(D_2)$  où  $R_1 = R_{X|Y_1}^*(D_1)$  est la limite Wyner-Ziv pour la couche 1, a été formulée dans [132]. En pratique, cette condition qui suppose que l'information adjacente  $Y_2$  dans la couche 2 n'apporte aucune information supplémentaire à celle utilisé dans la couche 1 est rarement vérifiée. L'information adjacente construite à partir des trames précédemment décodées sur la couche d'amélioration a de bonnes chances d'apporter de l'information supplémentaire à celle utilisée sur la couche de base.

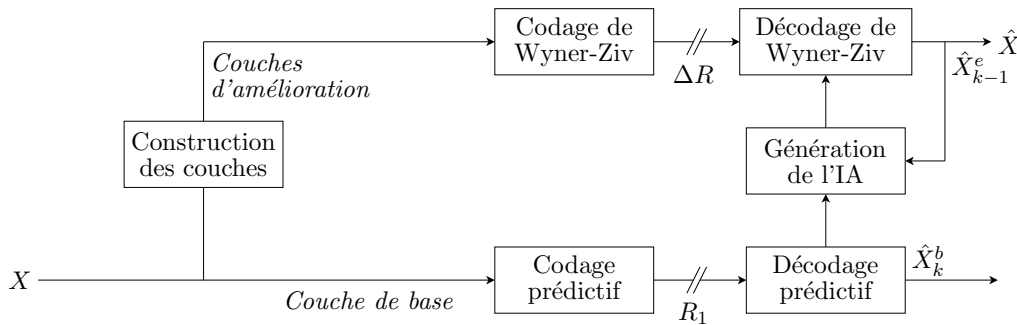


FIGURE 3.2 – Schéma de codage de Wyner-Ziv scalable avec codage prédictif de la couche de base.

Soient  $\hat{X}_k^b$  et  $\hat{X}_k^e$ , les couches de base et d'améliorations de la trame  $k$ . Soit  $\hat{X}_k^{e,j}$ ,  $j = 1, \dots, l - 1$  les  $l$  premiers plans de bits décodés de  $\hat{X}_k^e$ . Un schéma scalable en SNR est proposé dans [133] où la couche de base utilise un codec standard, et où les plans de bits des couches d'amélioration sont codés en Wyner-Ziv (comme le montre la figure 3.2). La trame reconstruite à partir de la base décodée  $\hat{X}_k^b$  et des couches d'amélioration  $\hat{X}_k^j$ ,  $j = 1, \dots, l - 1$  est utilisée comme information adjacente pour décoder  $X_k^l$ . La redondance temporelle dans les couches d'amélioration n'est pas exploitée. Dans [134], un codec basé sur PRISM scalable en spatial et en temporel est décrit. Pour la scalabilité spatiale, les vecteurs de mouvement estimés dans le codec conventionnel de la couche de base sont utilisés

aussi bien pour choisir entre une prédiction spatiale, temporelle, ou spatio-temporelle, que pour choisir les paramètres de corrélation (entraînement hors ligne) pour chaque type de prédicteur. Pour la scalabilité temporelle, les vecteurs de mouvement des couches de plus haut niveau sont obtenus à partir de ceux de la couche de base. Dans [135],  $\hat{X}_k^b$  est utilisé pour calculer un résidu  $U_k = X_k - \hat{X}_k^b$  par une boucle de prédiction fermée inter-couches. Ce résidu est alors soit codé avec un codage entropique, soit codé en Wyner-Ziv en utilisant  $V_k = \tilde{X}_k^e$  comme information adjacente, selon si la corrélation temporelle est faible ou pas.

Dans [136], Ouaret *et al.* proposent des schémas de codage vidéo distribué scalables conservant une faible complexité à l'encodeur. Ils permettent trois types de scalabilité : temporelle, spatiale et en qualité. L'information adjacente générée par l'interpolation compensée en mouvement sert à améliorer la résolution temporelle. Un sous-échantillonnage et un sur-échantillonnage spatiales sont utilisées sur les données Wyner-Ziv pour permettre la scalabilité spatiale. Enfin, la qualité de la vidéo décodée peut être contrôlée en choisissant le nombre de plans de bits à transmettre.

### 3.3 Codage de source distribué robuste

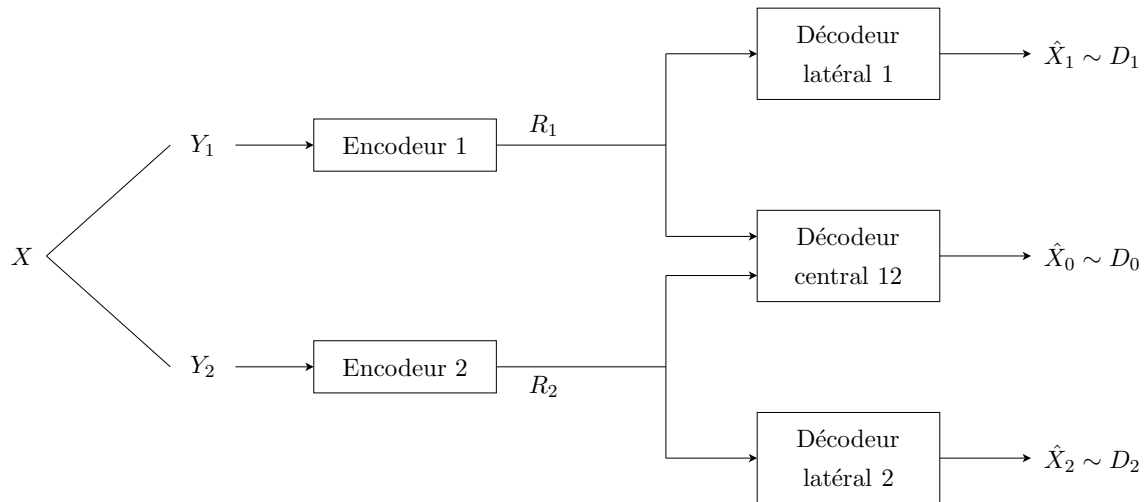


FIGURE 3.3 – Schéma générique de codage de source distribué robuste avec deux descriptions.

Le problème de codage de source distribué ne concerne que l'efficacité de codage et ignore la robustesse du système. Un schéma de codage de source distribué optimal au niveau de sa performance débit-distorsion peut être très sensible aux erreurs de transmission. En d'autres termes, il peut voir sa performance se dégrader rapidement si l'un des canaux de transmission est défaillant. À l'inverse, le problème de codage MD concerne la robustesse du système. Cependant, c'est un problème de codage de source centralisé et son architecture de codage ne peut pas s'appliquer directement à un scénario de codage de source distribué. C'est pourquoi il a été proposé d'étudier des schémas de codage de source distribué robuste capable d'ajuster le compromis entre ces deux paramètres : la robustesse du système et l'efficacité de compression.

Le problème de codage de source distribué robuste a d'abord été posé dans [137] et plus tard dans [138] et [139]. Il est présenté sur la figure 3.3. Soit  $\{X(t), Y_1(t), Y_2(t)\}_{t=1}^{\infty}$ ,

une source sans mémoire ayant comme distribution de probabilité conjointe  $P(x, y_1, y_2)$  sur  $\mathcal{X} \times \mathcal{Y}_1 \times \mathcal{Y}_2$ . La séquence  $\{X(t)\}_{t=1}^{\infty}$  n'est pas directement observable par les encodeurs. Seules  $\{Y_1(t)\}_{t=1}^{\infty}$  et  $\{Y_2(t)\}_{t=1}^{\infty}$ , deux versions dégradées de  $\{X(t)\}_{t=1}^{\infty}$  sont observables par l'encodeur 1 et l'encodeur 2 respectivement. L'encodeur  $i$  encode séparément des blocs de taille fini de la source  $\{Y_i(t)\}_{t=1}^{\infty}$ . Si un des canaux est défaillant, par exemple le canal 2, le décodage s'effectue par le décodeur latéral 1.

Chen et Berger [138] ont défini la région débit-distorsion pour ce type de problème dans le cas général avec le théorème suivant :

**Théorème 4** *De [138]. Les tuples  $(R_1, R_2, D_1, D_2, D_0)$  sont atteignables s'il existe des variables aléatoires  $(U_1, U_2, W_1, W_2)$  conjointement distribuées avec les variables de source  $(X, Y_1, Y_2)$  telles que les conditions suivantes sont satisfaites :*

1.

$$\begin{aligned} (U_1, W_1) - Y_1 - (X, Y_2, U_2, W_2), \\ (U_2, W_2) - Y_2 - (X, Y_1, U_1, W_1), \end{aligned}$$

où  $X - Y - Z$  signifie que  $X, Y, Z$  forment une chaîne de Markov, c'est-à-dire que  $X$  et  $Z$  sont conditionnellement indépendants de  $Y$  ;

2.

$$R_1 \geq I(Y_1; U_1) + I(Y_1; W_1 | U_1, U_2, W_2), \quad (3.1)$$

$$R_2 \geq I(Y_2; U_2) + I(Y_2; W_2 | U_1, U_2, W_1), \quad (3.2)$$

$$R_1 + R_2 \geq I(Y_1; U_1) + I(Y_2; U_2) + I(Y_1, Y_2; W_1, W_2 | U_1, U_2); \quad (3.3)$$

$$(3.4)$$

3. Il existe des fonctions :

$$f_i : U_i \rightarrow \mathcal{X}, \quad i = 1, 2, \quad (3.5)$$

$$f_0 : U_1 \times W_1 \times U_2 \times W_2 \rightarrow \mathcal{X}, \quad (3.6)$$

telles que  $\mathbb{E}[d(X, \hat{X}_i)] \leq D_i$ , où  $\hat{X}_i = f_i(U_i)$ ,  $i = 0, 1, 2$ .

$I(X; Y)$  est l'information mutuelle de  $X$  et  $Y$ .  $d(\cdot, \cdot)$  est une mesure de distorsion.

Le problème de codage de source distribué robuste subsume le problème de codage MD et le problème de codage multiterminal (décrit dans la section 2.2.3). Il devient un problème de codage multiterminal si  $\min(D_1, D_2) \geq D_{\max}$  où  $D_{\max} = \min_{x_0 \in \mathcal{X}} \mathbb{E}[d(X, x_0)]$ , et un problème de codage MD s'il existe des fonctions déterministes  $f_1(\cdot)$  et  $f_2(\cdot)$  telles que  $X(t) = f_1(Y_1(t)) = f_2(Y_1(t))$  pour  $t = 1, 2, \dots, \infty$ .

La construction d'un quantificateur vectoriel optimal pour ce problème est étudiée dans [140]. L'objectif du quantificateur est alors de minimiser la fonction distorsion suivante pour un débit donné  $R = R_1 + R_2$  :

$$E[d(X, \hat{X}_0) + \lambda_1 d(X, \hat{X}_1) + \lambda_2 d(X, \hat{X}_2)] \quad (3.7)$$

où  $d(\cdot, \cdot)$  est une mesure de distorsion quadratique. Les distorsions latérales sont pondérées par  $\lambda_1$  et  $\lambda_2$ , dont les valeurs dépendent de l'importance que l'on souhaite donner aux distorsions latérales par rapport à la distorsion centrale. Une technique de recuit simulé [141] est proposée pour construire un quantificateur optimal.

Le problème particulier du codage MD avec information adjacente a été étudié par Diggavi et Vaishampayan [142]. Ils considèrent deux séquences corrélées  $X$  et  $Y$  de variables aléatoires i.i.d prenant leurs valeurs dans deux alphabets finis. Le codage de  $X$  s'effectue par un codeur MD. Dans [142], la région des débits atteignables dans le cas général a été définie pour les trois configurations suivantes :

- le décodeur et l'encodeur de  $X$  ont tous les deux accès à l'information adjacente  $Y$  ;
- seul le décodeur a accès à l'information adjacente  $Y$  qui est commune aux trois décodeurs ;
- seul le décodeur a accès à une information adjacente  $Y$  qui est dégradée aux décodeurs latéraux.

Dans le cas particulier où  $X$  et  $Y$  sont deux sources gaussiennes avec  $Y = X + N$  où  $N$  est un bruit gaussien d'espérance nulle et de variance  $\sigma_N^2$ , Diggavi et Vaishampayan ont montré que la région débit-distorsion est la même, que l'encodeur ait accès ou non à l'information adjacente.

Wang *et al.* [143, 144] ont également étudié le problème du codage MD avec information adjacente mais ils se sont plus précisément intéressés au cas où les deux décodeurs utilisent deux informations adjacentes différentes  $Y_1$  et  $Y_2$ . Dans [144], ils ont établi la région débit-distorsion pour des sources gaussiennes lorsque les informations adjacentes sont connues au décodeur et au décodeur et ils l'ont comparée à la région obtenue dans [143] lorsqu'aucune des informations adjacentes n'est connue à l'encodeur. Ils ont montré que cette dernière région est incluse dans la première et qu'elles coïncident seulement si  $Y_1 = Y_2$ .

Dans le chapitre 5, nous proposons un schéma de codage MD avec information adjacente basée sur la MDSQ et testons ses performances sur des sources gaussiennes. L'information adjacente est la même pour tous les décodeurs et n'est pas connue à l'encodeur. Dans le chapitre 6, nous proposons un schéma de codage vidéo MD utilisant également la MSDQ pour générer des descriptions mais où l'information adjacente est également codée par des descriptions multiples ce qui fait que sa qualité diffère également en fonction du nombre de descriptions reçues.

---



## Chapitre 4

# Codage vidéo par descriptions multiples protégées par des principes de codage de source distribué

*Dans ce chapitre nous allons présenter<sup>1</sup> des schémas de codage par descriptions multiples basés sur des filtres temporels dyadiques et 3-bandes appliqués sur des séquences sous-échantillonnées. Nous introduirons alors plusieurs méthodes pour construire des descriptions corrélées à partir de ces bancs de filtres. Dans une seconde partie, pour augmenter leur robustesse et leur stabilité au cours du temps, nous introduirons un codec Wyner-Ziv au sein de ces schémas redondants : les trames qui en principe ne sont pas transmises dans une description seront ici codées en Wyner-Ziv. Dans un dernier schéma, on appliquera un filtrage temporel compensé en mouvement sur les trames codées en Wyner-Ziv pour ne transmettre que les sous-bandes de basses fréquences. Ce chapitre se termine par une étude des résultats de simulation sur deux séquences vidéo sélectionnées pour leurs caractéristiques très différentes de mouvement.*

### 4.1 Introduction

La nature temps réel des flux de données envisagés pour la distribution de contenu multimédia impose l'utilisation de protocoles de transport, tels que User Datagram Protocol (UDP) et/ou Real-time Transport Protocol (RTP) qui ne possèdent pas de mécanisme de contrôle qui garantirait un certain niveau de QoS. Les données transmises peuvent donc être sujettes à des pertes dues à des défaillances du réseau ou à des problèmes de congestion. Les approches traditionnelles pour combattre ces pertes se basent le plus souvent sur l'utilisation de techniques ARQ et/ou FEQ. ARQ offre au niveau applicatif un service ga-

---

1. Ce travail a fait l'objet d'une publication *Distributed temporal multiple description coding for robust video transmission* (O. Crave, C. Tillier, C. Guillemot, B. Pesquet-Popescu) dans «EURASIP Journal on Wireless Communications and Networking, Special Issue on Multimedia over Wireless Networks, January, 2008». La première partie de ce travail sur des schémas de codage par descriptions multiples dans le domaine temporel a fait l'objet d'une publication *A comparison of four video multiple description coding schemes* (C. Tillier, O. Crave, B. Pesquet-Popescu, C. Guillemot) dans «Proc. of European Signal Processing Conference (EUSIPCO), Poznań, September 4-8, 2007». La deuxième partie de ce travail avec l'introduction d'un flux codé en Wyner-Ziv a également fait l'objet d'une publication *A comparison of four video multiple description coding schemes* (O. Crave, C. Guillemot, B. Pesquet-Popescu, C. Tillier) dans «Proc. of EUSIPCO, Poznań, September 4-8, 2007».

---



ranti de transport de données. Cependant, le délai induit par la retransmission des paquets perdus n'est pas toujours approprié pour les applications multimédia avec des contraintes de délai. La méthode FEC consiste à envoyer de l'information redondante en même temps que l'information d'origine. L'avantage du FEC est qu'il n'est pas nécessaire d'avoir un canal de retour. Par contre, si la qualité du canal se dégrade rapidement à cause de «fading» ou «shadowing», ou si l'estimation de la probabilité d'avoir des erreurs de transmission est plus faible que la valeur réelle, l'information de parité FEC n'est pas suffisante pour corriger les erreurs. D'où une qualité de vidéo qui se dégrade rapidement, conduisant à l'indésirable effet de *cliff*.

Le codage par descriptions multiples a été récemment considéré pour la transmission robuste de vidéo sur des canaux avec pertes. Plusieurs représentations codées corrélées du signal sont créées et transmises sur de multiple canaux. Le problème adressé ici est comment atteindre les meilleures performances en rapport débit-distorsion lorsque tous les canaux fonctionnent, sous la contrainte de la distorsion moyenne lorsque seulement un sous-ensemble de canaux fonctionne. Des systèmes pratiques pour générer des descriptions qui approchent le plus ces bornes théoriques ont été proposés considérant les différents composants d'un système de compression, tels que la transformée spatio-temporelle ou la quantification. Un aperçu des différentes méthodes de codage par descriptions multiples est donné dans le chapitre 1.

Le codage Wyner-Ziv peut également être utilisé comme un mécanisme FEC. Le principe est appliqué dans [125, 127] au problème de transmission numérique robuste de vidéo. Cependant, dans un tel système, la propagation d'erreurs dans le flux MPEG peut avoir un impact négatif sur la qualité de l'information adjacente et dégrader les performances débit-distorsion du schéma.

Nous proposons une solution à ce problème en structurant les données à encoder en deux descriptions. L'idée d'ajouter un codec Wyner-Ziv dans un schéma de codage MD pour combattre les pertes de paquets lors d'une transmission sur des canaux bruités a d'abord été envisagée par Wu et al. [145]. Dans leur schéma, l'erreur de prédiction entre les deux descriptions est encodée par un code de Reed-Solomon. Seuls les bits de parité sont envoyés sur le canal bruité. Au décodeur, lorsque les deux descriptions sont disponibles, le décodeur central combine les deux descriptions pour reconstruire la vidéo et l'information codée en Wyner-Ziv n'est pas utilisée. En présence d'erreurs, les descriptions partiellement reçues peuvent être vues comme des versions bruitées de la description sans erreur. Le décodeur se sert alors des bits de parité et de la version bruitée de la description comme information adjacente pour décoder les bits erronés s'ils peuvent être corrigés.

Dans ce chapitre, nous commençons par présenter quatre schémas de codage MD. Dans le premier schéma, les trames paires et impaires sont séparées entre deux descriptions. Trois niveaux d'une décomposition de type Haar compensée en mouvement sont ensuite appliqués sur les trames de chaque description. Dans le second schéma, les trames sont d'abord séparées en groupes de deux trames consécutives entre les descriptions. Trois niveaux de décomposition temporelle de type Haar compensée en mouvement sont alors appliqués sur chaque description. Les troisième et quatrième schémas sont semblables au premier et deuxième schéma, mais différent dans leur utilisation d'un filtre 3-bandes de type Haar [146] au lieu d'un filtre 2-bandes. Ces schémas permettent d'atteindre de bonnes performances débit-distorsion au décodeur central, mais obtiennent de fortes variations de PSNR aux décodeurs latéraux.

Le compromis entre les performances aux décodeurs central et latéraux dépend évidemment du taux de redondance entre les deux descriptions. La qualité du signal reconstruit

par les décodeurs latéraux peut être améliorée en effectuant un codage systématique avec pertes des descriptions. La séquence d'origine est séparée en deux sous-séquences, une étant codée comme dans les schémas précédemment décrits, l'autre étant codée en Wyner-Ziv. Cela revient à avoir un codage Wyner-Ziv systématique avec pertes de chaque trame de chaque description. Ce système de contrôle d'erreur peut être utilisé comme une alternative à ARQ ou FEC. Le flux de données additionnel peut être envoyé systématiquement au décodeur ou peut être envoyé à sa demande, selon qu'il existe un canal de retour et/ou une tolérance de l'application à la latence. La quantité de redondance ajoutée dans chaque description est principalement contrôlée par la quantification des données Wyner-Ziv. Cette première approche conduit à des performances débit-distorsion satisfaisantes au niveau des décodeurs latéraux, mais souffre d'une trop grande redondance, qui pénalise la description centrale, lorsque le schéma est utilisé sous la forme d'un mécanisme FEC. Pour résoudre ce problème, la méthode proposée est ensuite étendue à l'utilisation d'un filtrage temporel compensé en mouvement (en anglais Motion Compensated Temporal Filtering, MCTF) pour les trames Wyner-Ziv, auquel cas seules les sous-bandes de basses fréquences sont codées en Wyner-Ziv et envoyées dans les descriptions.

Le chapitre est organisé de la façon suivante. La section 4.2 décrit quatre schémas de codage MD en vidéo basés sur différentes structures de séparation temporelle et des MCTF de 2 ou 3-bandes. Les sections 4.3 et 4.4 montrent comment rendre plus robuste ces schémas en utilisant le codage de description systématique avec pertes. La section 4.5 présente les résultats de simulation pour les codecs proposés. Elle est suivie d'une conclusion dans la section 4.6.

## 4.2 Schémas de codage par descriptions multiples

Commençons par considérer le schéma illustré sur la figure 4.1, où les trames paires et impaires sont réparties entre les deux descriptions. Un niveau de décomposition de type Haar compensée en mouvement est appliqué sur les trames de chaque description. Les trames temporelles de détail sont encodées, tandis que le passage d'un niveau à l'autre est effectué en entrelaçant les trames d'approximation des deux descriptions. Cette nouvelle séquence sera par la suite distribuée encore une fois entre les deux descriptions. Ce schéma sera appelé le schéma MDC temporel Haar au niveau trame (MDCT-T Haar).

Le second schéma (voir figure 4.2), appelé le schéma MDC temporel Haar au niveau GOF (MDCT-G Haar), commence par séparer des groupes de deux trames consécutives pour les regrouper dans deux descriptions. A nouveau, un niveau de MCTF de type Haar est appliqué sur ces couples de trames, et les détails sont encodés dans leur description respective. Comme précédemment, le passage du premier niveau au suivant est effectué en entrelaçant les trames d'approximation provenant des deux descriptions. Ensuite, la procédure de codage se poursuit comme pour le schéma MDCT-T Haar, en encodant avec un MCTF de type Haar les trames paires et impaires dans les différentes descriptions. On peut remarquer qu'il n'est pas possible d'avoir le même regroupement comme pour le premier niveau dans les groupes de deux trames, car le filtrage temporel serait réalisé sur les trames d'approximation venant de différentes descriptions, donc au cas où l'une d'entre elles devait être perdue, il serait impossible de les reconstruire. Une autre remarque est qu'il ne serait pas évident d'utiliser des filtres temporels plus longs dans ce type de structure, car pour tous les schémas de codage MD présentés ici, la distance temporelle entre les trames dans la même description est supérieure à un, et plus le filtre est long, plus la corrélation entre les trames sera faible. C'est pourquoi nous nous restreindrons à un MCTF de type

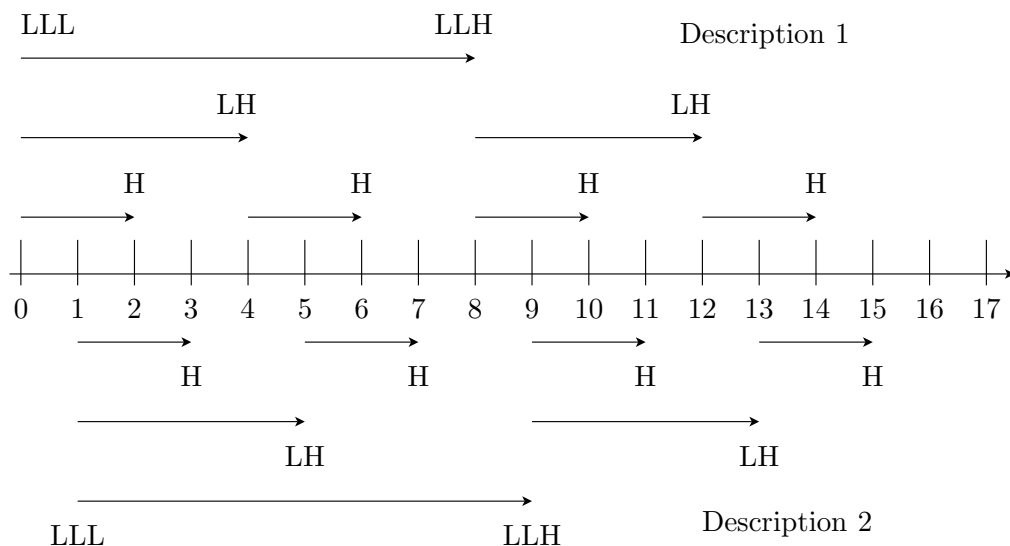


FIGURE 4.1 – MDCT-T Haar : les trames vont deux à deux dans les descriptions puis un MCTF de type Haar est appliqué sur chacune des descriptions.

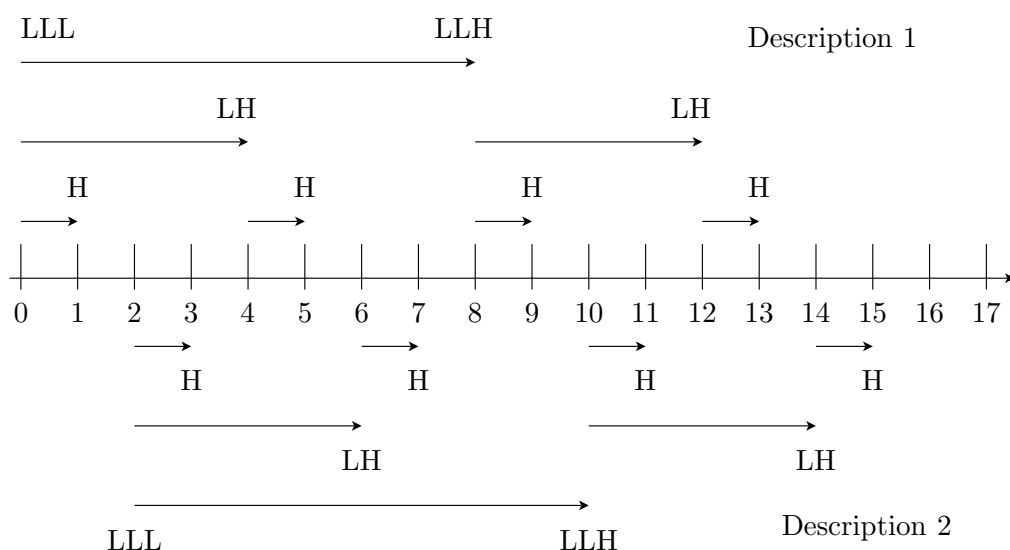


FIGURE 4.2 – MDCT-G Haar : un MCTF de type Haar est appliqué sur des groupes de deux trames dans chaque description.

---

Haar, même si les performances de codage du MCTF 5/3 sont connues pour être meilleures en l'absence de perte.

Dans le second schéma, comme l'encodage est réalisé sur des couples de trames successives, on peut espérer obtenir de meilleures performances de la part du décodeur central par rapport au schéma MDCT-T Haar, où une trame sur deux est considérée dans chaque description. Cependant, dans le schéma MDCT-T Haar, lorsqu'une seule description est reçue, le décodeur latéral aura à reconstruire une trame sur deux. La distance temporelle entre les trames manquantes étant seulement de un, cette tâche n'est pas très difficile, et on peut s'attendre à ce que les performances visuelles et objectives (PSNR) soient bonnes. D'un autre côté, pour le schéma MDCT-G Haar, la distance temporelle entre les trames manquantes de la description perdue est de deux, donc leur interpolation sera plus complexe.

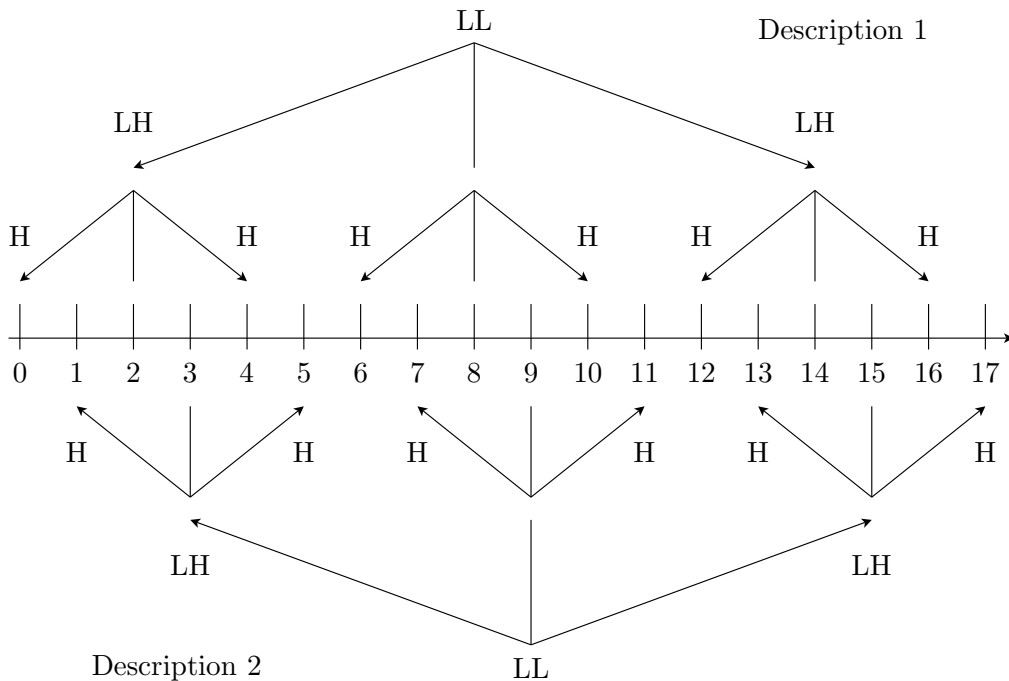


FIGURE 4.3 – MDCT-T 3-bandes : les trames paires et impaires sont séparées et un MCTF 3-bandes est appliqué sur chaque description.

Le troisième schéma, appelé le schéma MDCT-T 3-bandes, illustré sur la figure 4.3, comporte une séparation temporelle des trames en entrée entre les paires et les impaires, pour les deux descriptions, suivi d'un MCTF 3-bandes de type Haar sur chaque flux. Les trames d'approximation sont entrelacées pour former une nouvelle séquence au deuxième niveau de décomposition. Le MCTF 3-bandes de type Haar fonctionne comme le MCTF 2-bandes de type Haar : un opérateur de prédiction est appliqué de façon symétrique entre  $x_{3t}$  et  $x_{3t+1}$ , respectivement entre  $x_{3t}$  et  $x_{3t-1}$ , aboutissant à deux trames de détail. Ensuite, l'étape de mise à jour implique le calcul de la moyenne des détails compensés en mouvement avec la trame centrale  $x_{3t}$ . Des opérateurs de mise à jour qui minimisent l'erreur de reconstruction dans ces structures de filtrage spatio-temporel ont été proposés dans [147] pour les schémas 2 et 3-bandes.

Le dernier schéma de codage MD, appelé le schéma MDCT-G 3-bandes, est identique au schéma MDCT-T 3-bandes, mis à part que les descriptions avant filtrage sont obtenues

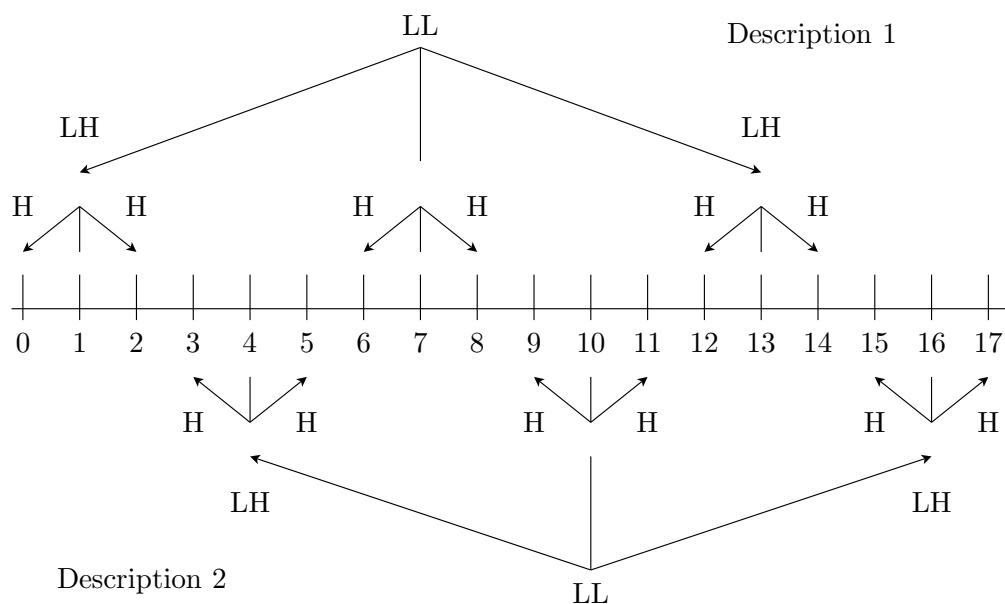


FIGURE 4.4 – MDCT-G 3-bandes : un MCTF 3-bandes est appliqué sur des groupes de trois trames dans chaque description.

en séparant des groupes de trois trames consécutives dans chaque description (voir la figure 4.4). Un MCTF 3-bandes de type Haar est appliqué sur les triplets. Comme dans le cas des schémas 2-bandes, pour cette décomposition, par rapport à la précédente, on peut espérer de meilleures performances au décodeur central. Aux décodeurs latéraux, à cause de la plus grande distance temporelle entre les trames utilisées pour interpoler celles qui manquent, on peut s’attendre à une détérioration par rapport au schéma MDCT-T 3-bandes. En effet, pour le schéma MDCT-T 3-bandes, la distance temporelle entre les trames manquantes est seulement un, tandis que pour le schéma MDCT-G 3-bandes les décodeur latéraux auront à interpoler des trames espacées de trois trames pour remplir les trous résultants de la perte d’une description. D’un autre côté, il y aura un gain en performance grâce au fait que le filtrage d’origine est effectué sur des groupes de trames consécutives, au lieu de trames espacées de un. Ces deux tendances antagonistes seront étudiées dans la section 4.5.

### 4.3 Codage de descriptions systématique avec pertes dans le domaine pixel

Les schémas précédents présentent différents compromis entre la qualité (PSNR et visuelle) des descriptions centrale et latérales. Ces compromis dépendent de la quantité de redondance introduite dans les deux descriptions. Dans les schémas de codage MD vus plus tôt, la redondance résulte principalement du fait de la séparation temporelle des trames de la séquence d’entrée dans les deux sous-séquences qui forment les descriptions, la corrélation temporelle entre les trames adjacentes dans la séquence d’origine n’étant pas exploitée de façon optimale. La qualité du signal reconstruit par les décodeurs latéraux peut être accrue en encodant de façon systématique avec pertes les descriptions. Dans cette section et dans les résultats de simulation, nous nous intéresserons aux schémas

---

MDCT-T 3-bandes (figure 4.3) et MDCT-G 3-bandes (figure 4.4) de la section 4.2 mais les schémas MDCT-T Haar et MDCT-G Haar dyadiques peuvent aussi être modifiés de la même manière.

Commençons par nous intéresser à l'architecture de codage représentée dans les figures 4.5 (encodeur) et 4.6 (décodeur). A l'encodeur, la source est d'abord divisée en deux séquences conduisant à deux descriptions *non redondantes* de la séquence d'origine. Deux approches sont ensuite considérées pour la séparation des trames. Dans la première, de la même façon que pour le schéma MDCT-T 3-bandes de la section précédente, les deux sous-séquences sont construites en séparant les trames paires et impaires comme on peut le voir dans la figure 4.7, alors que la seconde approche consiste à séparer les trames par groupe de trois trames, ainsi que le fait le schéma MDCT-G 3-bandes comme le montre la figure 4.8. Les schémas correspondants seront appelés les schémas MDC 3-bandes distribué au niveau trame (MDCD-T 3-bandes) et MDCD-G 3-bandes. Dans chaque description, les trames d'une sous-séquence sont considérées comme *trames clé* tandis que les trames de l'autre sous-séquence sont considérées comme trames Wyner-Ziv. La sous-séquence des trames clé est d'abord transformée temporellement en utilisant un MCTF 3-bandes de type Haar avec deux niveaux de décomposition temporelle. Les trames restantes (trames Wyner-Ziv) sont transformées avec une transformée en cosinus discrète (DCT) entière sur des blocs de taille  $4 \times 4$  et quantifiées avec un quantificateur scalaire uniforme. Les coefficients transformés sont structurés dans des sous-bandes spatiales et chaque plan de bits des sous-bandes quantifiées est ensuite séparément codé par un turbo code. Les bits de parité résultants sont stockés dans une mémoire-tampon. Aux décodeurs latéraux, les trames clé sont décodées et une information adjacente est générée en interpolant les trames intermédiaires à partir des trames clé. Le turbo décodeur corrige ensuite cette information adjacente en utilisant les bits de parité. Les séquences de parité stockées dans la mémoire-tampon sont transmises en petites quantités à la demande du décodeur via un canal de retour. Si l'estimation du taux d'erreurs bit à la sortie du décodeur excède un certain seuil, davantage de bits de parité sont demandés à l'encodeur. Cela revient à contrôler le débit du code en choisissant différents modèles de poinçonnage à la sortie du turbo code. Le taux d'erreurs bit est estimé à partir du rapport de log-vraisemblance sur les bits en sortie du turbo décodeur. Le paramètre de corrélation utilisé dans le turbo décodage est obtenu à partir du résidu des trames clé compensées en mouvement.

Les trames encodées en tant que trames clé dans la première description sont encodées en tant que trames Wyner-Ziv dans la deuxième description et vice versa. Ainsi, si les deux descriptions sont reçues, le décodeur n'utilise que les trames clé pour reconstruire la séquence. D'un autre côté, si seulement une description est reçue, le décodeur utilise l'information Wyner-Ziv dans la description reçue pour reconstruire les trames manquantes. La quantité de redondance est définie par la quantification des trames Wyner-Ziv : plus la quantification est fine, plus le débit Wyner-Ziv est élevé. Lorsque le schéma est utilisé dans un scénario FEC, les flux Wyner-Ziv sont systématiquement envoyées et ignorées au décodeur central. Dans un scénario ARQ, les flux Wyner-Ziv sont seulement envoyés si demandés par le décodeur en introduisant cependant une certaine latence dans l'application. Dans les résultats présentés plus loin, seul le scénario FEC est envisagé.

Il est important de remarquer que le débit Wyner-Ziv ne dépend pas seulement du degré de quantification des trames Wyner-Ziv, mais également de la qualité de l'information adjacente, et donc du degré de quantification des trames clé.

---

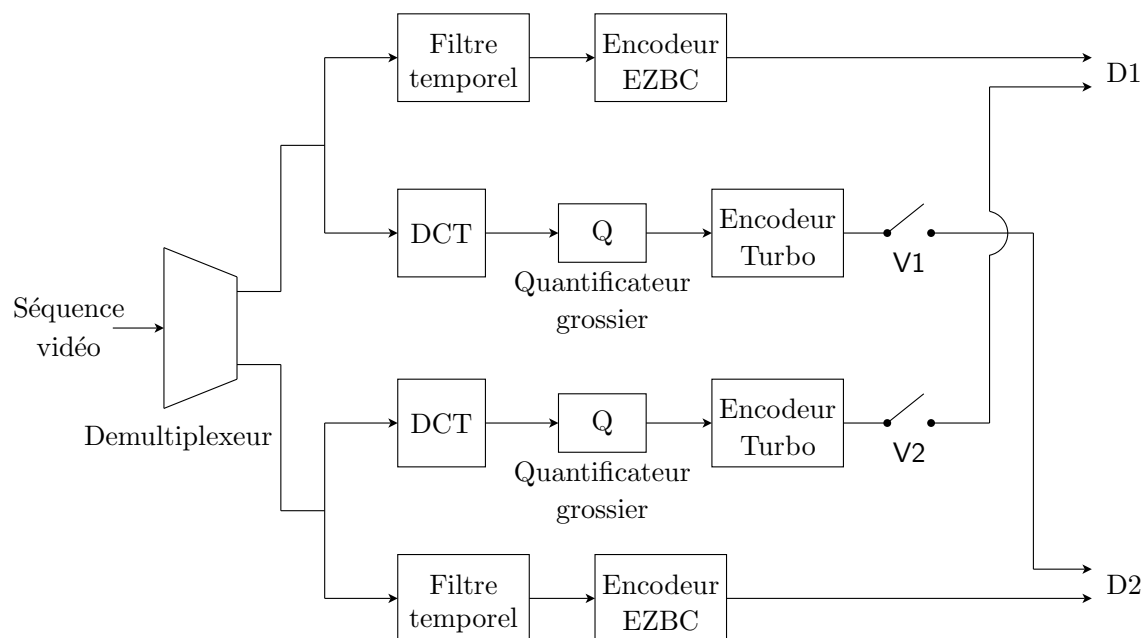


FIGURE 4.5 – Implémentation de l’encodeur de description systématique avec pertes dans le domaine pixel.

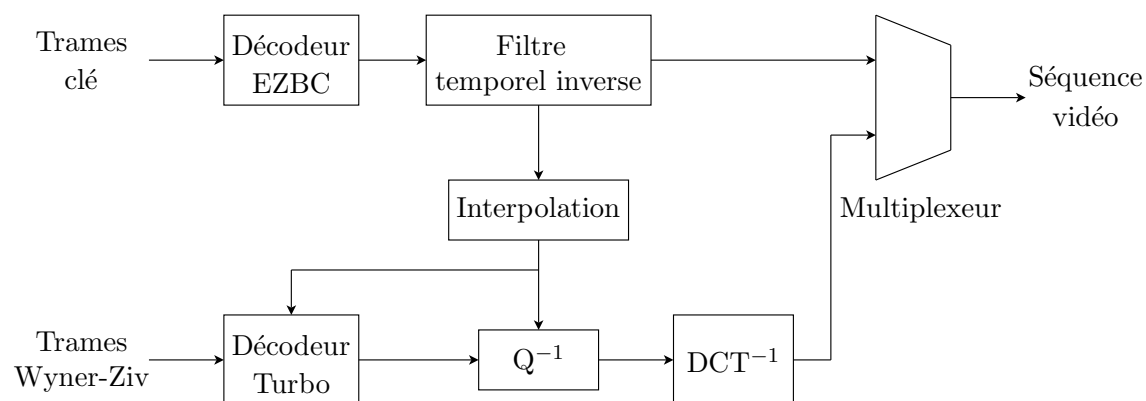


FIGURE 4.6 – Implémentation du décodeur de description systématique avec pertes dans le domaine pixel.

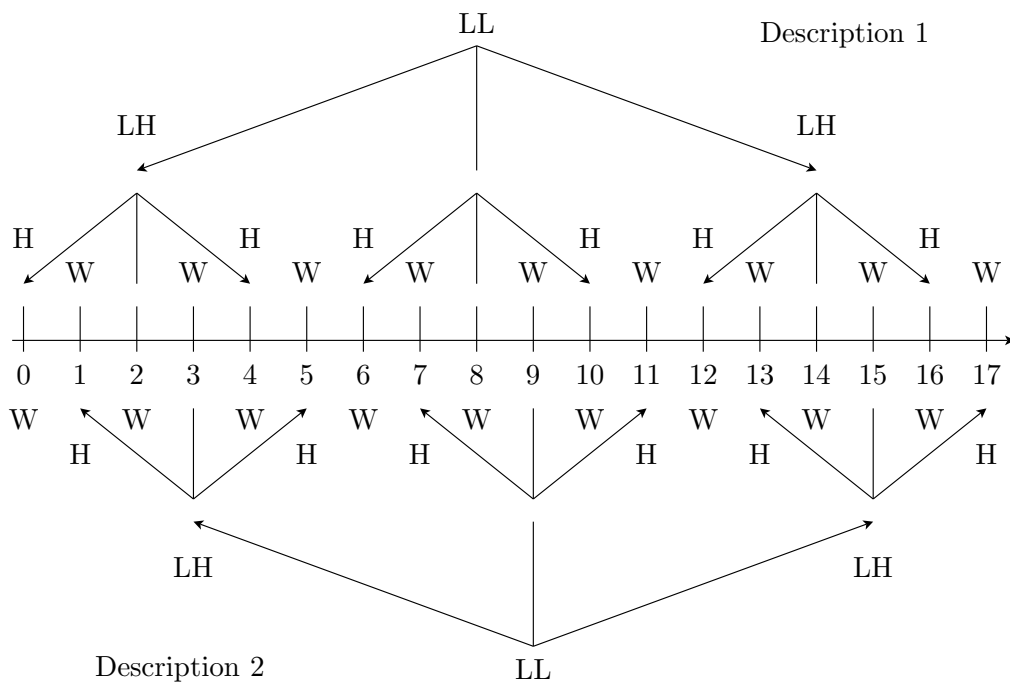


FIGURE 4.7 – MDCD-T 3-bandes : La séquence est séparée en trames paires et impaires. Une sous-séquence est codée classiquement tandis que l’autre est codée en Wyner-Ziv.

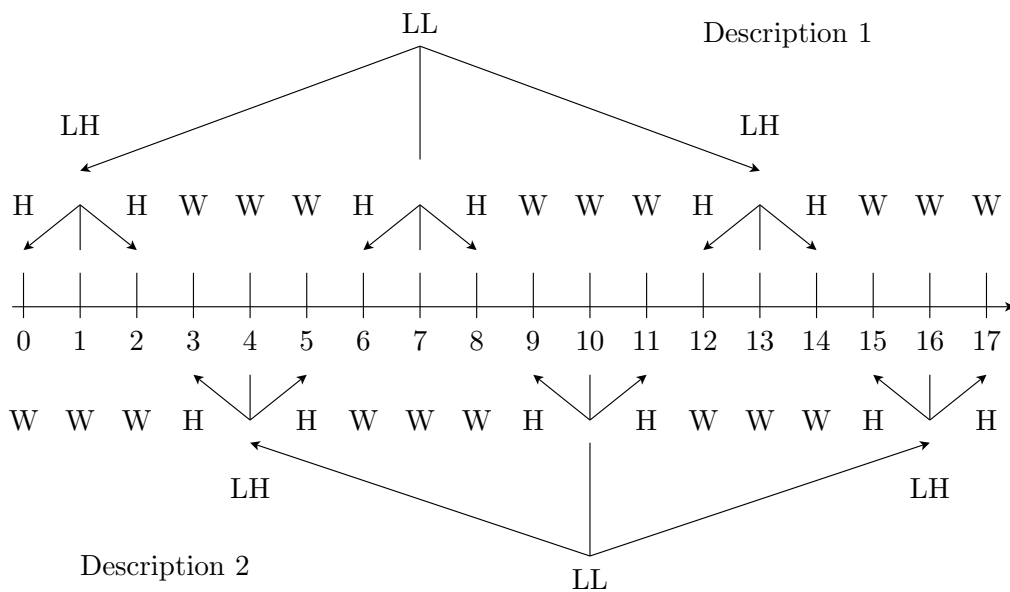


FIGURE 4.8 – MDCD-G 3-bandes : La séquence est séparée par groupes de trois trames consécutives. Une sous-séquence est codée classiquement tandis que l’autre est codée en Wyner-Ziv.



### 4.4 Codage de descriptions systématique avec pertes dans le domaine MCTF

Afin de réduire le débit Wyner-Ziv et d'améliorer les performances au décodeur central, nous proposons une seconde architecture, où les trames Wyner-Ziv sont tout d'abord transformées par le même MCTF 3-bandes de type Haar que celui utilisé pour les trames clé dans le schéma MDCT-G 3-bandes, mais avec seulement un niveau temporel pour conserver une distance raisonnable entre les sous-bandes. De plus, avant d'entrer dans l'encodeur Wyner-Ziv, les sous-bandes sont filtrées par un filtre passe-bas de telle façon que seules les sous-bandes de basse fréquence soient codées en Wyner-Ziv. L'architecture de ce nouveau codec est présentée dans les figures 4.9 (encodeur) et 4.10 (décodeur). Pour ce codec, l'approche qui consiste à séparer les trames selon la taille du GOP utilisé par le filtre temporel a été privilégiée pour obtenir les deux sous-séquences, comme le montre la figure 4.11. Les vecteurs de mouvement sont codés par le même codage arithmétique adaptatif que celui utilisé pour les trames clé. Aux décodeurs latéraux, l'information adjacente est obtenue en transformant les trames interpolées avec un MCTF 3-bandes de type Haar et les basses fréquences résultantes sont utilisées comme information adjacente pour aider au décodage des sous-bandes codées en Wyner-Ziv. Pour reconstruire les trames, les sous-bandes de basses fréquences qui viennent d'être décodées sont combinées avec les sous-bandes de hautes fréquences des trames interpolées pour obtenir une séquence de sous-bandes qui est finalement filtrée par le banc de filtres de synthèse et combinée avec la séquence de trames clé.

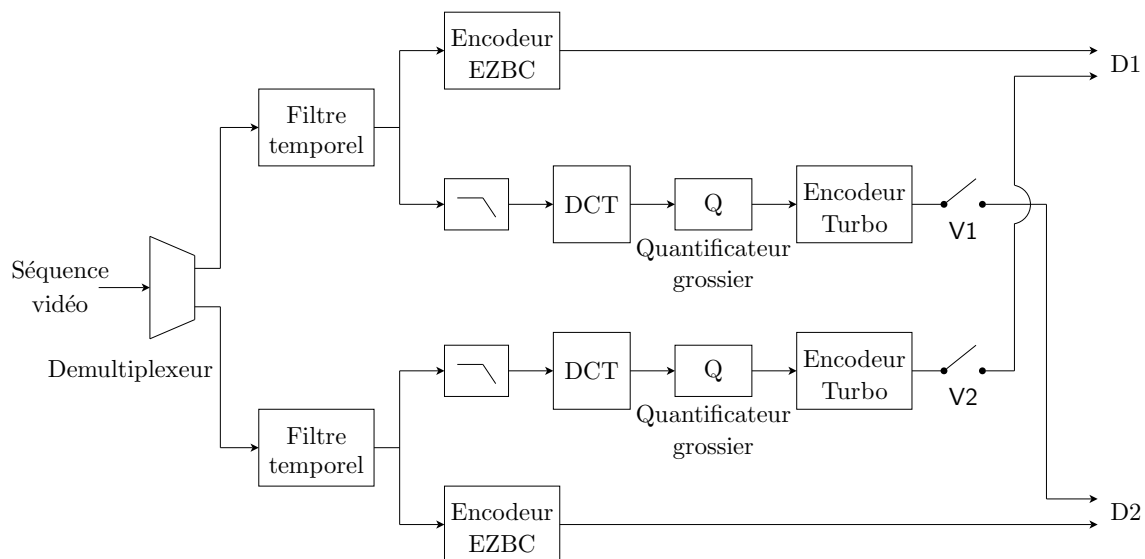


FIGURE 4.9 – Implémentation du schéma d'encodage de description systématique avec pertes dans le domaine MCTF.

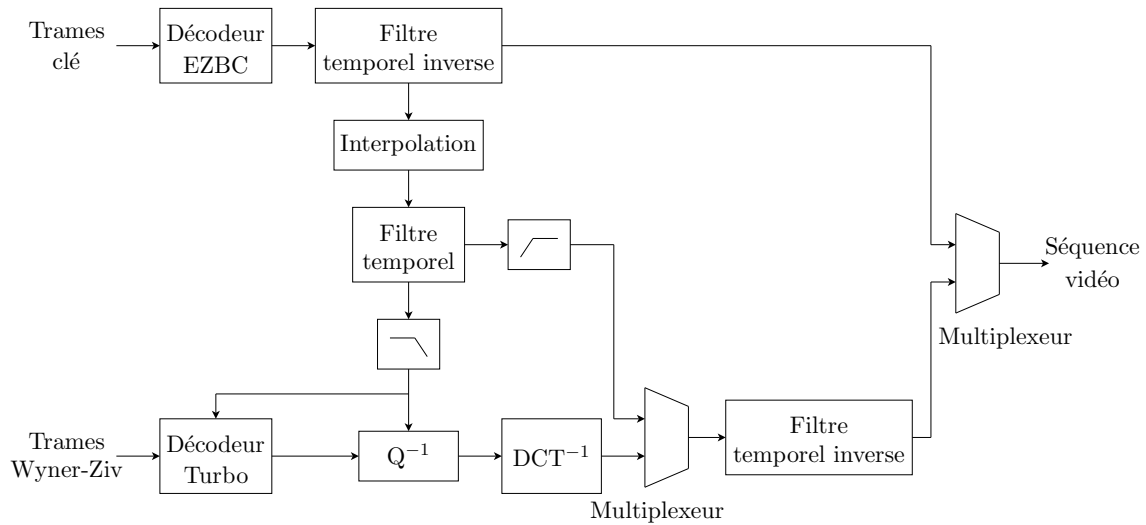


FIGURE 4.10 – Implémentation du schéma de décodage de description systématique avec pertes dans le domaine MCTF.

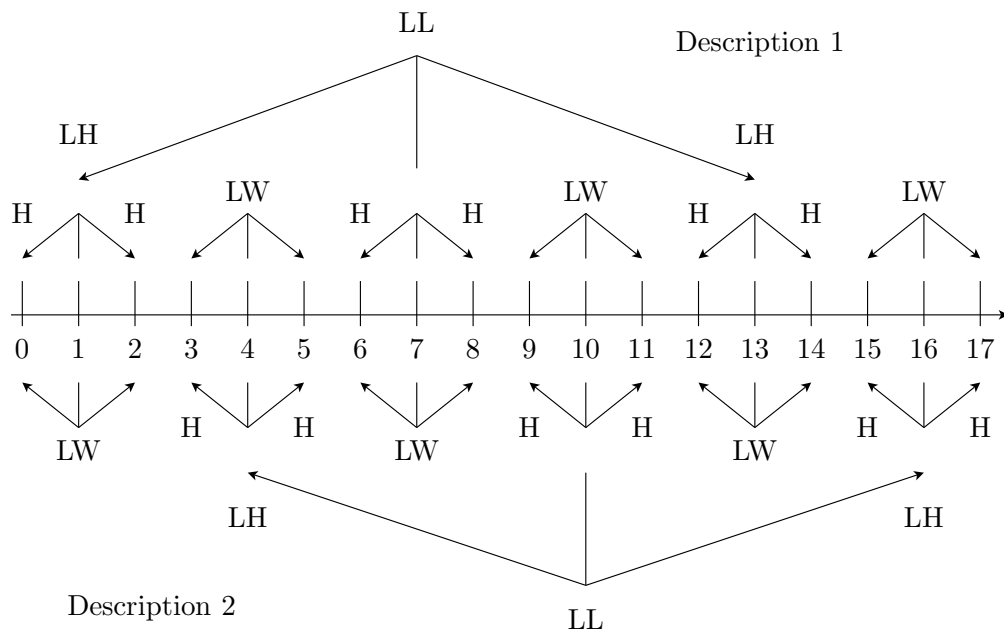


FIGURE 4.11 – Schéma MDCD-G 3-bandes dans le domaine MCTF : La séquence est séparée en groupes de trois trames. Une sous-séquence est codée à l'aide d'un codeur conventionnel tandis que l'autre est filtrée temporellement et seules les sous-bandes de basse fréquence sont codées en Wyner-Ziv.

## 4.5 Résultats de simulation

### 4.5.1 Analyse des performances des schémas de codage MD

Nous commençons par comparer les quatre schémas de codage MD de la section 4.2. Ils ont été implémentés en utilisant le logiciel MC-EZBC [148]. Trois niveaux de décomposition sont réalisés pour les schémas 2-bandes MCTF (c'est-à-dire les schémas MDCT-T Haar et MDCT-G Haar) et deux niveaux pour les schémas MCTF 3-bandes (c'est-à-dire les schémas MDCT-T 3-bandes et MDCT-G 3-bandes). L'estimation et la compensation de mouvement dans le MCTF sont réalisées en utilisant un algorithme de block-matching hiérarchique à taille de bloc variable (HVSBM) avec des tailles de blocs variant de  $64 \times 64$  à  $4 \times 4$  et une précision d'1/8ème de pixel.

Les performances centrales et latérales des schémas MDCT-T Haar et MDCT-G Haar, basés sur un MCTF 2-bandes, sont montrées sur les figures 4.12 et 4.13. Comme prévu, le décodeur central du schéma MDCT-G Haar est plus performant que celui du schéma MDCT-T Haar. Le décodeur latéral du schéma MDCT-T Haar est légèrement plus efficace que celui du schéma MDCT-G Haar. Ces résultats reflètent la difficulté d'interpoler deux trames consécutives lorsqu'une seule description est reçue dans le schéma MDCT-G. Pour la séquence Foreman, on peut également remarquer que même si les deux schémas diffèrent seulement au premier niveau de décomposition temporelle, l'écart entre les performances de codage est assez élevé (entre 2 dB et 1 dB pour les décodeurs central et latéraux respectivement).

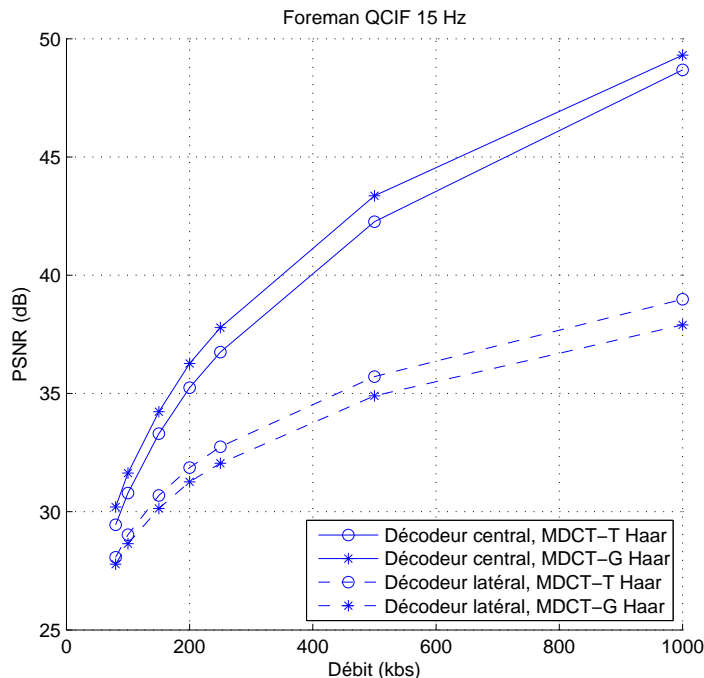


FIGURE 4.12 – Comparaison des performances des schémas MDCT-T Haar et MDCT-G Haar (Foreman, QCIF, 15 Hz).

Les performances des schémas MDCT-T 3-bandes et MDCT-G 3-bandes, basés sur un MCTF 3-bandes, sont illustrées sur les figures 4.14 et 4.15. Comme dans le cas des schémas

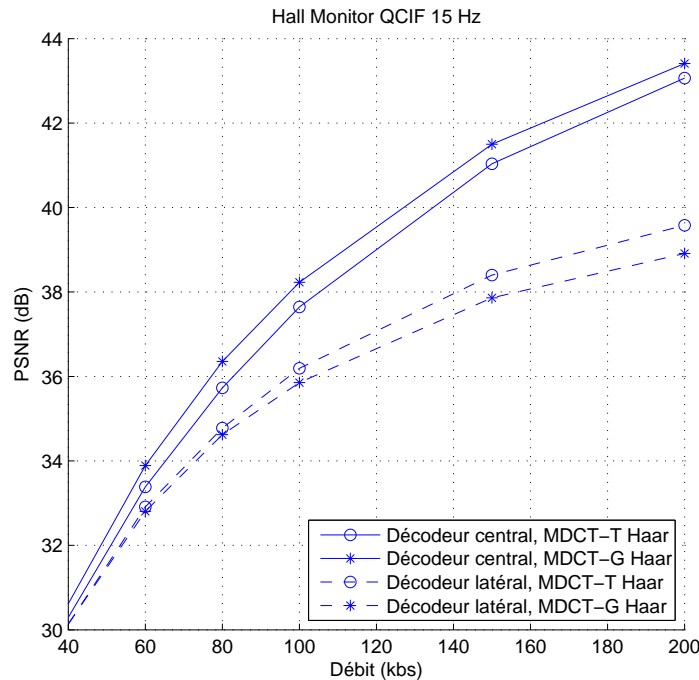


FIGURE 4.13 – Comparaison des performances des schémas MDCT-T Haar et MDCT-G Haar (Hall Monitor, QCIF 15 Hz).

2-bandes MCTF, le fait de regrouper des trames consécutives avant d’effectuer leur filtrage et leur encodage dans les différentes descriptions conduit, comme prévu, à de meilleurs résultats au décodeur central du schéma MDCT-G. Une amélioration de qualité pouvant aller jusqu’à 1.5 dB pour la séquence Foreman et 0.5 dB pour Hall Monitor peut être observée. Cependant, ce gain est obtenu au prix d’une chute du PSNR (jusqu’à 2 dB pour Foreman et 1 dB pour Hall Monitor) aux décodeurs latéraux, les décodeurs latéraux ayant besoin d’interpoler trois trames manquantes à partir de trames éloignées temporellement.

#### 4.5.2 Analyse des performances des schémas de codage MD distribué

Nous allons maintenant nous intéresser à l’apport au niveau des performances visuelles et en termes de PSNR des séquences codées en Wyner-Ziv. Pour ce faire, les résultats des schémas MDCD-T et MDCD-G sont confrontés à ceux obtenus par le schéma MDC 3-bandes [65]; ce schéma est construit sur une structure temporelle 3-bandes compensées en mouvement où la redondance provient du sous-échantillonnage de la séquence d’origine d’un facteur 2 au lieu d’un facteur 3.

Les tests ont été effectués pour quatre points de débit-distorsion, correspondant aux matrices de quantification  $4 \times 4$  illustrées sur la figure 4.16. Dans chaque matrice de quantification  $4 \times 4$ , la valeur à la position  $k$  indique le nombre de niveaux de quantification associés aux bandes de coefficients DCT  $b_k$ ; la valeur 0 signifie qu’aucun bit Wyner-Ziv n’a été transmis pour la bande correspondante. Par la suite, les différentes matrices seront appelées  $Q_i$  avec  $i = 1, \dots, 4$ . Plus l’indice  $i$  est élevé, plus le débit et la qualité sont élevés.

Les débits utilisés pour les trames clé sont 20, 40, 60, 80, 100, 150 et 200 kbs pour Hall Monitor et 80, 100, 150, 200, 250, 500 et 1000 kbs pour Foreman. Les figures 4.17

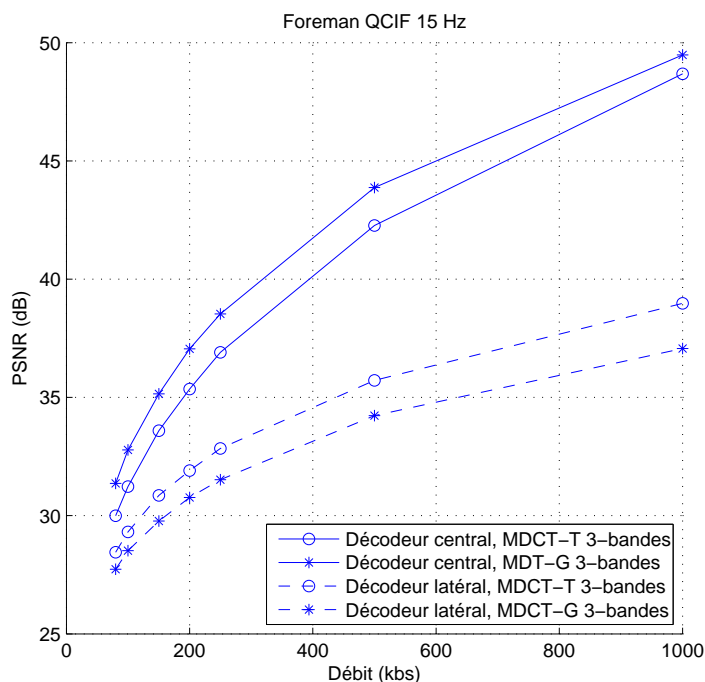


FIGURE 4.14 – Comparaison des performances des schémas MDCT-T 3-bandes et MDCT-G 3-bandes (Foreman, QCIF, 15 Hz).

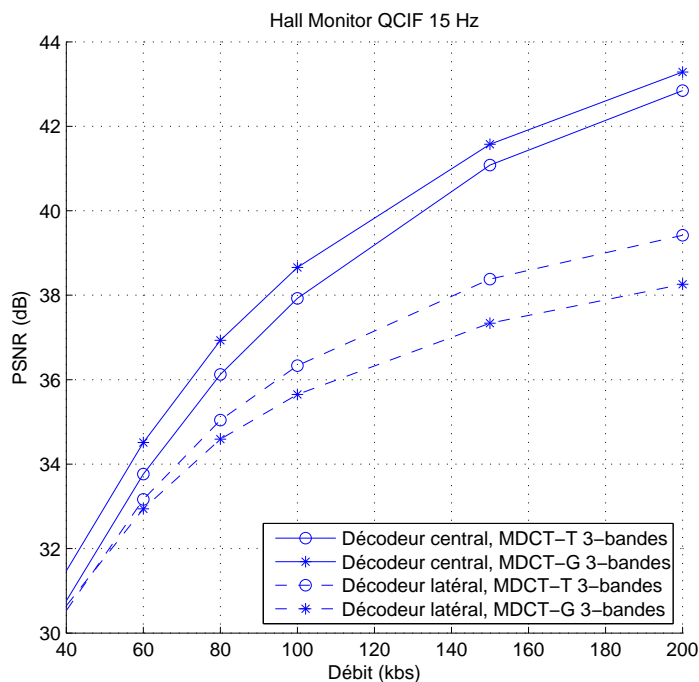


FIGURE 4.15 – Comparaison des performances des schémas MDCT-T 3-bandes et MDCT-G 3-bandes (Hall Monitor, QCIF 15 Hz).

16	8	0	0		32	8	0	0
8	0	0	0		8	0	0	0
0	0	0	0		0	0	0	0
0	0	0	0		0	0	0	0
$Q_1$					$Q_2$			
32	8	4	0		32	16	8	4
8	4	0	0		16	8	4	0
4	0	0	0		8	4	0	0
0	0	0	0		4	0	0	0
$Q_3$					$Q_4$			

FIGURE 4.16 – Quatre matrices de quantification correspondants à différentes performances débit-distorsion.

et 4.18 montrent les performances obtenues avec le schéma MDCD-T 3-bandes au décodeur central pour Foreman et Hall Monitor. Le débit correspond au débit global (celui des deux descriptions). Pour Hall Monitor, le schéma MDCT-T 3-bandes est systématiquement meilleur que le schéma MDC 3-bandes (+1 dB) mais est moins bon (-0.5 dB) dans le cas de Foreman. Comme prévu, le PSNR augmente lorsqu'un flux Wyner-Ziv est envoyé dans chaque description. Les figures 4.19 et 4.20 montrent les performances du schéma MDCD-T 3-bandes au décodeur latéral. Cette fois, le schéma MDCD-T 3-bandes est légèrement meilleur que le schéma MDC 3-bandes avec ou sans information supplémentaire, en particulier pour la séquence Foreman et pour des débits élevés.

Une comparaison des schémas se basant uniquement sur le PSNR moyen (la moyenne des PSNR entre les trames reçues et les trames perdues qui sont obtenues par interpolation avec ou sans information supplémentaire) n'est pas suffisante, car elle ne prend pas en compte les variations de PSNR au cours du temps. La figure 4.21 montre les variations de PSNR de la 50ème à la 100ème trame de la séquence Foreman à 307 kbs pour le schéma MDCD-T 3-bandes utilisant la matrice de quantification  $Q_1$  et le schéma MDC 3-bandes aux décodeurs central et latéraux. Aux décodeurs latéraux, la figure montre une chute brutale des valeurs de PSNR du schéma MDC 3-bandes (jusqu'à 16.5 dB) lorsque les trames manquantes sont simplement interpolées, tandis que le PSNR reste plus stable pour le schéma MDCD-T 3-bandes (la valeur la plus faible étant 25.9 dB), alors que le PSNR moyen est seulement 1 dB plus faible pour le schéma MDC 3-bandes que pour le schéma MDCD-T 3-bandes. Cependant, au décodeur central, le schéma MDC 3-bandes est plus performant que le schéma MDCD-T 3-bandes (+2.2 dB) à cause de la présence des données contenues dans le flux Wyner-Ziv qui ne contribuent pas au décodage central et sont simplement ignorées.

Les figures 4.22 et 4.23 montrent la variance du PSNR entre les trames pour les décodeurs central et latéraux. Au décodeur central, la variance est plus élevée pour le schéma MDCD-T que pour les schémas MDCT-T 3-bandes et MDC 3-bandes mais reste raison-

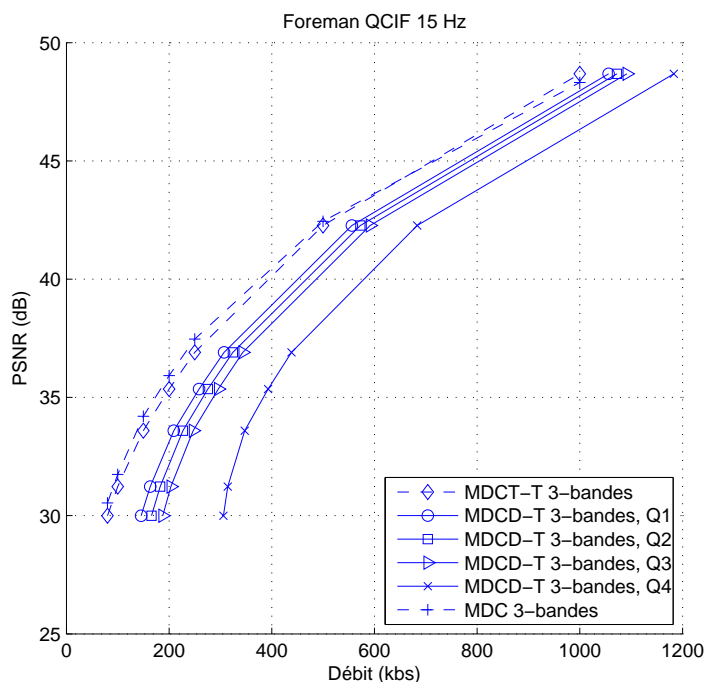


FIGURE 4.17 – Comparaison des distorsions centrales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps).

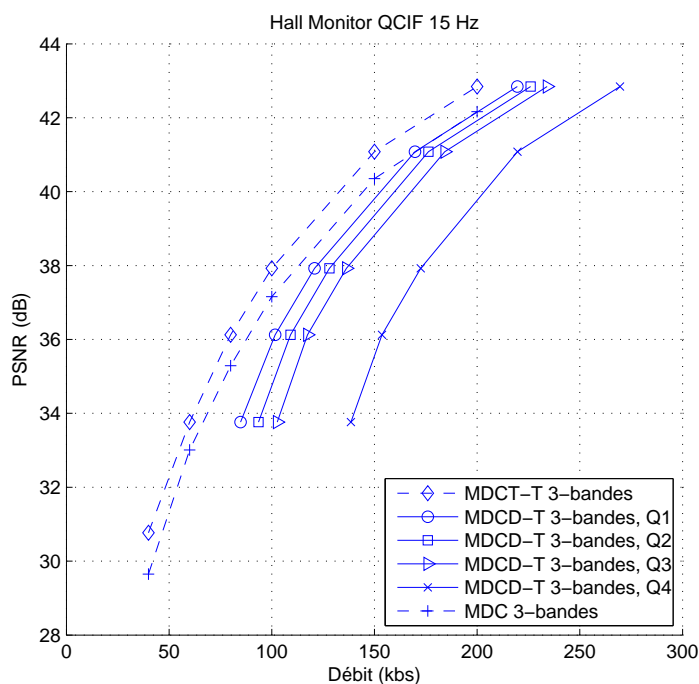


FIGURE 4.18 – Comparaison des distorsions centrales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).

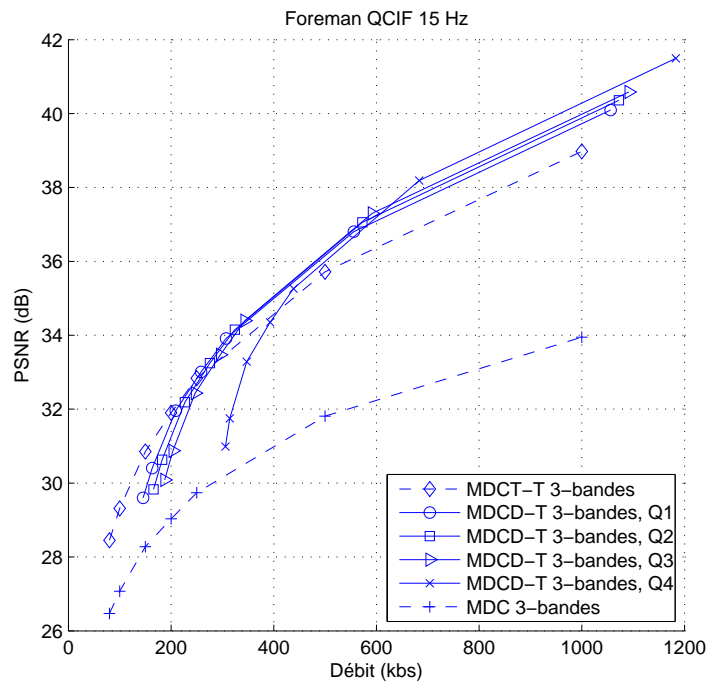


FIGURE 4.19 – Comparaison des distorsions latérales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps).

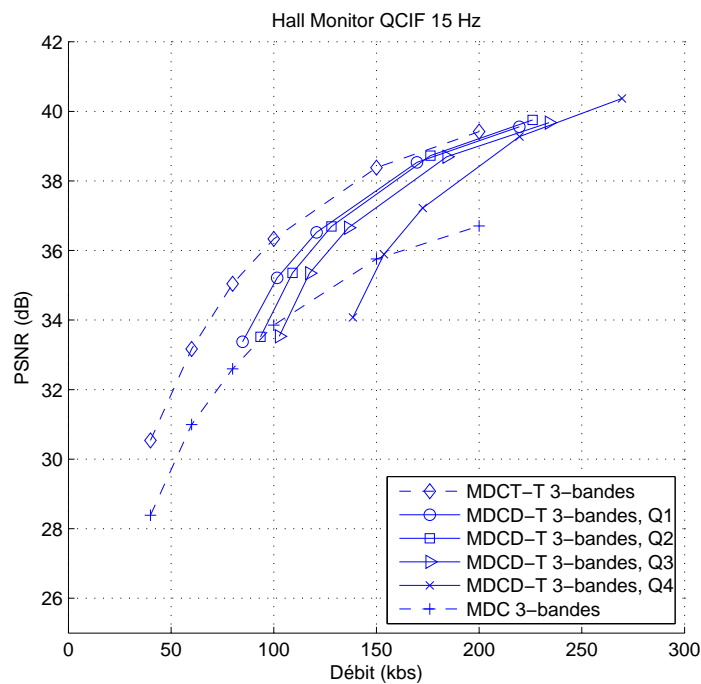


FIGURE 4.20 – Comparaison des distorsions latérales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).



nable (moins de 1.8). Aux décodeurs latéraux, l'utilisation d'un flux Wyner-Ziv additionnel réduit dramatiquement les fluctuations de PSNR. La diminution de la variance peut atteindre 100 comparée au schéma MDC 3-bandes à 1000 kbs. Cette figure montre clairement l'avantage que l'on peut tirer d'une utilisation de valeurs élevées pour  $Q_i$  aux décodeurs latéraux;  $Q_4$  donnant dans cet exemple les résultats les plus stables.

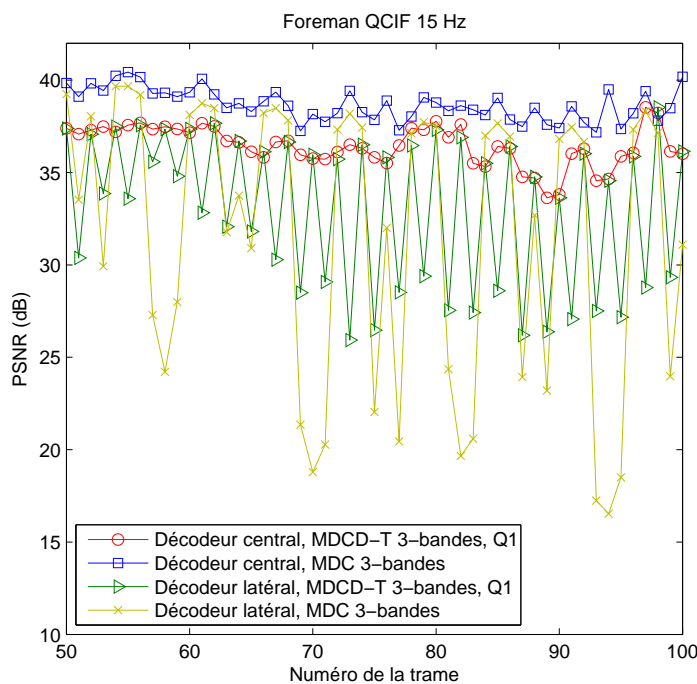


FIGURE 4.21 – Variations centrales et latérales du PSNR entre la 50ème et la 100ème trame de la séquence Foreman (QCIF, 15 fps) at 307 kbs.

Les figures 4.24 et 4.25 montrent les performances du schéma MDCD-G 3-bandes au décodeur central pour les séquences Foreman et Hall Monitor respectivement. Comme prévu, les performances de codage sont meilleures que celle du schéma MDCT-T 3-bandes et, cette fois, le schéma MDCT-T 3-bandes est systématiquement plus efficace que le schéma MDC 3-bandes (+1.5 dB pour Foreman et +2 dB pour Hall Monitor). Cependant, le schéma MDCD-G 3-bandes avec un flux codé en Wyner-Ziv donne toujours de moins bons résultats que le schéma MDC 3-bandes, en particulier à bas débit, et plus  $Q_i$  augmente, plus faibles sont les performances débit-distorsion au décodeur central. Les figures 4.26 et 4.27 montrent les performances du schéma MDCD-G 3-bandes au décodeur latéral. Dans ce cas, le schéma MDC 3-bandes donne de moins bons résultats, bien que l'interpolation soit faite sur trois trames consécutives. Comme on peut le voir, le schéma MDCD-G 3-bandes n'est pas très efficace par rapport au schéma MDCD-T 3-bandes à cause de la quantité importante de bits de parité qui sont demandés au moment du turbo décodage par le décodeur pour palier à la mauvaise qualité de l'information adjacente.

Générer les deux descriptions en séparant la séquence en deux sous-séquences de trames paires et impaires rend le filtrage temporel moins efficace, affaiblit la corrélation entre les trames et résulte en de mauvaises performances débit-distorsion au décodeur central. De plus, en envoyant les données Wyner-Ziv pour chaque trame de la séquence, on aboutit à

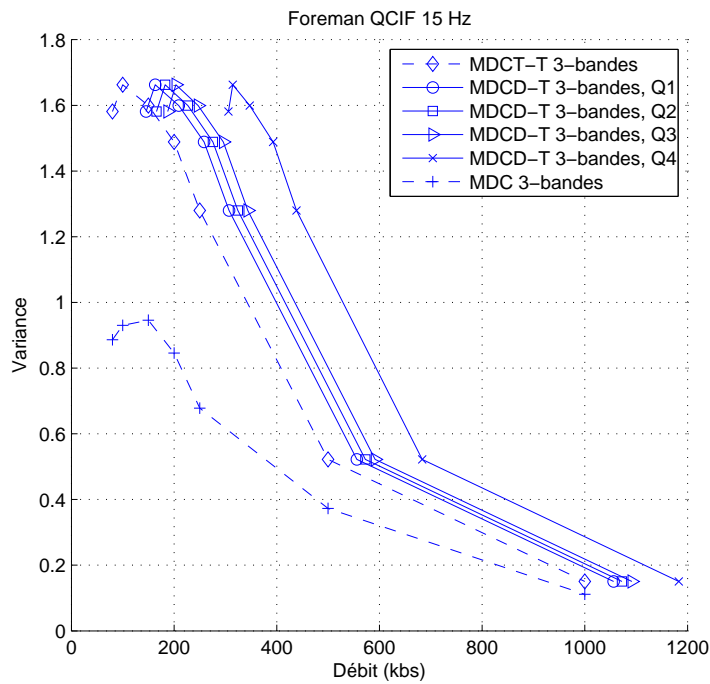


FIGURE 4.22 – Comparaison des variations du PSNR au décodeur central pour le schéma MDCD-T 3-bandes et le codec MDC 3-bandes (Foreman, QCIF 15 fps).

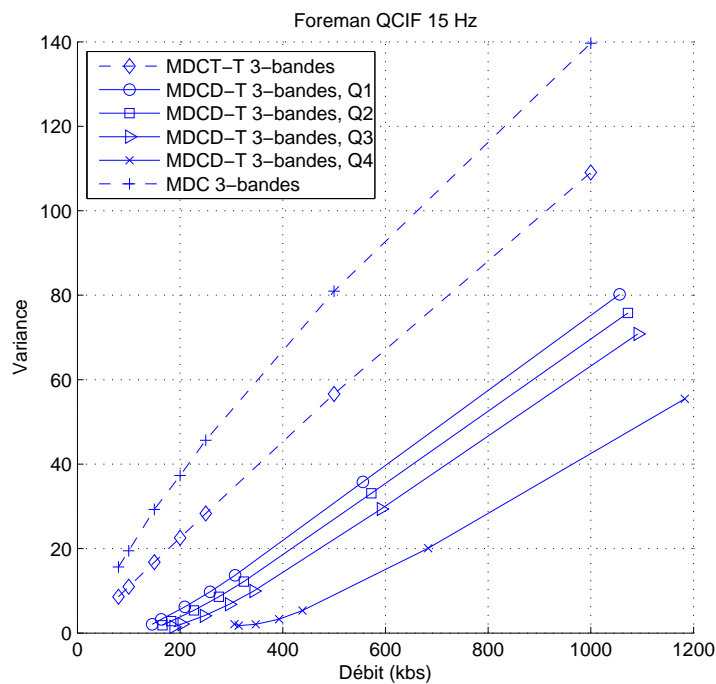


FIGURE 4.23 – Comparaison des variations du PSNR aux décodeurs latéraux pour le schéma MDCD-T 3-bandes et le codec MDC 3-bandes (Foreman, QCIF 15 fps).

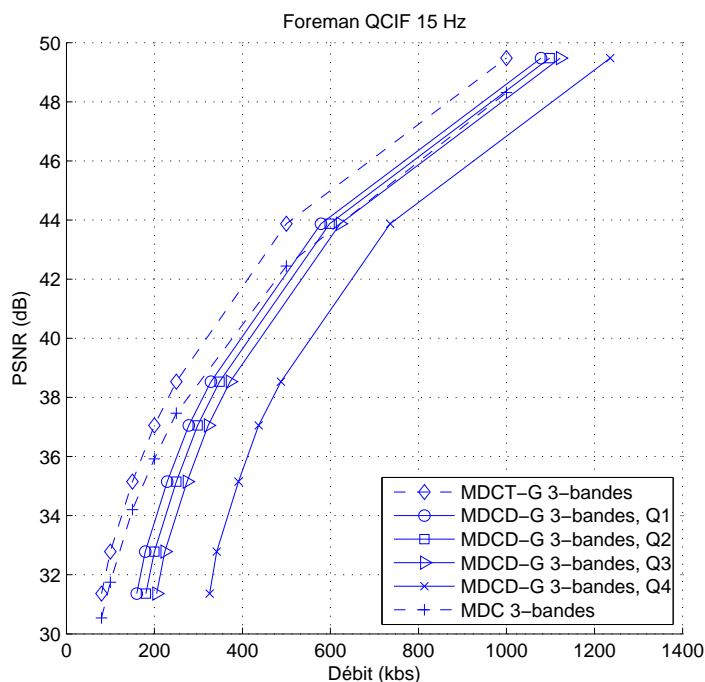


FIGURE 4.24 – Comparaison des distorsions centrales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps).

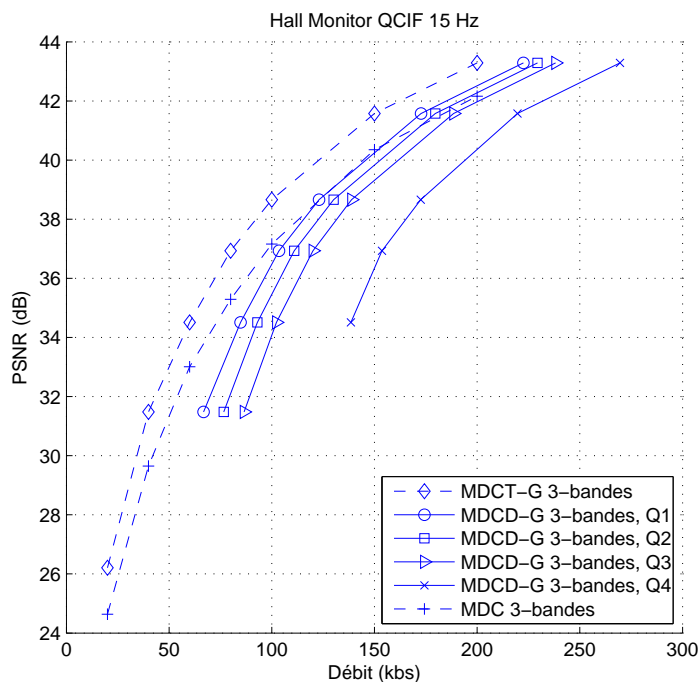


FIGURE 4.25 – Comparaison des distorsions centrales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).

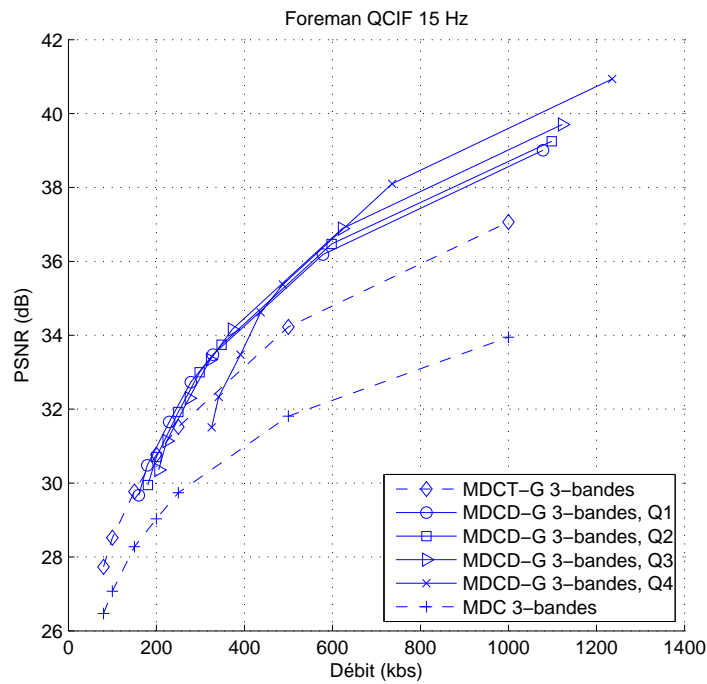


FIGURE 4.26 – Comparaison des distorsions latérales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps).

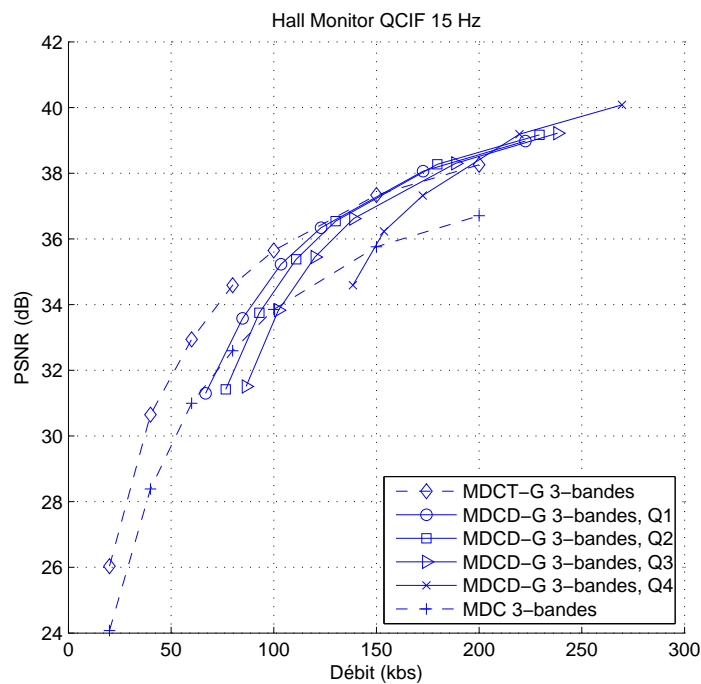


FIGURE 4.27 – Comparaison des distorsions latérales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).

un schéma totalement redondant. Pour résoudre ce problème, nous proposons le schéma MDGD-G 3-bandes dans le domaine MCTF où la séparation des trames est effectuée par groupes de trois trames, comme le montre la figure 4.11, et où seules les sous-bandes de basse fréquence sont codées en Wyner-Ziv.

Les figures 4.28 et 4.29 montrent les résultats obtenus par le schéma MDGD-G 3-bandes dans le domaine MCTF au décodeur central pour Foreman et Hall Monitor. Sur cette figure, on peut voir que le schéma MDGD-G 3-bandes dans le domaine MCTF est plus performant que le schéma MDC 3-bandes pour les valeurs de  $Q_i$  les plus faibles ( $i < 4$ ) et les débits les plus élevés (à partir de 300 kbs pour Foreman et 60 kbs pour Hall Monitor). En même temps, les performances aux décodeurs latéraux que montrent les figures 4.30 et 4.31 sont toujours supérieures à celles du schéma MDC 3-bandes, bien qu'elles soient plus faibles que celles obtenues précédemment par les schémas MDGD-T 3-bandes et MDGD-G 3-bandes.

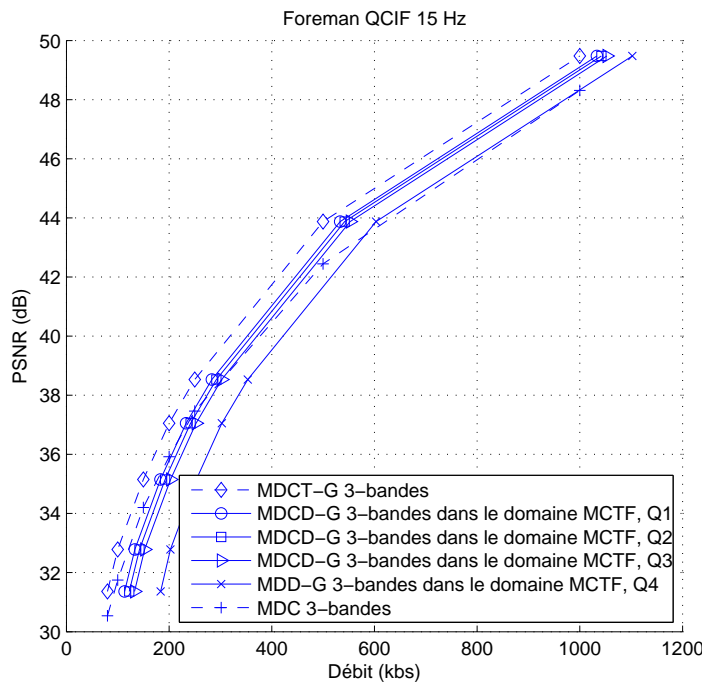


FIGURE 4.28 – Comparaison des distorsions centrales du schéma MDGD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Foreman, QCIF 15 fps).

## 4.6 Conclusion

Nous avons présenté dans ce chapitre une architecture de codage MD basée sur la séparation temporelle des trames dans une séquence suivie d'un MCTF. Cette structure a ensuite été généralisée à une séparation par groupes de trames et à un MCTF 3-bandes de type Haar. Les résultats expérimentaux ont montré que regrouper des trames consécutives avant de les filtrer et de les encoder dans différentes descriptions nous permet d'obtenir de meilleurs résultats au décodeur central et de moins bons résultats aux décodeurs latéraux que d'effectuer une séparation entre les trames paires et impaires. De plus, il a été montré que cet effet est davantage visible lorsque les séquences étudiées sont sujettes à beaucoup

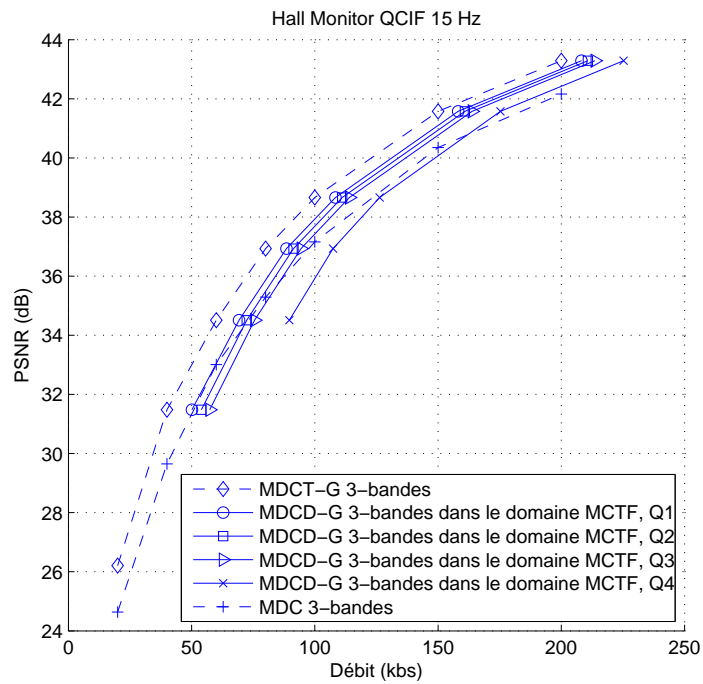


FIGURE 4.29 – Comparaison des distorsions centrales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).

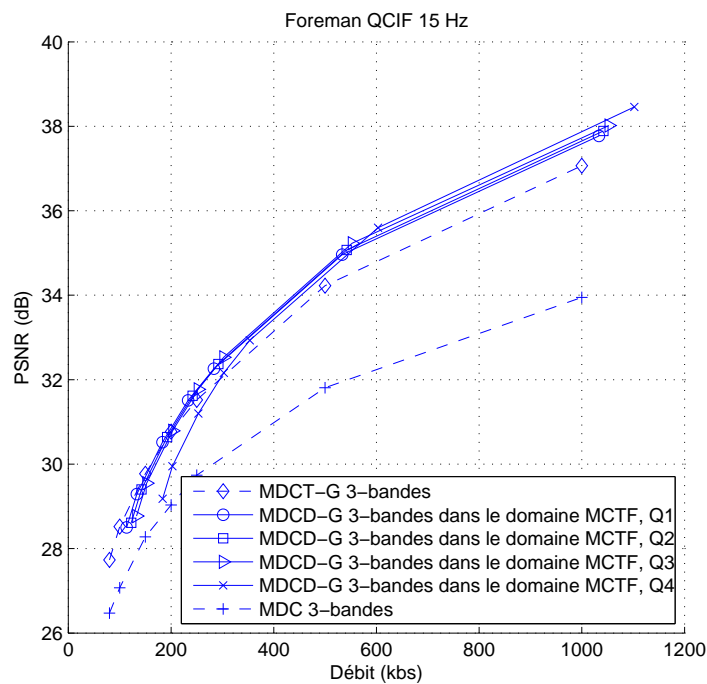


FIGURE 4.30 – Comparaison des distorsions latérales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Foreman, QCIF 15 fps).

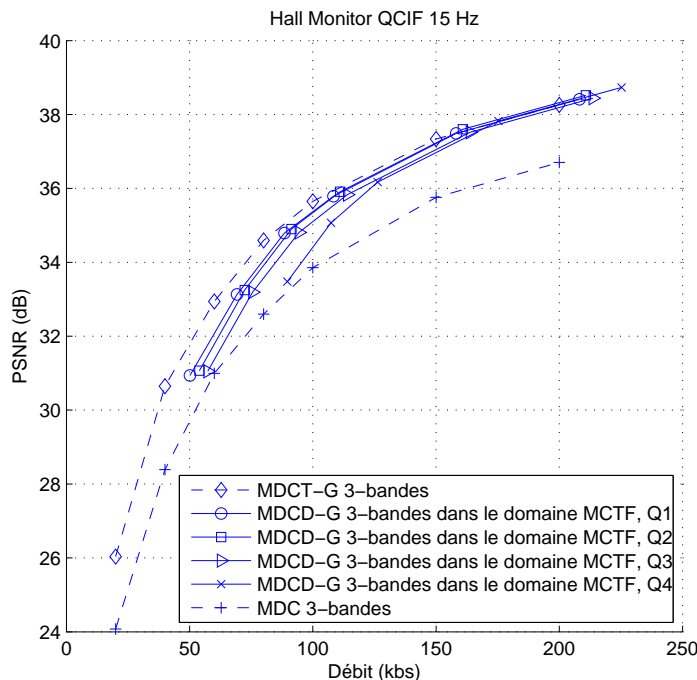


FIGURE 4.31 – Comparaison des distorsions latérales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).

de mouvement.

Les schémas de codage de description systématique avec pertes, où les trames manquantes sont codées en Wyner-Ziv dans chaque description, ont été proposées dans l'objectif de limiter les fortes variations de la qualité au cours du temps des descriptions latérales des approches MDC temporelles. Les résultats ont montré que chacun des schémas est plus performant que le schéma MDC 3-bandes aux décodeurs latéraux pour la plupart des débits considérés et que les variations en qualité entre les trames étaient réduites, conduisant à moins d'artefacts. Cependant, les performances débit-distorsion au décodeur central sont toujours pires que celles du schéma MDC 3-bandes, bien que les mêmes schémas sans information supplémentaire soient plus efficaces. Ceci est dû au fait que, jusque là, nous n'avons toujours considéré les schémas que dans un mécanisme FEC, où l'information Wyner-Ziv est simplement ignorée car redondante au décodeur central. Il est à noter qu'en présence d'un canal de retour, la quantité de données Wyner-Ziv peut être contrôlée par les erreurs observées sur le canal de transmission. Afin d'avoir un réglage plus fin du débit des données Wyner-Ziv qui ont un impact très important sur le compromis entre la qualité des descriptions centrale et latérales, lorsque elles sont utilisées dans le cadre d'un mécanisme FEC, un schéma supplémentaire a été proposé où les trames Wyner-Ziv sont d'abord filtrées temporellement et où seules les sous-bandes de basses fréquences sont codées en Wyner-Ziv et envoyées en tant qu'information supplémentaire dans les descriptions. Les résultats montrent que ce schéma peut-être plus efficace que le schéma MDC 3-bandes pour les débits les plus élevés et les indices de quantification les plus faibles. Les performances débit-distorsion des décodeurs latéraux ne souffrent pas trop du fait qu'aucune information Wyner-Ziv n'a été envoyée pour les sous-bandes de hautes fréquences.

## Chapitre 5

# Codage par descriptions multiples avec information adjacente

*Dans ce chapitre nous allons présenter<sup>1</sup> un schéma de codage par descriptions multiples avec information adjacente basé sur une quantification scalaire à descriptions multiples (MDSQ). Dans un premier temps, les indices générés par la MDSQ sont codés séparément par un codeur de Slepian-Wolf et décodés séparément à l'aide d'une information adjacente qui sert également à la reconstruction des coefficients. Dans un deuxième temps, un décodage croisé est proposé pour décoder conjointement les indices et ainsi profiter de la redondance entre les descriptions.*

### 5.1 Introduction

Dans les schémas de codage MD du chapitre précédent qui utilisaient une information Wyner-Ziv, cette information n'était utilisée qu'aux décodeurs latéraux et ignorée au décodeur central, où elle était considérée redondante. Dans ce chapitre, nous proposons un schéma de codage MD où l'information Wyner-Ziv est utile à la fois au décodeur central et aux décodeurs latéraux. Le codage MD a été présenté comme une généralisation de codage de source sujet à un critère de fidélité pour les systèmes de communication, qui utilise la diversité pour surmonter les problèmes réseau. C'est un outil intéressant pour la communication robuste sur des réseaux avec pertes tels qu'Internet, les réseaux pair à pair, les réseaux sans fil et les réseaux de capteurs. Les approches reposant sur le codage MD peuvent être vues comme des techniques de codage conjoint source-réseau.

Ce chapitre va plus loin en considérant le cas où une information adjacente corrélée avec la source transmise est disponible au récepteur. Le codage MD avec information adjacente au récepteur est particulièrement approprié pour la transmission robuste dans les réseaux de capteurs où des données corrélées sont transmises à un récepteur commun, aussi bien que pour la compression vidéo robuste. La région dans le plan débit-distorsion pour le codage MD lorsque l'information adjacente à propos d'un processus aléatoire corrélé est seulement connue au décodeur a été établie par Diggavi et Vaishampayan [142]. Les expressions

---

1. Ce travail a fait l'objet de deux publications *Multiple description source coding with side information* (O. Crave, C. Guillemot, B. Pesquet-Popescu) dans «Proc. of European Signal Processing Conference (EUSIPCO), Lausanne, August 25-29, 2008» et *Multiple description coding with side information : Practical scheme and iterative decoding* (O. Crave, C. Guillemot, B. Pesquet-Popescu) soumis à «IEEE Transactions on Communications».

---



analytiques des bornes débit-distorsion sont dérivées pour des sources gaussiennes et un modèle de corrélation gaussien, en supposant l'information adjacente commune pour tous les décodeurs. Dans ce chapitre, nous proposons un schéma de codage MD avec information adjacente au récepteur. Il repose à la fois sur des principes de codage MD et sur des principes de codage de Slepian-Wolf. La source en entrée est d'abord quantifiée à l'aide d'une MDSQ : après avoir quantifié la source sur un alphabet donné, deux indices sont attribués aux symboles discrets résultants. L'assignation d'index peut être vue comme une étape de codage MD *sans perte* qui introduit de la redondance ou de la corrélation dans les flux transmis, afin de profiter de la diversité des chemins sur le réseau. Les séquences d'indices résultantes sont codées en Slepian-Wolf, c'est à dire codées séparément et décodées conjointement. Alors que la première étape (MDSQ) joue le rôle de code canal, la seconde (codage de Slepian-Wolf) joue le rôle de compression de source des séquences d'indices quantifiées.

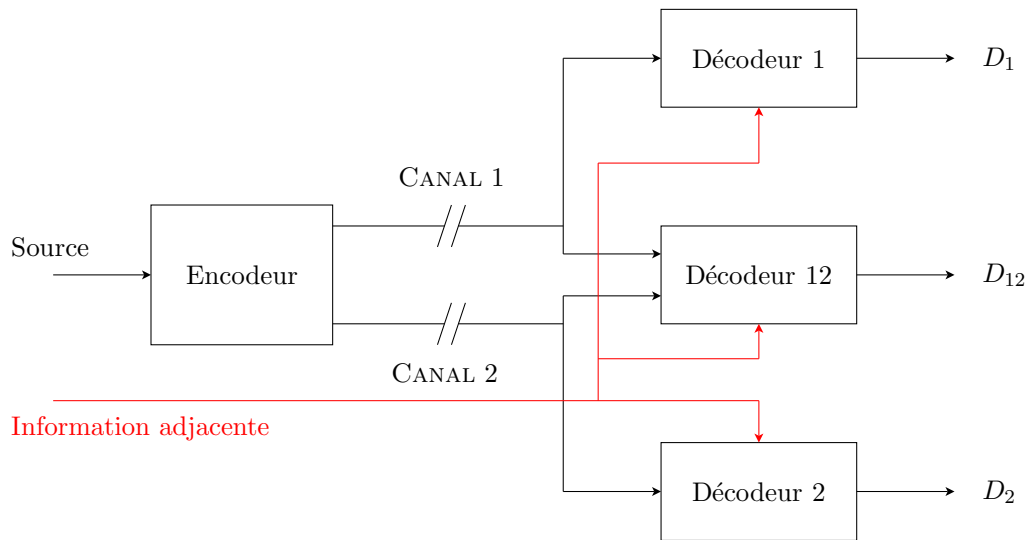


FIGURE 5.1 – Codage de source à deux descriptions avec information adjacente commune connue uniquement du décodeur.

Récemment, Valipour et Lahouti [149] ont proposé une méthode de recuit simulé pour construire un quantificateur vectoriel à descriptions multiples avec une information adjacente disponible au décodeur. Ils ont étudié la performance de ce quantificateur sur des canaux bruités et à pertes et l'ont comparé avec la limite débit-distorsion. Cependant, dans leur système, chaque description est codée indépendamment par un codeur de Slepian-Wolf idéal. Contrairement à eux, dans ce chapitre, nous présentons un schéma complet de codage MD avec information adjacente où des codes de canal sont utilisés comme codes de Slepian-Wolf. La construction de bons quantificateurs pour ce problème n'est pas discutée. A la place, nous étudions l'influence du taux de redondance sur le décodage de Slepian-Wolf, ainsi que l'impact que peut avoir un usage de l'information adjacente au moment de la reconstruction, et nous proposons également une méthode de décodage conjoint source-canal avec information adjacente.

Dans ce chapitre, nous nous intéressons seulement au cas où l'information adjacente est commune et uniquement connue des décodeurs. Dans le cas particulier de deux descriptions, cette approche nous conduit à un schéma équilibré de codage par deux descriptions avec information adjacente commune (voir la figure 5.1). Dans un deuxième temps, le déco-

dage croisé de deux descriptions, qui permet de prendre en compte aussi bien la corrélation avec l'information adjacente que la corrélation entre les deux descriptions est proposé. En supposant les canaux on-off (une description est reçue ou perdue), il a été observé que, pour une certaine quantité de corrélation entre la source  $X$  et l'information adjacente  $Y$ , augmenter la redondance au niveau de la MDSQ n'augmente pas trop le taux de transmission. Lorsque la corrélation entre les deux descriptions et l'information adjacente augmente, le débit du codeur de Slepian-Wolf diminue. Ainsi, la robustesse supplémentaire apportée par l'accroissement de la redondance dans la MDSQ est rendue possible pour un coût en débit modéré.

## 5.2 Contexte théorique

### 5.2.1 Codage sans perte

La dualité existante entre le codage MD *sans perte* et le codage de Slepian-Wolf a été étudié par Stankovic, Cheng et Xiong [150], dans le cas particulier où une description  $D_1$  est transmise à son débit total et utilisée comme information adjacente pour décoder la deuxième description  $D_2$ . Il a été démontré que les points anguleux des régions des débits atteignables de Slepian-Wolf et de codage MD se chevauchent. Dans la configuration équilibrée proposée ici, où chaque description est codée en Slepian-Wolf et décodée avec l'aide de l'information adjacente supplémentaire  $Y$  corrélée avec la source, les deux régions se chevauchent (voir la figure 5.2). Pour le décodeur central, où les deux descriptions sont décodées conjointement, tous les points de débit de la région de Slepian-Wolf sont atteignables. Dans le cas sans perte, le théorème de Slepian-Wolf [77] montre que le débit minimum ( $R = R_{D_1} + R_{D_2}$ ) pour comprimer les deux sources est l'entropie conjointe  $H(D_1, D_2|Y)$  avec :

$$R_{D_1} \geq H(D_1|D_2, Y) \quad (5.1)$$

$$R_{D_2} \geq H(D_2|D_1, Y) \quad (5.2)$$

$$R_{D_1} + R_{D_2} \geq H(D_1, D_2|Y) \quad (5.3)$$

### 5.2.2 Codage avec perte

Le problème de codage MD avec information adjacente a déjà été étudié par Diggavi et Vaishampayan [142]. Ils ont déterminé la région débit-distorsion pour les cas où l'information adjacente est différente pour chaque décodeur ou lorsqu'ils ont accès à une information adjacente commune, et dans les cas où l'encodeur et le décodeur ont connaissance de l'information adjacente ou lorsqu'elle n'est accessible qu'au décodeur. Pour le cas général, la région débit-distorsion est donnée par le théorème suivant [142] :

**Théorème 5** De [142].

Soit  $(X(1), Y(1)), (X(2), Y(2)) \dots$  une séquence sans mémoire (*i.i.d.*) de variables aléatoires définies sur un alphabet fini suivant la fonction densité de probabilités  $Q(x, y) = \mathbb{P}(X = x, Y = y), x \in \mathcal{X}, y \in \mathcal{Y}$ . Seul le décodeur a accès à l'information adjacente  $\{Y(k)\}$ . Alors tous les quintuplets  $(R_1, R_2, D_1, D_2, D_{12})$  sont atteignables s'il existe des variables aléatoires  $(W_0, W_1, W_2, W_{12})$  ayant pour fonction densité de probabilités  $p(x, y, w_0, w_1, w_2, w_{12}) =$

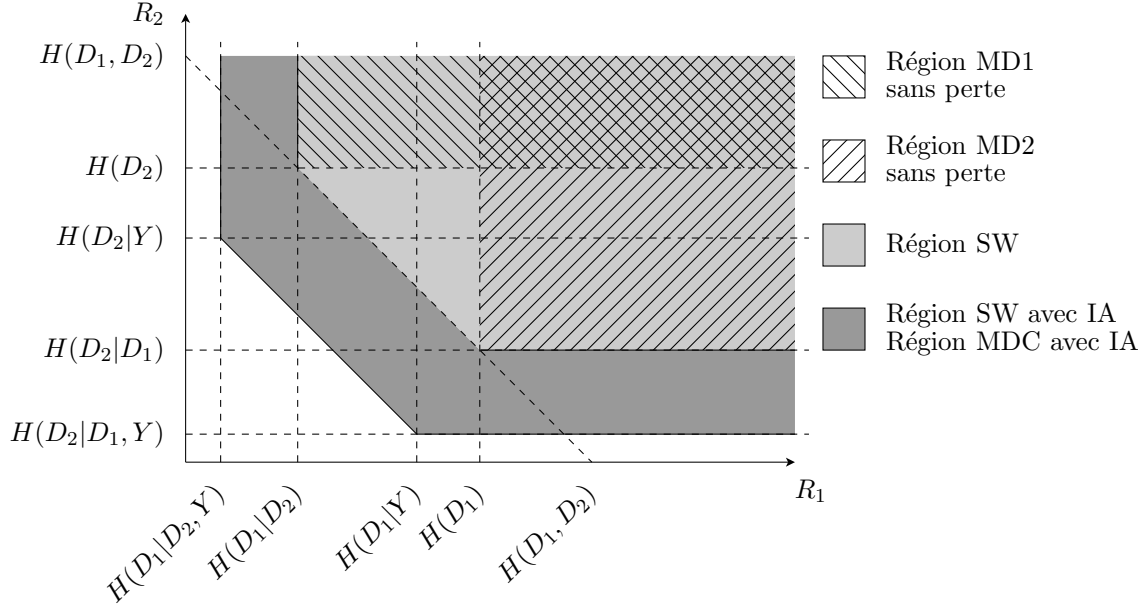


FIGURE 5.2 – Région des débits atteignables pour le problème de codage de deux descriptions avec information adjacente.

$Q(x, y)p(w_0, w_1, w_2, w_{12}|x)$ , telles que

$$\begin{aligned} R_i &> I(X; W_0, W_i|Y), \quad i \in \{1, 2\}, \\ R_1 + R_2 &> 2I(X; W_0|Y) + I(X; W_{12}, W_1, W_2|W_0, Y) + I(W_1; W_2|W_0, Y) \end{aligned} \quad (5.4)$$

et il existe des fonctions de reconstruction  $f_1, f_2, f_{12}$  qui satisfont

$$\begin{aligned} D_i &\geq \mathbb{E}[d_i(X, f_i(Y, W_i, W_0))] \\ D_{12} &\geq \mathbb{E}[d_{12}(X, f_{12}(Y, W_{12}, W_1, W_2, W_0))] \end{aligned} \quad (5.5)$$

où  $I(X; Y)$  est l'information mutuelle de  $X$  et  $Y$ , et  $d_i(\cdot, \cdot)$  et  $d_{12}(\cdot, \cdot)$  sont respectivement les mesures de distorsions utilisées pour les décodeurs latéraux et central.

De plus, les auteurs de [142] ont défini la région débit-distorsion pour le cas gaussien à travers le théorème suivant :

**Théorème 6** De [142].

Soit  $(X(1), Y(1)), (X(2), Y(2)) \dots$  une séquence sans mémoire (i.i.d.) de variables aléatoires conjointement gaussiennes et d'espérance nulle. On peut écrire sans perte de généralité que  $Y(k) = \alpha[X(k) + Z(k)]$ , où  $\alpha > 0$ ,  $\mathbb{E}[X^2] = \sigma_X^2$ ,  $\mathbb{E}[Z^2] = \sigma_Z^2$ . Seul le décodeur a accès à l'information adjacente  $\{Y(k)\}$ . Si les mesures de distorsion sont  $d_m(x, \hat{x}_m) = \|x - \hat{x}_m\|^2$ ,  $m = 1, 2, 12$  où  $\hat{x}_m$  est la version reconstruite de  $x$ , alors l'ensemble des quintuplets atteignables  $(R_1, R_2, D_1, D_2, D_{12})$  est donné par

$$\begin{aligned} D_i &> \sigma_{\mathcal{F}}^2 e^{-2R_i}, \quad i \in \{1, 2\}, \\ D_{12} &> \frac{\sigma_{\mathcal{F}}^2 e^{-2(R_1+R_2)}}{1 - (\sqrt{\tilde{\Pi}} - \sqrt{\tilde{\Delta}})^2} \end{aligned} \quad (5.6)$$

où

$$\begin{aligned}\sigma_{\mathcal{F}}^2 &= \frac{\sigma_X^2 \sigma_Z^2}{\sigma_X^2 + \sigma_Z^2}, \\ \tilde{\Pi} &= \left(1 - \frac{D_1}{\sigma_{\mathcal{F}}^2}\right) \left(1 - \frac{D_2}{\sigma_{\mathcal{F}}^2}\right), \\ \tilde{\Delta} &= \left(\frac{D_1}{\sigma_{\mathcal{F}}^2}\right) \left(\frac{D_2}{\sigma_{\mathcal{F}}^2}\right) - e^{-2(R_1+R_2)}\end{aligned}\tag{5.7}$$

Ce théorème établit que, de la même façon que dans le cas gaussien à description simple [15], la région de débit-distorsion dans le cas gaussien à deux descriptions où l'information adjacente est seulement connue du décodeur est la même que celle obtenue lorsque l'information adjacente est également connue à l'encodeur.

Dans ce chapitre, nous nous intéresserons seulement au scénario dans lequel l'information adjacente est commune et connue seulement au décodeur (voir figure 5.1). Un schéma concret à deux descriptions avec information adjacente au décodeur est présenté dans la section suivante.

### 5.3 Quantification scalaire à descriptions multiples avec information adjacente

La MDSQ consiste à générer deux descriptions grossières d'une source scalaire en utilisant deux (ou plus) quantificateurs scalaires indépendants. Les quantificateurs s'améliorent mutuellement de façon à garantir une faible distorsion pour la description centrale, lorsque les deux descriptions sont reçues au décodeur. Cela est possible grâce à une partition de la ligne réelle et une assignation de paires ordonnées d'indices aux cellules de partitionnement. Le choix d'assignation d'index entraîne la définition des partitions des décodeurs latéraux et donc autorise un compromis systématique entre la distorsion centrale et les distorsions latérales. Des approches concrètes pour construire des matrices d'assignation d'index sont présentées dans [18].

Par exemple, considérons les matrices de la figure 5.3. Les indices  $q \in \{1, 2, \dots, Q\}$  appartenant aux cellules de partition du quantificateur central occupent des positions distinctes dans la matrice et leur sont donc assignés une paire d'indices, qu'on appelle l'indice de ligne  $i \in \{1, 2, \dots, M\}$  et l'indice de colonne  $j \in \{1, 2, \dots, M\}$ . Chacun de ces indices constitue une description latérale envoyée sur un canal distinct. Si les deux canaux sont disponibles au récepteur, le décodage peut être effectué en utilisant les deux indices dans la matrice. Si on n'a accès qu'à une seule description, le décodeur sait que la valeur correcte se trouve parmi les indices d'une certaine ligne ou d'une certaine colonne. La redondance est contrôlée par le choix du nombre de diagonales couvertes par l'assignation d'index.

Le schéma proposé est décrit dans la figure 5.4. Tout d'abord, les flux  $I$  et  $J$  sont générés pour la source  $X$  en utilisant une MDSQ qui consiste en un quantificateur scalaire suivi d'une assignation d'index. Chaque flux ainsi généré est ensuite codé en utilisant un code canal (des turbo codes ou des codes LDPC). Seuls les bits de parité sont envoyés dans les descriptions au décodeur. Le décodeur commence par décoder séparément les indices en utilisant  $Y$  comme information adjacente. Ensuite, selon le nombre de descriptions reçues, une certaine qualité est atteinte pour  $\hat{X}$ , la version reconstruite de  $X$ . Au décodeur central, les indices sont combinés pour obtenir l'intervalle de quantification où se trouve  $X$  et  $\hat{X}_{12}$  est reconstruit avec l'aide de l'information adjacente  $Y$ . Les décodeurs latéraux

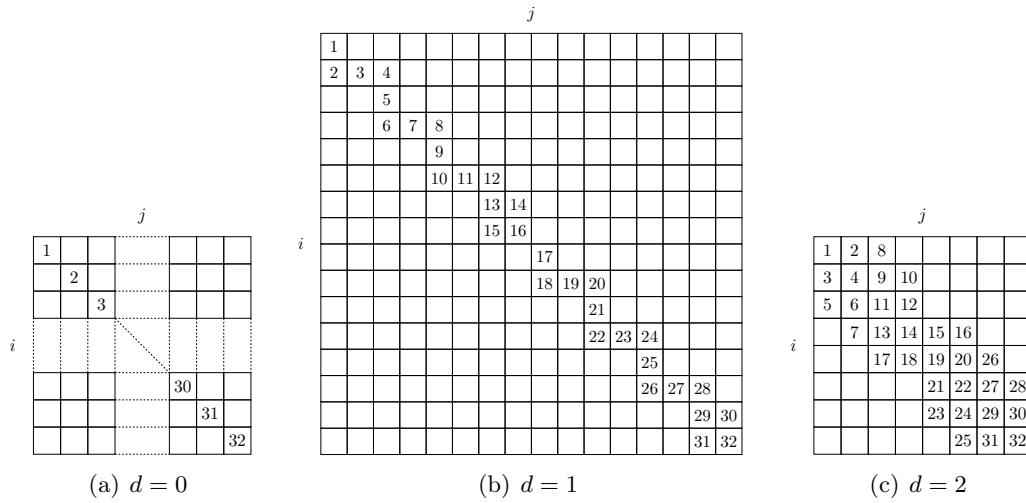


FIGURE 5.3 – Assignment d'index MDSQ pour un dictionnaire central de dimension  $Q = 32$ , avec (a) 1 diagonale ( $d = 0$ ), (b) 3 diagonales ( $d = 1$ ), (c) 5 diagonales ( $d = 2$ ), où  $2d + 1$  est le nombre de diagonales couvertes par l'assignation d'index.

ont seulement accès à un indice pour  $X$  qui correspond soit à une ligne, soit à une colonne dans la matrice d'assignation d'index. Les intervalles de quantification correspondants et l'information adjacente  $Y$  sont utilisés par les décodeurs latéraux pour reconstruire  $\hat{X}_1$  et  $\hat{X}_2$ . Leur qualité dépend de la quantité de redondance introduite par la MDSQ et par la corrélation entre  $X$  et  $Y$ .

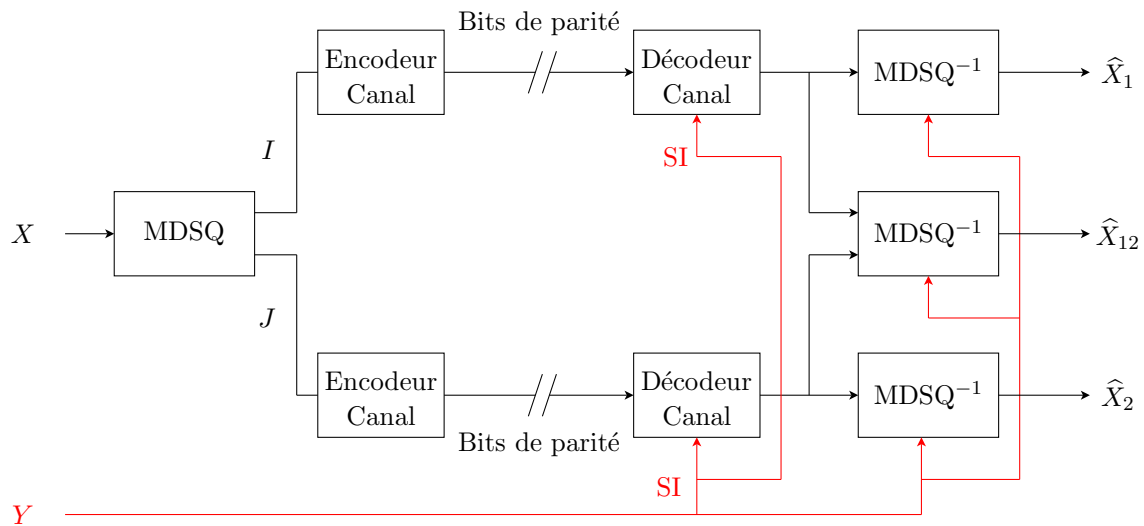


FIGURE 5.4 – Implémentation de la MDSQ avec information adjacente.

## 5.4 Décodage croisé de descriptions multiples avec information adjacente

Pour améliorer les performances du schéma présenté dans la section précédente, nous proposons d'exploiter la redondance entre les descriptions au décodeur central. Cette idée

a été initialement proposée par Srinivasan [48] dans le cas d'un turbo décodage croisé entre les descriptions, puis étudiée dans [49, 151] pour une transmission dans des environnements de communications sans fil. Nous proposons de généraliser cette approche aux turbo codes et aux codes LDPC dans le cas où l'information adjacente représente l'information systématique bruitée. La corrélation entre les descriptions est alors donnée par la matrice d'assignation d'index. Par exemple, considérons la matrice de la figure 5.3(c), nous avons  $P(i = 0|j = 0) = \frac{1}{3}$ ,  $P(i = 0|j = 1) = \frac{1}{4}$ ,  $P(i = 0|j = 2) = \frac{1}{5}$  etc. Cette information de corrélation peut être utilisée comme connaissance *a priori* de  $i$  par le décodeur de  $i$ , la réciproque est vraie pour  $j$ . Le décodeur global doit pouvoir combiner l'information extrinsèque  $L^{\text{out},(1)}$  (respectivement  $L^{\text{out},(2)}$ ) à la sortie du décodeur de  $i$  (resp.  $j$ ) avec la distribution de probabilité conditionnelle  $P(j|i)$  (resp.  $P(i|j)$ ) et envoyer les résultats comme information *a priori* aux décodeur de  $i$  (resp.  $j$ ) (voir la figure 5.5). Le schéma final découlant de cette méthode est présenté sur la figure 5.6.

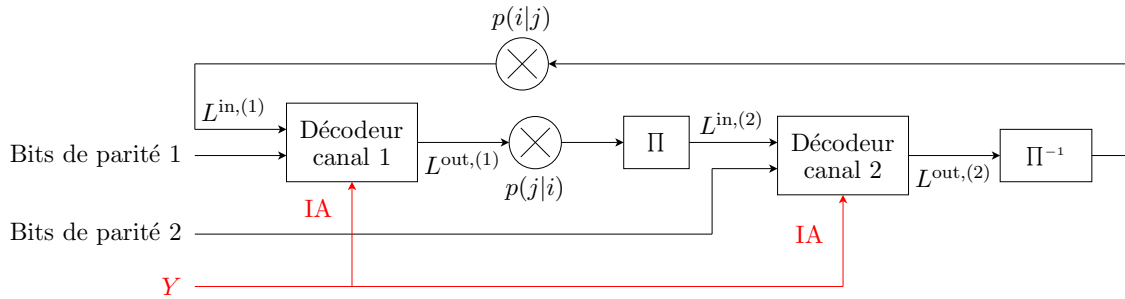


FIGURE 5.5 – Décodage croisé de deux descriptions avec information adjacente.

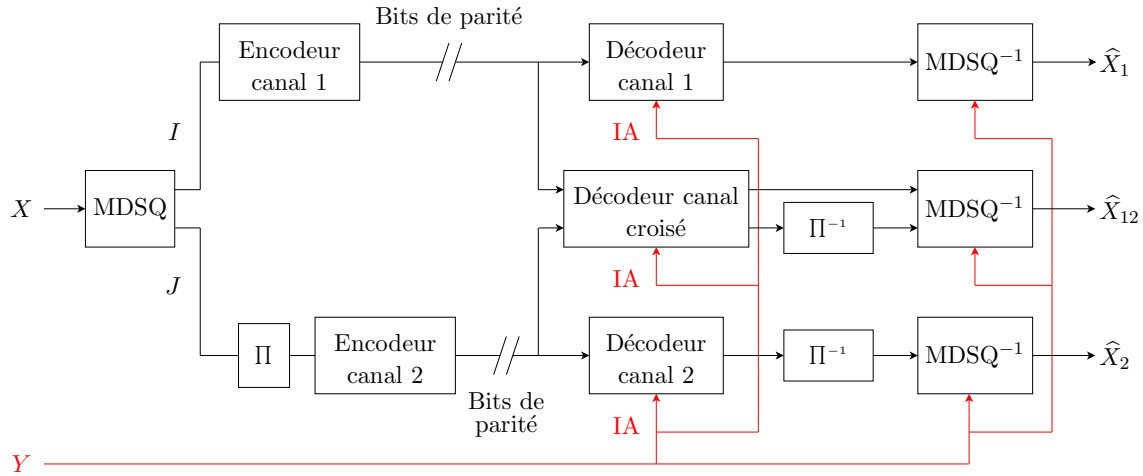


FIGURE 5.6 – Schéma de codage par deux descriptions et décodage croisé au décodeur central.

#### 5.4.1 Décodage croisé de turbo codes avec information adjacente

Nous commençons par décrire le décodage croisé de turbo codes avec information adjacente. Soit  $\{X_n, n = 1, 2, \dots, N\}$ , les échantillons d'une source i.i.d sans mémoire. Cette source est encodée à un taux moyen de  $r$  bits par symbole (bps) et par canal,

par le biais d'un encodeur à descriptions multiples, produisant deux flux corrélés,  $\mathbf{u}^{(s)} = \{u_1^{(s)}, \dots, u_{rN}^{(s)}\}$ ,  $s = 1, 2$ . Chaque flux est encodé séparément par un encodeur canal. Aux récepteurs, un flux de bits d'information est obtenu à partir de l'information adjacente,  $\mathbf{y} = \{y_1, \dots, y_N\}$ . Le récepteur décode séparément chaque plan de bits en commençant par le plan de bits de poids fort. Chaque décodeur génère un rapport de log-vraisemblance (LLR) extrinsèque. Dans le cas des turbo codes, il s'agit de

$$L_{(k-1)r+t}^{\text{out},(s)} = \log \frac{P(u_{(k-1)r+t}^{(s)} = 1|y_k)}{P(u_{(k-1)r+t}^{(s)} = 0|y_k)} - \log \frac{P(u_{(k-1)r+t}^{(s)} = 1)}{P(u_{(k-1)r+t}^{(s)} = 0)}, \quad s = 1, 2 \quad (5.8)$$

où  $k = 1, \dots, N$ ,  $t = 1, \dots, r$ . Il correspond au LLR *a posteriori* moins le LLR *a priori*. Nous ne décrivons que le transfert d'information du premier décodeur au second décodeur. Le décodeur commence par déterminer les probabilités d'avoir un 0 ou un 1 pour chaque symbole de la première description à partir du LLR en sortie du décodeur,

$$P(u_{(k-1)r+t}^{(1)} = 0) = \frac{\exp\left(-L_{(k-1)r+t}^{\text{out},(1)}\right)}{1 + \exp\left(-L_{(k-1)r+t}^{\text{out},(1)}\right)} \quad (5.9)$$

$$P(u_{(k-1)r+t}^{(1)} = 1) = \frac{\exp\left(L_{(k-1)r+t}^{\text{out},(1)}\right)}{1 + \exp\left(L_{(k-1)r+t}^{\text{out},(1)}\right)} \quad (5.10)$$

Ces probabilités permettent de déterminer la probabilité de distribution des bits qui constituent la seconde description.

$$P(u_{(k-1)r+t}^{(2)} = 1) = P(u_{(k-1)r+t}^{(2)} = 1|u_{(k-1)r+t}^{(1)} = 1) \times P(u_{(k-1)r+t}^{(1)} = 1) + P(u_{(k-1)r+t}^{(2)} = 1|u_{(k-1)r+t}^{(1)} = 0) \times P(u_{(k-1)r+t}^{(1)} = 0) \quad (5.11)$$

Les échantillons étant i.i.d, les probabilités conditionnelles ne dépendent pas de  $k$ . Nous pouvons donc écrire,  $\forall k \in \{1, \dots, L\}$ ,

$$P(u_{(k-1)r+t}^{(2)} = 1|u_{(k-1)r+t}^{(1)} = 1) = \sum_{\substack{l:b_t(l)=1 \\ m:b_t(m)=1}} P(j = m|i = l) \quad (5.12)$$

$$P(u_{(k-1)r+t}^{(2)} = 1|u_{(k-1)r+t}^{(1)} = 0) = \sum_{\substack{l:b_t(l)=0 \\ m:b_t(m)=1}} P(j = m|i = l) \quad (5.13)$$

où  $l \in \{1, \dots, M\}$ ,  $m \in \{1, \dots, M\}$  et  $\{b_t(l), t = 1, \dots, r\}$  est la représentation binaire pour l'indice de quantification  $l$ . Les LLRs pour la deuxième description sont obtenus à

partir de la probabilité des bits.

$$P(u_{(k-1)r+t}^{(2)} = 1) = \sum_{\substack{l:b_t(l)=1 \\ m:b_t(m)=1}} P(j = m|i = l) \times P(u_{(k-1)r+t}^{(1)} = 1) \quad (5.14)$$

$$+ \sum_{\substack{l:b_t(l)=0 \\ m:b_t(m)=1}} P(j = m|i = l) \times P(u_{(k-1)r+t}^{(1)} = 0),$$

$$P(u_{(k-1)r+t}^{(2)} = 0) = 1 - P(u_{(k-1)r+t}^{(2)} = 1), \quad (5.15)$$

$$L_{(k-1)r+t}^{\text{in},(2)} = \log \frac{P(u_{(k-1)r+t}^{(2)} = 1)}{P(u_{(k-1)r+t}^{(2)} = 0)} \quad (5.16)$$

Ces LLRs sont utilisés comme information *a priori* pour le décodage de la deuxième description qui, à son tour, génère un LLR extrinsèque pour le premier décodeur. Le transfert d'information du deuxième au premier décodeur est effectué de la même manière. Pour un débit donné pour les bits de parité, un décodage est effectué à chaque nouvelle itération à chaque décodeur, et se poursuit jusqu'à ce que la probabilité de taux d'erreur bit reste stable ou que le nombre d'itérations atteigne un certain seuil (les résultats présentés dans la section 5.6.2 ont été obtenus pour un seuil de 18 itérations). Dans les deux cas, si le décodage n'est pas terminé, davantage de bits de parité sont demandés par le décodeur.

Un entrelaceur avant l'encodage d'une des descriptions est nécessaire afin d'être sûr que l'information contenue dans une description n'est pas redondante avec l'information contenue dans l'autre description pour un débit donné.

#### 5.4.2 Décodage croisé de codes LDPC avec information adjacente

Nous allons maintenant décrire le décodage croisé de codes LDPC avec information adjacente. Pour décrire cet algorithme itératif de décodage par propagation de croyance, nous utiliserons les notations suivantes :

- $\mathbf{v}^{(s)} = \{v_1^{(s)}, \dots, v_N^{(s)}\}$ ,  $s = 1, 2$  sont les nœuds de variable ;
- $f_k \in \{2, 3, \dots\}$ ,  $k = 1, 2, \dots, N$ , est le degré de  $v_k^{(s)}$ ,  $s = 1, 2$  ;
- $q_{k,a}^{\text{out},(s)}(q_{k,a}^{\text{in},(s)}) \in \mathbb{R}$ ,  $k = 1, 2, \dots, N$ ,  $a = 1, 2, \dots, f_k$ ,  $s = 1, 2$ , est le LLR envoyé le long de la  $a$ -ème arête de (vers)  $v_k^{(s)}$

Lorsque les estimés des bits de source du décodeur  $s$  convergent, le LLR suivant est envoyé à partir des nœuds de contrôle vers le nœud de variable  $v_k^{(s)}$ ,

$$L_k^{\text{out},(s)} = \sum_{a=1}^{f_k} q_{k,a}^{\text{in},(s)}, \quad s = 1, 2 \quad (5.17)$$

où  $k = 1, \dots, N$ . Cette information peut être vue comme l'information extrinsèque qui peut être envoyée au décodeur de l'autre description comme information *a priori*. Comme pour les turbo codes, nous ne décrivons que le transfert d'information du premier décodeur au second décodeur. La conversion vers une information *a priori* s'effectue exactement de la même manière qu'avec des turbo codes. Une fois la conversion terminée, les LLRs résultantes sont utilisés par le second décodeur. On définit

$$q_{k,0}^{(2)} = \log \frac{P(u_{(k-1)r+t}^{(2)} = 0|y_k)}{P(u_{(k-1)r+t}^{(2)} = 1|y_k)} + L_k^{\text{in},(2)} \quad (5.18)$$



où  $t$  est le plan de bit courant. Le LLR envoyé à partir du  $k$ -ème nœud variable  $v_k^{(2)}$  le long de la  $a$ -ème arête est alors

$$q_{k,a}^{\text{out},(2)} = q_{k,0}^{(2)} + \sum_{c=1, c \neq a}^{f_k} q_{k,c}^{\text{in},(2)} \quad (5.19)$$

## 5.5 Quantification inverse optimale

On considère le cas de deux sources gaussiennes  $X$  et  $Y$  sans mémoire et corrélées. Le modèle de corrélation est défini ainsi :  $X = Y + Z$ , où  $Z$  est un bruit gaussien avec une espérance nulle et une variance  $\sigma_Z^2$ . Soient  $Q$  le nombre de niveaux de quantification et  $z_0 < z_1 < \dots < z_Q$  les niveaux de quantification de la source  $x$ . Nous cherchons la valeur de  $\hat{x}_{\text{opt}}$  qui minimise la moyenne du carré de l'erreur d'estimation de  $x$  (aux récepteurs central et latéraux).  $\hat{x}_{\text{opt}}$  est donné par :

$$\hat{x}_{\text{opt}} = E[x|x \in \bigcup_{k=1}^K [z_i^k, z_{i+1}^k), y] = \frac{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} x f_{X|Y}(x) dx}{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} f_{X|Y}(x) dx} \quad (5.20)$$

$$= \frac{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} x p_Z(x-y) dx}{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} p_Z(x-y) dx} = \frac{\sum_{k=1}^K g(k)}{\sum_{k=1}^K h(k)} \quad (5.21)$$

où le nombre  $K$  d'intervalles de quantification pour un  $x$  donné dépend du nombre de descriptions reçues et du nombre de diagonales dans la matrice d'assignation d'index. Au décodeur central,  $K = 1$ . Aux décodeurs latéraux,  $K$  est le nombre de coefficients quantifiés dans la ligne ou la colonne de la matrice d'assignation d'index qui correspond à l'indice reçu. Connaissant l'expression de la p.d.f. du bruit de corrélation entre  $X$  et  $Y$ , on en déduit

$$\begin{aligned} h(k) &= \int_{z_i^k}^{z_{i+1}^k} \frac{1}{\sigma_Z \sqrt{2\pi}} e^{-(x-y)^2/2\sigma_Z^2} dx \\ &= \left[ \frac{1}{2} \left( 1 + \operatorname{erf} \left( \frac{x-y}{\sigma_Z \sqrt{2}} \right) \right) \right]_{z_i^k}^{z_{i+1}^k} \\ &= \frac{1}{2} \left( \operatorname{erf} \left( \frac{z_{i+1}^k - y}{\sigma_Z \sqrt{2}} \right) - \operatorname{erf} \left( \frac{z_i^k - y}{\sigma_Z \sqrt{2}} \right) \right) \end{aligned} \quad (5.22)$$

et

$$\begin{aligned} g(k) &= \int_{z_i^k}^{z_{i+1}^k} \frac{1}{\sigma_Z \sqrt{2\pi}} x e^{-(x-y)^2/2\sigma_Z^2} dx \\ &= \left[ x \times \frac{1}{2} \left( 1 + \operatorname{erf} \left( \frac{x-y}{\sigma_Z \sqrt{2}} \right) \right) \right]_{z_i^k}^{z_{i+1}^k} - \int_{z_i^k}^{z_{i+1}^k} \frac{1}{2} \left( 1 + \operatorname{erf} \left( \frac{x-y}{\sigma_Z \sqrt{2}} \right) \right) dx. \end{aligned}$$

Soient  $a = \frac{z_{i+1}^k - y}{\sigma_Z \sqrt{2}}$  et  $b = \frac{z_i^k - y}{\sigma_Z \sqrt{2}}$ ,

$$2g(k) = z_{i+1}^k \times (1 + \operatorname{erf}(a)) - z_i^k \times (1 + \operatorname{erf}(b)) - z_{i+1}^k + z_i^k + \int_{z_i^k}^{z_{i+1}^k} \operatorname{erf}\left(\frac{x - y}{\sigma_Z \sqrt{2}}\right) dx.$$

En effectuant la substitution  $u = \frac{x - y}{\sigma_Z \sqrt{2}}$ , on obtient  $du = \frac{1}{\sigma_Z \sqrt{2}} dx$  et

$$\begin{aligned} 2g(k) &= z_{i+1}^k \times \operatorname{erf}(a) - z_i^k \times \operatorname{erf}(b) - \sigma_Z \sqrt{2} \left[ u \times \operatorname{erf}(u) + \frac{e^{-u^2}}{\sqrt{\pi}} \right]_b^a \\ &= z_{i+1}^k \times \operatorname{erf}(a) - z_i^k \times \operatorname{erf}(b) - \sigma_Z \sqrt{2} (a \times \operatorname{erf}(a) - b \times \operatorname{erf}(b)) + \frac{\sigma_Z \sqrt{2}}{\sqrt{\pi}} (e^{-b^2} - e^{-a^2}) \\ &= y \times (\operatorname{erf}(a) - \operatorname{erf}(b)) + \frac{\sigma_Z \sqrt{2}}{\sqrt{\pi}} (e^{-b^2} - e^{-a^2}). \end{aligned} \quad (5.23)$$

En changeant  $g(k)$  et  $h(k)$  dans (5.21) par les équations (5.23) et (5.22), on obtient finalement :

$$\hat{x}_{opt} = y + \frac{\frac{\sigma_Z \sqrt{2}}{\sqrt{\pi}} \sum_{k=1}^K (e^{-b^2} - e^{-a^2})}{\sum_{k=1}^K (\operatorname{erf}(a) - \operatorname{erf}(b))}. \quad (5.24)$$

## 5.6 Résultats expérimentaux sur des sources gaussiennes

Les résultats ont été obtenus pour des séquences de 1584 échantillons d'une source gaussienne d'espérance nulle et de variance unité pour  $Y$ . Les échantillons ont tout d'abord été traités par l'encodeur MDSQ, qui consiste en un quantificateur de Lloy-Max générant 32 intervalles de quantification, suivi d'une assignation d'index similaire à celles présentées sur la figure 5.3, avec 1, 3 et 5 diagonales, correspondant à 5, 4 et 3 bps en sortie pour  $i$  et  $j$ . Les matrices d'assignation d'index ont été construites en utilisant une stratégie de création de tables *indexation emboîtée* [152] qui permet d'améliorer les performances en débit-distorsion lorsque tous les plans de bits ne sont pas reçus. Certains symboles ont été retirés manuellement pour conserver un nombre fixe de niveaux de quantification, ce qui implique que les matrices sont loin d'être optimales. Cependant, la non optimalité de la MDSQ n'est pas le point central de cette étude et ne devrait pas avoir d'influence sur notre travail d'analyse.

Les séquences de parité stockées dans la mémoire-tampon sont transmises en petite quantité à la demande du décodeur via un canal de retour. Lorsque le taux d'erreur bit à la sortie du décodeur atteint un certain seuil, des bits de parité supplémentaires sont demandés à l'encodeur. Cela revient à contrôler le débit du code en sélectionnant différents modèles de poinçonnage à la sortie du turbo code. Le taux d'erreur bit est estimé à partir du rapport de log-vraisemblance sur les bits en sortie du turbo décodeur. Les performances peuvent être considérées comme étant les mêmes aux deux décodeurs latéraux (le schéma de codage MD étant équilibré). Par la suite, les performances latérales seront toujours le résultat de la moyenne des performances des deux décodeurs latéraux. Nous ne présentons que les résultats obtenus avec des turbo codes. Chaque description est codée en utilisant

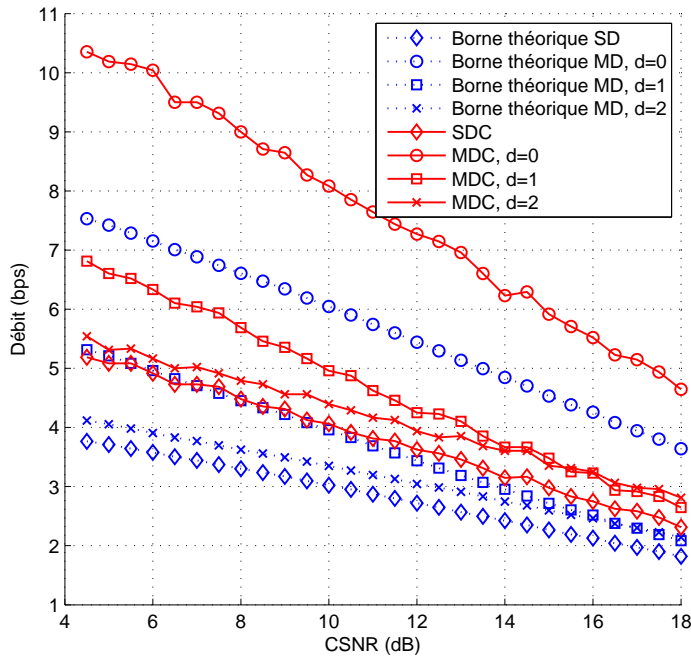


FIGURE 5.7 – Comparaison des débits obtenus par les schémas de codage SD et MD.

un turbo code qui consiste en deux codes convolutifs  $\frac{1}{2}$  utilisés sous leur forme récursive systématique. Le code est essentiellement le même que celui présenté dans [153]. Chaque décodeur réalise 18 itérations de l'algorithme MAP.

### 5.6.1 MDSQ avec information adjacente

Les figures 5.7 et 5.8 montrent les performances obtenues par les schémas de codage SD et MD pour 28 valeurs de rapport signal sur bruit de corrélation<sup>2</sup> (CSNR), pour deux séquences d'échantillons. Comme on peut le voir sur la figure 5.8, les valeurs de SNR restant les mêmes pour le schéma de codage SD, toutes les techniques de codage MD au décodeur central, et pour le schéma de codage MD avec  $d = 0$  aux décodeurs latéraux. L'utilisation de l'information adjacente n'a presque aucun effet sur ces valeurs, mis à part pour les valeurs de CSNR les plus élevées, tandis que son impact est plus grand sur les performances des décodeurs latéraux pour les schémas de codage MD avec  $d = \{1, 2\}$ . La figure 5.7 montre les débits obtenus par les différents schémas. Pour les trois matrices d'assignation d'index considérées, nous avons tracé le nombre théorique minimum de bits par symbole pour le cas où le décodage des descriptions est fait de façon disjointe. Le théorème de Slepian-Wolf [77] nous dit que le nombre minimum de bits par symbole que l'on peut atteindre pour comprimer une source  $X$  lorsque seul le décodeur a accès à une source corrélée  $Y$  est  $R_X \geq H(X|Y)$ . Pour le schéma de codage SD, cette limite est donnée par  $R_X \geq H(X_Q|Y)$ , où  $X_Q$  est la version quantifiée de  $X$ ; pour les schémas de codage MD, elle correspond à  $R_X \geq H(I|Y) + H(J|Y)$ . Comme prévu, lorsque l'on augmente le nombre de diagonales, la redondance introduite par la MDSQ devient plus faible et le

2.  $CSNR = 10 \log_{10}(\sigma_Y^2/\sigma_Z^2)$

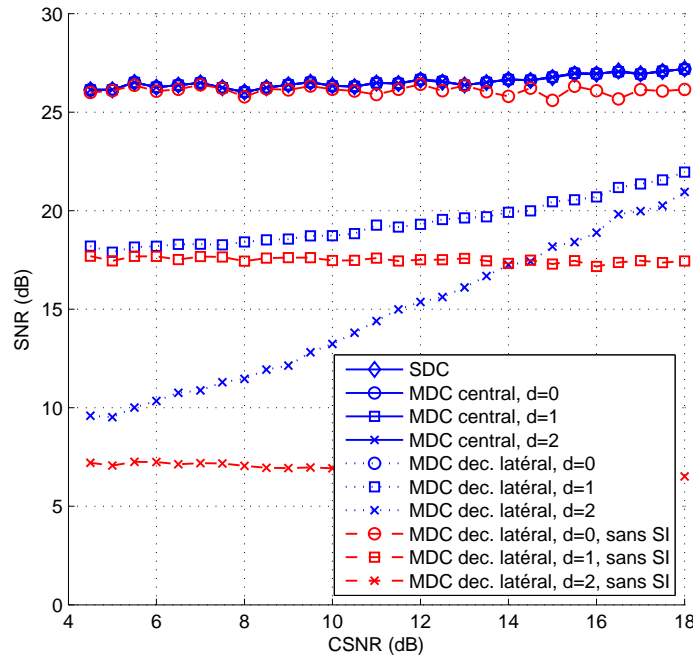


FIGURE 5.8 – Comparaison des SNR obtenus par les schémas de codage SD et MD.

débit se rapproche de celui que l'on obtient avec le schéma de codage MD. On remarque que l'impact de la corrélation entre  $Y$  et les descriptions  $I$ ,  $J$  ne dépend pas seulement du CSNR, mais aussi du nombre de diagonales. Cet effet est clairement visible lorsque les courbes correspondant aux schémas de codage MD pour  $d = 1$  et  $d = 2$  se croisent aux plus fortes valeurs de CSNR. Le même effet est observé sur le schéma proposé : lorsque  $d$  augmente, le débit devient plus faible, sauf pour  $d = 2$  et des valeurs de CSNR supérieures à 15 dB, auquel cas le schéma de codage MD avec  $d = 1$  est plus performant.

La figure 5.9 montre les SNR théoriques atteignables selon le théorème 6 pour les cas de codage MD et SD en utilisant les débits de la figure 5.7. La limite théorique est la même pour le schéma de codage SD et le décodeur latéral du schéma de codage MD avec  $d = 0$ . On remarque que pour les schémas de codage SD et MD avec  $d = 0$ , les valeurs des bornes théoriques décroissent lorsque le CSNR augmente, tandis qu'elles restent à peu près stables pour  $d = 1$  et augmentent pour  $d = 2$ . Sachant d'après la figure 5.8 que le SNR aux décodeurs latéraux de tous les schémas reste à peu près le même quelque soit la valeur du CSNR, on peut conclure que l'information adjacente est mieux utilisée avec de faibles valeurs de  $d$ . On observe également que pour le décodeur central du schéma de codage MD avec  $d = 2$ , le SNR atteint presque la borne théorique mais seulement pour les valeurs de CSNR les plus faibles.

### 5.6.2 Décodage croisé de descriptions multiples avec information adjacente

On se propose dans cette section d'étudier l'influence de l'utilisation du décodage croisé sur les performances du décodeur central. La figure 5.10 permet une comparaison des schémas de codage SD et MD avec ou sans décodage croisé pour différentes valeurs de  $d$ . Ces

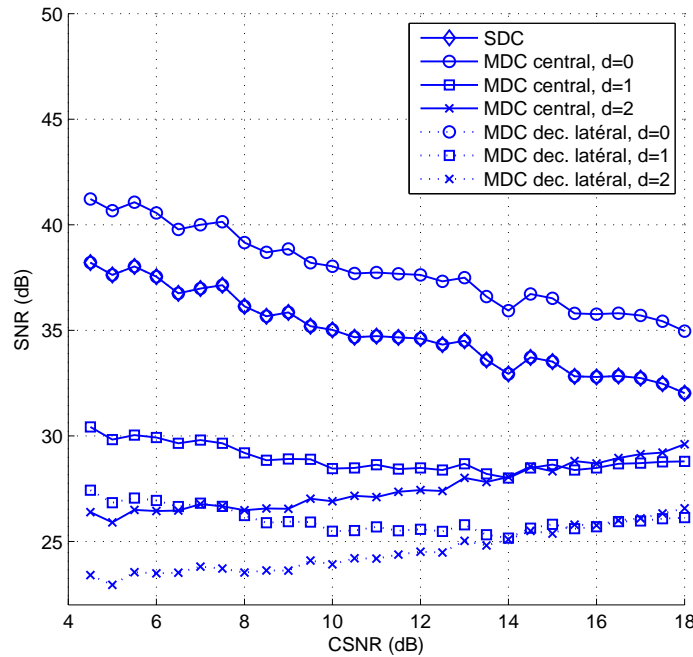


FIGURE 5.9 – SNR atteignables par les schémas de codage SD et MD.

résultats montrent que le bénéfice que l'on peut tirer d'une utilisation du décodage croisé s'accroît lorsque  $d$  décroît. Pour  $d = 0$ , le décodage croisé peut offrir une économie de débit de l'ordre de 2 bps pour les plus faibles valeurs de CSNR, tandis que pour  $d = 1$  et  $d = 2$ , l'économie est seulement de l'ordre de 0.65 et 0.13 bps respectivement. Ces résultats sont consistants avec le fait que plus les descriptions sont corrélées, plus le fait de faire circuler l'information entre elles aura un impact positif sur les performances. On note que pour  $d = 0$ , le débit avec décodage croisé devient plus faible que le débit théorique sans décodage croisé donné par la figure 5.7; ceci montre bien qu'en exploitant efficacement la corrélation entre  $I$  et  $J$  au décodeur, le débit central peut devenir inférieur à  $H(I|Y) + H(J|Y)$ .

Les figures 5.11 et 5.12 ont été obtenues en utilisant 100 séquences de 1584 échantillons. Elles montrent les courbes débit-distorsion des décodeurs central et latéraux pour une valeur de CSNR de 10 dB. Chaque point sur les courbes a été obtenu pour un nombre différent de plans de bits parfaitement décodés, c'est-à-dire que le premier point correspond au plan de bits de poids fort (MSB) parfaitement décodé, le deuxième aux MSB et second plan de bits, etc. Les plans de bits qui n'ont pas été décodés sont remplacés par les plans de bits correspondants de l'information adjacente. Le nombre de points sur chaque courbe correspond au nombre de bits nécessaires pour représenter les indices (5 pour le codage SD et  $d = 0$ , 4 pour  $d = 1$  et 3 pour  $d = 2$ ). Les courbes centrale et latérale du schéma de codage MD avec  $d = 0$  se chevauchent parfaitement. A bas débit, lorsque tous les plans de bits ne sont pas parfaitement décodés, les décodeurs centraux peuvent offrir de moins bonnes performances débit-distorsion que les décodeurs latéraux correspondants. Grâce au décodage croisé, les performances centrales augmentent et la quantité de redondance a moins d'influence sur celles-ci, en particulier à bas débit.

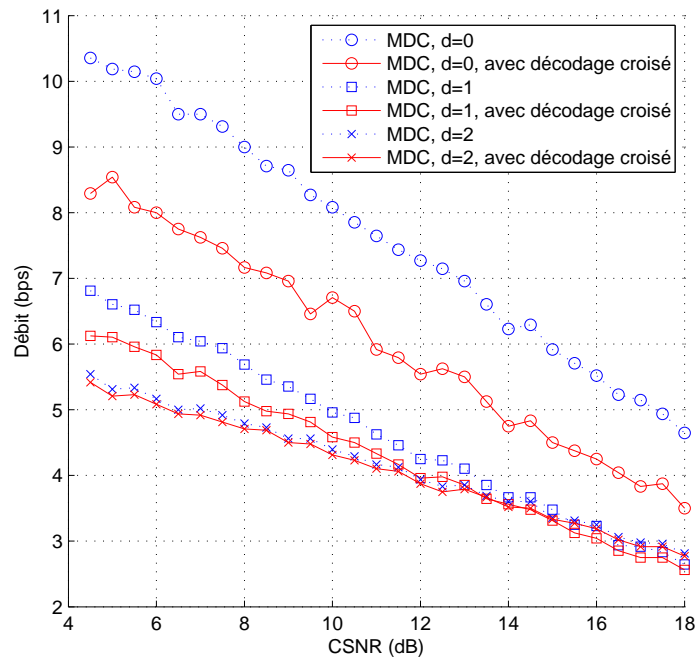


FIGURE 5.10 – Comparaison des débits centraux obtenus par les schémas de codage MD avec et sans décodage croisé pour différentes valeurs de  $d$ .

## 5.7 Conclusion

Nous avons présenté un schéma équilibré de codage par deux descriptions avec une information adjacente uniquement connue aux décodeurs et où cette dernière est la même pour tous les décodeurs. Les résultats de simulation dans le cas gaussien ont montré que l'approche proposée peut être utilisée pour améliorer la robustesse des schémas de codage de source distribué sans sacrifier les performances en débit-distorsion. Il a été montré qu'en effectuant un décodage croisé, on pouvait réduire efficacement le débit au décodeur central en exploitant la redondance entre les descriptions. Dans le chapitre suivant nous verrons une application des principes évoqués dans ce chapitre en vidéo, avec un schéma de codage MD avec information adjacente, où l'information adjacente est dégradée aux décodeurs latéraux.

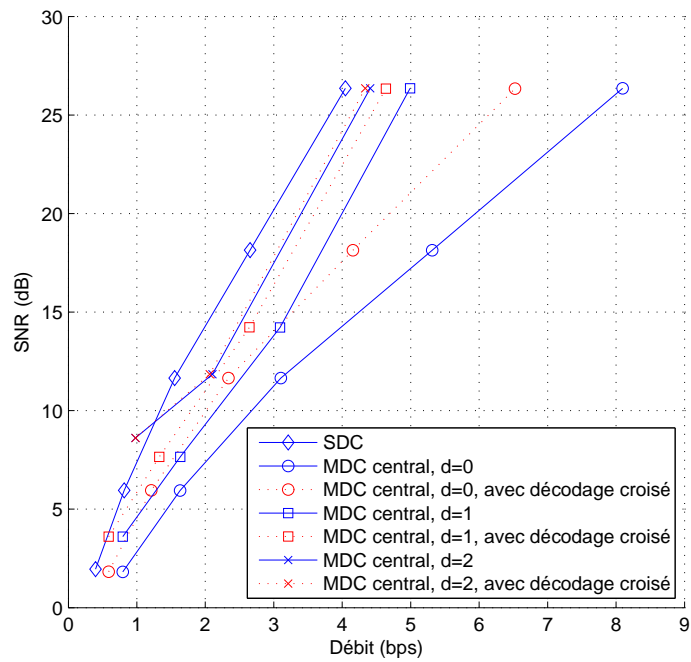


FIGURE 5.11 – Comparaison des performances débit-distorsion obtenues par les schémas de codage SD et MD pour une valeur de CSNR de 10 dB au décodeur central.

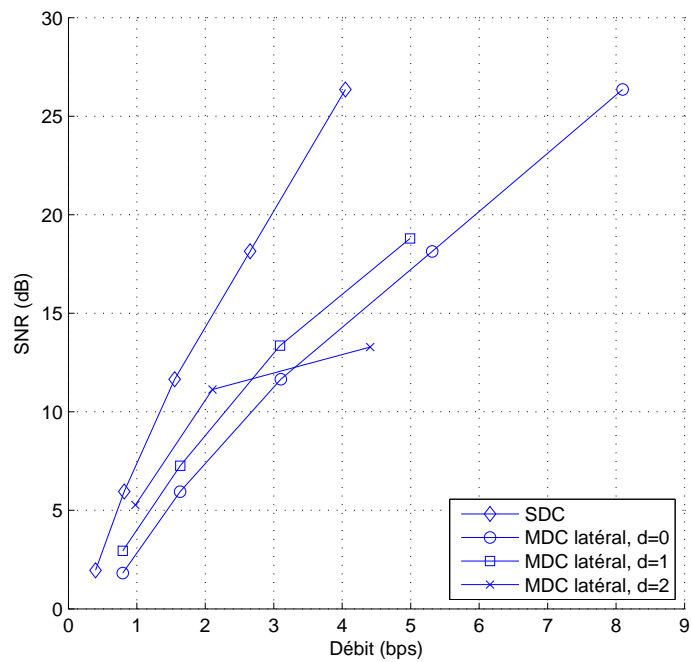


FIGURE 5.12 – Comparaison des performances débit-distorsion obtenues par les schémas de codage SD et MD pour une valeur de CSNR de 10 dB aux décodeurs latéraux.

## Chapitre 6

# Codage vidéo distribué robuste

*Dans ce chapitre nous allons présenter<sup>1</sup> une manière d'améliorer la robustesse en codage vidéo distribué tout en conservant de bonnes performances débit-distorsion. La séquence d'entrée est structurée en groupes d'images, chacun contenant une trame clé et une trame codée en Wyner-Ziv. Pour chaque trame, deux descriptions sont générées par un quantificateur scalaire à descriptions multiples (MDSQ). Cette technique permet d'obtenir deux descriptions équilibrées contenant chacune une information partielle sur chaque trame dans la vidéo d'entrée. Au décodeur central, le décodage itératif de deux descriptions avec information adjacente est présenté. Nous étudions l'influence du décodage croisé et du taux de redondance sur les performances.*

### 6.1 Introduction

Dans le chapitre précédent nous avons présenté un schéma de codage par descriptions multiples avec information adjacente commune et l'avons testé sur des sources gaussiennes. Nous allons maintenant voir une application de ces principes à la vidéo. Une différence notable entre le schéma du chapitre précédent et celui que nous allons présenter ici est que, dans le cas de la vidéo, la qualité de l'information adjacente diffère selon le nombre de descriptions reçues.

En codage vidéo distribué, la méthode la plus couramment utilisée pour mettre en oeuvre les principes de codage de source distribué consiste à séparer temporellement les trames en deux sous-groupes de trames : des trames clé codées conventionnellement et des trames codées en Wyner-Ziv. Les trames clé sont utilisées par le décodeur pour générer une information adjacente servant à décoder les trames codées en Wyner-Ziv. Le décodeur de Slepian-Wolf (généralement basé sur des principes de codage canal) se sert de l'information adjacente et les bits de parité reçus pour décoder le flux de symboles codés en Wyner-Ziv. La question que nous nous posons dans ce chapitre est la suivante : comment améliorer efficacement la robustesse d'un tel schéma tout en conservant les propriétés qui le caractérisent ? Nous voulons notamment faire en sorte que la complexité de l'encodeur soit faible et que les descriptions décodées ne soient pas sujettes à des fluctuations de qualité au cours du temps.

---

1. Ce travail a fait l'objet de deux publications *Multiple description video coding and iterative decoding of LDPCA codes with side information* (O. Crave, C. Guillemot, B. Pesquet-Popescu) soumis à «IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2009» et *Robust video coding based on multiple description scalar quantization with side information* (O. Crave, B. Pesquet-Popescu, C. Guillemot) soumis à «IEEE Transactions on Circuits and Systems for Video Technology».

---



Le schéma proposé repose sur une utilisation séparée de la quantification scalaire à descriptions multiples (MDSQ) pour le codage des trames clé et des trames codées en Wyner-Ziv. Une description des trames clé est ensuite combinée avec une description des trames codées en Wyner-Ziv pour former une description de la séquence vidéo d'origine. Les deux descriptions ainsi créées sont envoyées sur deux canaux on-off (une description est reçue ou perdue). Au récepteur, si les deux descriptions sont reçues, l'information adjacente sera de bonne qualité et servira à décoder les deux descriptions de trames codées en Wyner-Ziv. Si une seule description est reçue, l'information adjacente sera de moins bonne qualité et servira à décoder les indices de l'unique description des trames Wyner-Ziv. Afin de réduire le débit au décodeur central, un décodage croisé sera envisagé pour décoder conjointement les deux descriptions de trames codées en Wyner-Ziv.

## 6.2 Codage vidéo par descriptions multiples avec information adjacente

Le problème posé dans ce chapitre est un problème de codage par descriptions multiples avec une information adjacente connue uniquement du décodeur et où la qualité de l'information adjacente diffère selon le nombre de descriptions reçues (voir la figure 6.1).

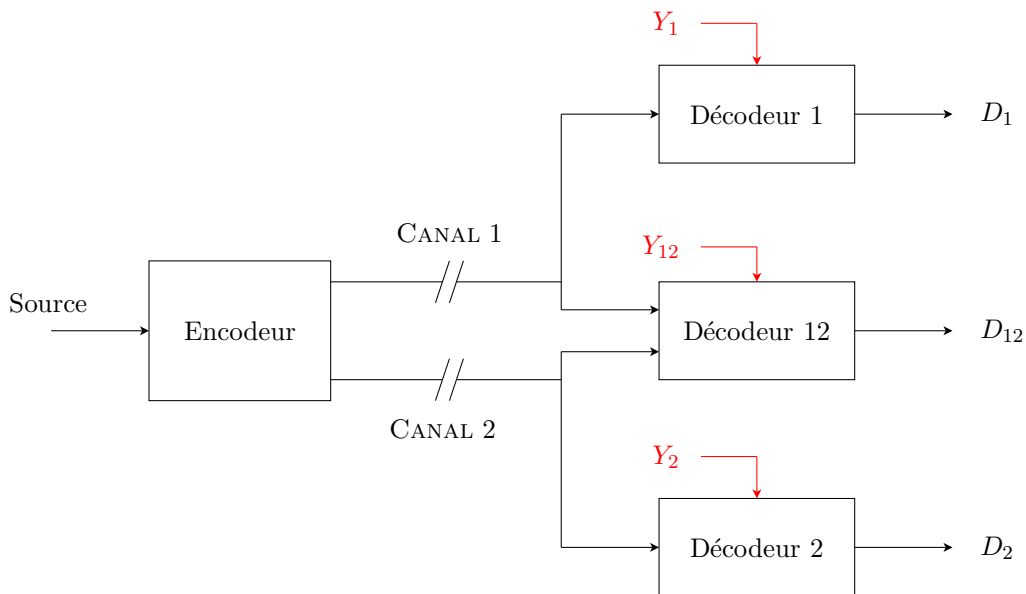


FIGURE 6.1 – Codage par deux descriptions avec information adjacente pour chaque décodeur.

### 6.2.1 Principes

Diggavi et Vaishampayan [142] ont défini la région débit-distorsion pour ce type de problème dans le cas général avec le théorème suivant :

**Théorème 7** De [142]. Soit  $(X(1), Y_1(1), Y_2(1), Y_{12}(1)), \dots$  une séquence de variables aléatoires i.i.d. définies sur un alphabet fini suivant la fonction densité de probabilités  $Q(x, y_1, y_2, y_{12})$ . Soit le décodeur le seul à avoir accès à l'information adjacente  $\{Y_1(k), Y_2(k), Y_{12}(k)\}$ . De

plus, soit l'information adjacente dégradée, c'est-à-dire,  $p(x, y_t, y_{12}) = p(x)p(y_{12}|x)p(y_t|y_{12})$ ,  $t = 1, 2$ . Alors tous les tuples  $(R_1, R_2, D_1, D_2, D_{12})$  sont atteignables s'il existe des variables aléatoires  $(W_0, W_1, W_2, W_{12})$  ayant pour fonction densité de probabilités  $p(x, y_1, y_2, y_{12}, w_0, w_1, w_2, w_{12}) = Q(x, y_1, y_2, y_{12})p(w_0, w_1, w_2, w_{12}|x)$ , telles que

$$\begin{aligned}
R_1 &> \max\{I(X; W_0|Y_1), I(X; W_0|Y_2)\} \\
&+ I(X; W_1|Y_1, W_0) \\
R_2 &> \max\{I(X; W_0|Y_1), I(X; W_0|Y_2)\} \\
&+ I(X; W_2|Y_2, W_0) \\
R_1 + R_2 &> 2 \max\{I(X; W_0|Y_1), I(X; W_0|Y_2)\} \\
&+ I(X; W_{12}|W_1, W_2, W_0, Y_{12}) + I(X; W_1, W_2|W_0) \\
&+ I(W_1; W_2|W_0) - I(W_1; Y_1|W_0) - I(W_2; Y_2|W_0)
\end{aligned} \tag{6.1}$$

et il existe des fonctions de reconstruction  $f_1, f_2, f_{12}$  qui satisfont

$$\begin{aligned}
D_1 &\geq \mathbb{E}[d_1(X, f_1(Y_1, W_1, W_0))] \\
D_2 &\geq \mathbb{E}[d_2(X, f_2(Y_2, W_2, W_0))] \\
D_{12} &\geq \mathbb{E}[d_{12}(X, f_{12}(Y_{12}, W_{12}, W_1, W_2, W_0))].
\end{aligned} \tag{6.2}$$

où  $I(X; Y)$  est l'information mutuelle de  $X$  et  $Y$ , et  $d_i(\cdot, \cdot)$  et  $d_{12}(\cdot, \cdot)$  sont respectivement les mesures de distorsion utilisées pour les décodeurs latéraux et central.

Wang *et al.* [143, 144] ont défini la région débit-distorsion dans le cas de sources gaussiennes et ont montré que lorsque l'information adjacente n'est pas disponible à l'encodeur, il y a des pertes.

L'idée générale est illustrée sur la figure 6.2, avec l'exemple de deux descriptions. Au transmetteur, un sous-ensemble de trames régulièrement espacées dans la séquence est comprimé à l'aide d'un codeur vidéo de trames clé en mode Intra. Les trames entre les trames clé sont codées en Wyner-Ziv. Chaque codeur génère deux descriptions. Une description de trames clé est combinée avec une description de trames Wyner-Ziv pour créer une description de l'intégralité de la séquence d'origine. Chaque description résultante est envoyée sur un canal à effacement au décodeur. Du côté du décodeur, si une seule description est reçue, la séquence entière peut être reconstruite avec une qualité acceptable. Lorsque les deux descriptions sont reçues, les descriptions des trames clé et des trames Wyner-Ziv sont décodées conjointement et permettent d'obtenir une meilleure qualité qu'aux décodeurs latéraux pour chaque trame.

## 6.2.2 Codec vidéo

L'encodeur vidéo est décrit sur la figure 6.3. La vidéo à coder est tout d'abord décomposée en deux séquences : une contiendra les trames clé, et l'autre les trames Wyner-Ziv. Par exemple, les trames paires pourraient aller dans une séquence, et les trames impaires dans l'autre. Les trames clé sont codées en utilisant un codeur Intra (ici, JPEG2000), qui consiste en une DWT suivie d'une quantification uniforme à zone morte. Les coefficients ainsi quantifiés sont ensuite convertis en une paire d'indices par une assignation d'index comme expliquée au chapitre précédent. Pour finir, les deux séquences d'indices sont codées par des codes à longueur variable. L'optimisation débit/distorsion est faite indépendamment pour chaque description comme si elles étaient issues d'un codage SD. Les trames

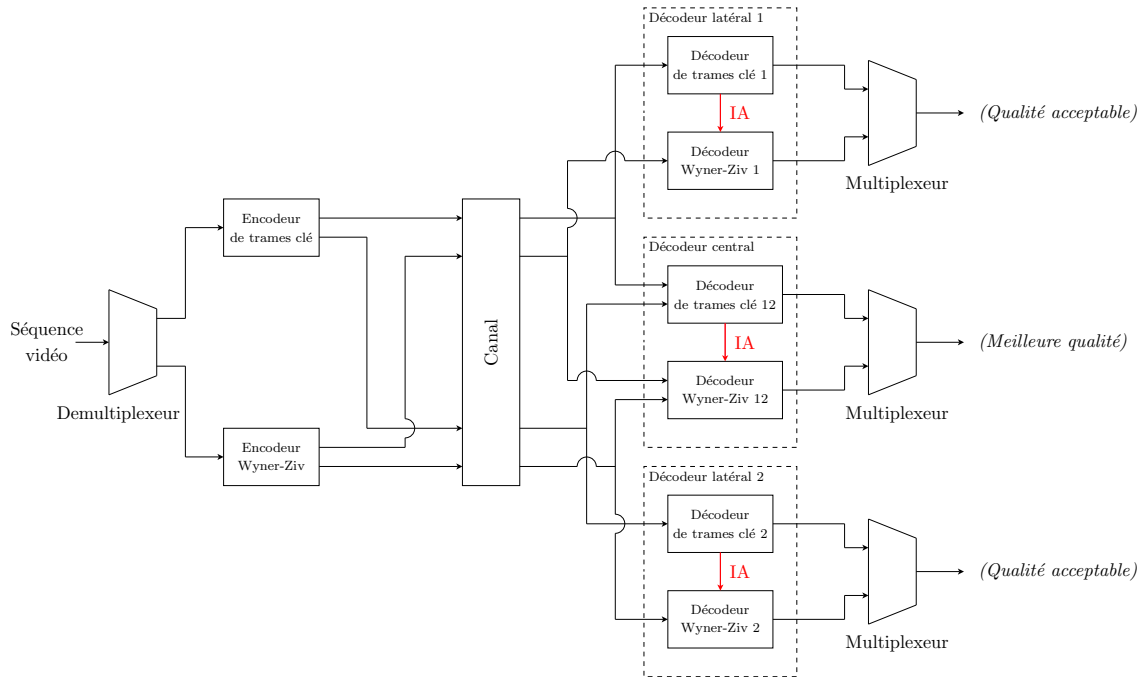


FIGURE 6.2 – Codec vidéo à deux descriptions avec information adjacente.

Wyner-Ziv passent par la même transformée spatiale, suivie d'une quantification uniforme. Une paire d'indices est attribuée à chaque coefficient quantifié à l'aide d'une matrice d'assignation d'index qui peut être la même ou différer de celle utilisée pour les trames clé. Les plans de bits des deux séquences d'indices sont codées séparément (du plus significatif au moins significatif) en utilisant un codeur de Slepian-Wolf. Seule l'information de parité est conservée et transmise dans les descriptions. Une description de trames clé est combinée avec une description de trames Wyner-Ziv, de façon à obtenir deux descriptions contenant des informations sur la totalité de la séquence.

Au décodeur central, représenté sur la figure 6.4, les descriptions des trames clé sont décodées entropiquement séparément, puis les deux séquences d'indices sont combinées en utilisant la matrice d'assignation d'index inverse pour obtenir les coefficients quantifiés. Les trames clé sont finalement décodées après reconstruction et transformation inverse des coefficients d'ondelettes quantifiés. Pour décoder les trames Wyner-Ziv, une information adjacente est obtenue par interpolation des trames clé. L'information de parité est transmise en petites quantités à la demande du décodeur via le canal de retour. Dans une première approche, les deux descriptions Wyner-Ziv sont décodées séparément à l'aide de la même information adjacente. Dans une seconde approche, nous proposons d'utiliser l'algorithme de décodage croisé décrit dans le chapitre précédent, 5.4, pour décoder conjointement les deux descriptions. La matrice d'assignation d'index inverse est ensuite utilisée pour retrouver les coefficients quantifiés à partir des indices reçus et la reconstruction des coefficients est également effectuée à l'aide de l'information adjacente issue des trames clé. Après transformation inverse des coefficients, les trames Wyner-Ziv sont finalement combinées avec les trames clé précédemment décodées pour former la séquence complète d'images.

Aux décodeurs latéraux, représentés sur les figures 6.5 et 6.6, l'architecture de décodage est très proche de celle d'un décodeur vidéo distribué classique. La description des trames clé, après décodage, est utilisée pour générer une information adjacente servant à aider au

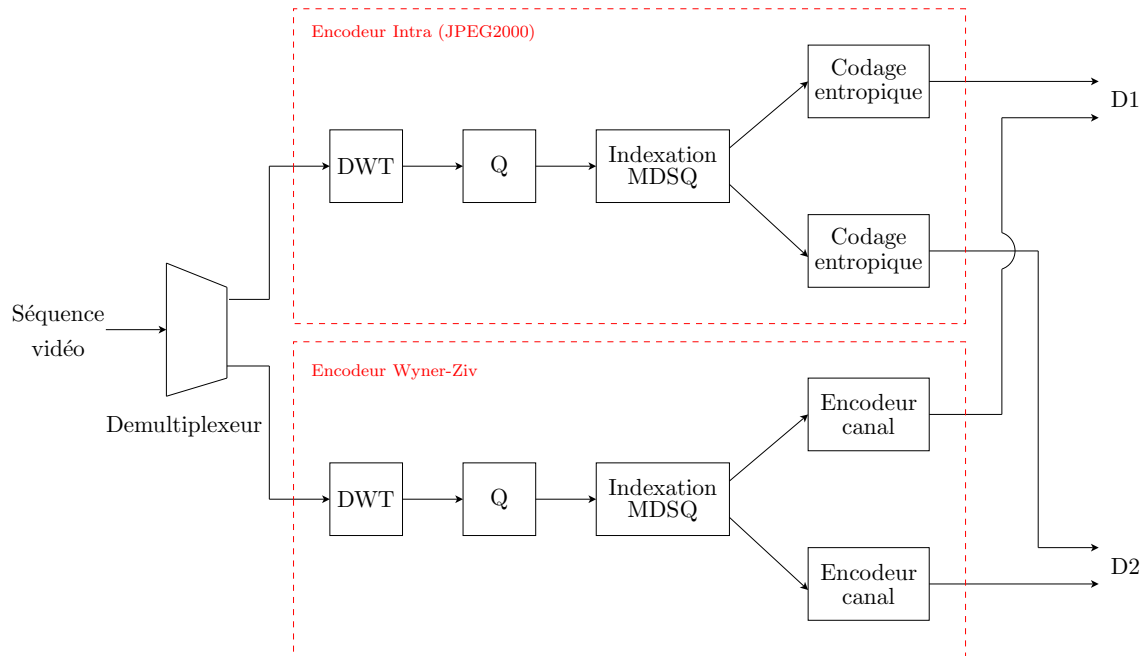


FIGURE 6.3 – Encodeur vidéo distribué par descriptions multiples.

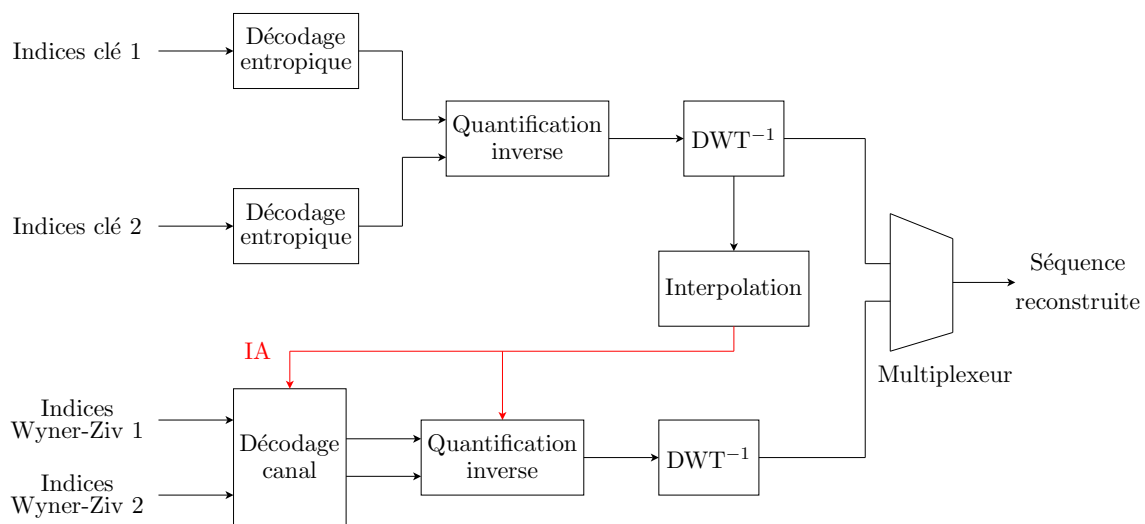


FIGURE 6.4 – Décodeur vidéo distribué central.

décodage de la description des trames Wyner-Ziv. La différence avec un décodeur vidéo distribué classique réside dans le fait que les symboles transmis sont des indices déduits de la matrice d'assignation d'index et non des coefficients de quantification.

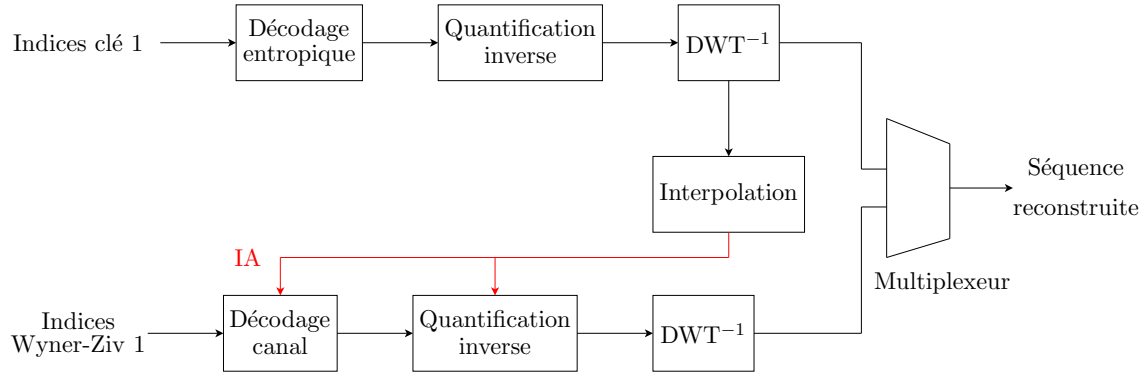


FIGURE 6.5 – Décodeur vidéo distribué latéral de la description 1.

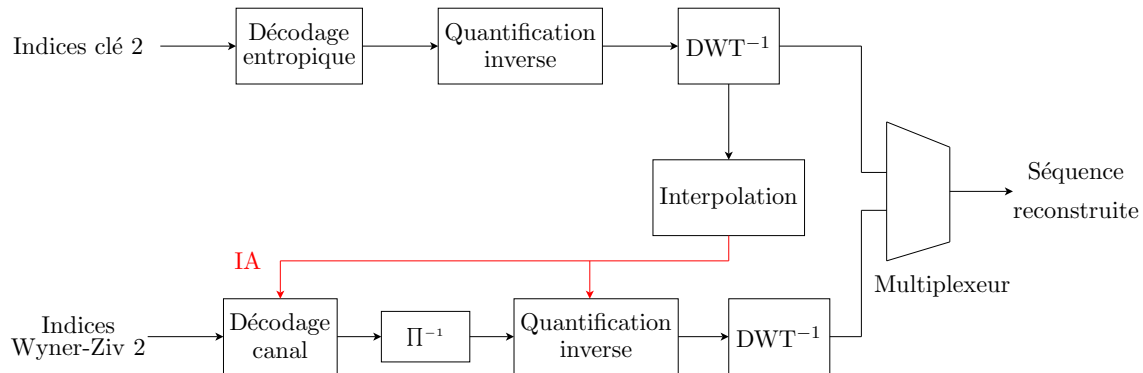


FIGURE 6.6 – Décodeur vidéo distribué latéral de la description 2.

Si le taux de redondance introduite par l'assignation d'index est choisi pour être le même pour les trames clé et les trames Wyner-Ziv, la réception d'une seule description devrait conduire à une dégradation répartie uniformément sur la séquence, sans introduire de *flickering* aux récepteurs central et latéraux.

### 6.2.3 Reconstruction optimale

On considère que le résidu entre les sous-bandes de coefficients d'ondelettes de la trame d'origine  $X$  et celles de l'information adjacente  $Y$  a une distribution laplacienne. Cela implique que  $X = Y + U$ , où le bruit  $U$  suit une distribution laplacienne ayant comme p.d.f. :

$$p_U(n) = \frac{\alpha}{2} e^{-\alpha|n|} \quad (6.3)$$

Comme nous cherchons à minimiser l'erreur quadratique moyenne, l'estimation optimale  $\hat{x}_{opt}$  est donnée par

$$\begin{aligned} \hat{x}_{opt} = E[x|x \in \bigcup_{k=1}^K [z_i^k, z_{i+1}^k], y] &= \frac{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} x f_{X|Y}(x) dx}{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} f_{X|Y}(x) dx} \\ &= \frac{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} x p_U(x-y) dx}{\sum_{k=1}^K \int_{z_i^k}^{z_{i+1}^k} p_U(x-y) dx} \end{aligned} \quad (6.4)$$

où  $z_0 < z_1 < \dots < z_N$  représentent les niveaux de quantification,  $N$  étant le nombre de niveaux de quantification. Le nombre  $K$  d'intervalles de quantification pour un  $x$  donné dépend du nombre de descriptions reçues et du nombre de diagonales dans la matrice d'assignation d'index. Au décodeur central,  $K = 1$ . Aux décodeurs latéraux,  $K$  est le nombre de coefficients quantifiés dans la ligne ou la colonne de la matrice d'assignation d'index qui correspond à l'indice reçu. Connaissant l'expression de la p.d.f. du bruit de corrélation entre  $x$  et  $y$ , on obtient finalement :

$$\hat{x}_{opt} = \frac{\sum_{k=1}^K g(k)}{\sum_{k=1}^K h(k)} \quad (6.5)$$

où

$$g(k) = \begin{cases} \left( \frac{1}{\alpha} + z_i^k \right) e^{\alpha(y-z_i^k)} - \left( \frac{1}{\alpha} + z_{i+1}^k \right) e^{\alpha(y-z_{i+1}^k)} & \text{si } y < z_i^k, \\ \left( \frac{1}{\alpha} - z_i^k \right) e^{-\alpha(y-z_i^k)} - \left( \frac{1}{\alpha} + z_{i+1}^k \right) e^{-\alpha(z_{i+1}^k-y)} + 2y & \text{si } y \in [z_i^k, z_{i+1}^k), \\ \left( \frac{1}{\alpha} - z_i^k \right) e^{\alpha(z_i^k-y)} - \left( \frac{1}{\alpha} - z_{i+1}^k \right) e^{\alpha(z_{i+1}^k-y)} & \text{si } y \geq z_{i+1}^k \end{cases} \quad (6.6)$$

et

$$h(k) = \begin{cases} e^{\alpha(y-z_i^k)} - e^{\alpha(y-z_{i+1}^k)} & \text{si } y < z_i^k, \\ 2 - e^{-\alpha(y-z_i^k)} - e^{-\alpha(z_{i+1}^k-y)} & \text{si } y \in [z_i^k, z_{i+1}^k), \\ e^{\alpha(z_{i+1}^k-y)} - e^{\alpha(z_i^k-y)} & \text{si } y \geq z_{i+1}^k. \end{cases} \quad (6.7)$$

### 6.3 Résultats expérimentaux

Pour les simulations, nous avons utilisé la séquence Foreman (au format QCIF à 15 Hz). La taille du GOP utilisé est de 2 trames (les trames paires sont codées en tant que trames clé et les trames impaires en tant que trames Wyner-Ziv). Les débits utilisés pour les trames clé sont 114, 266, 418 et 532 kbs. Le PSNR aux décodeurs latéraux est calculé comme la moyenne des PSNR des deux décodeurs latéraux et les débits correspondants sont la somme des débits des deux descriptions. Le nombre de plans de bits par sous-bandes

pour les indices codés en Wyner-Ziv ont été choisis manuellement de façon à garantir une différence en qualité entre la séquence décodée de trames clé et celle des trames codées en Wyner-Ziv la plus faible possible (la différence est toujours inférieure à 1 dB au décodeur central et inférieure à 2 dB aux décodeur latéraux).

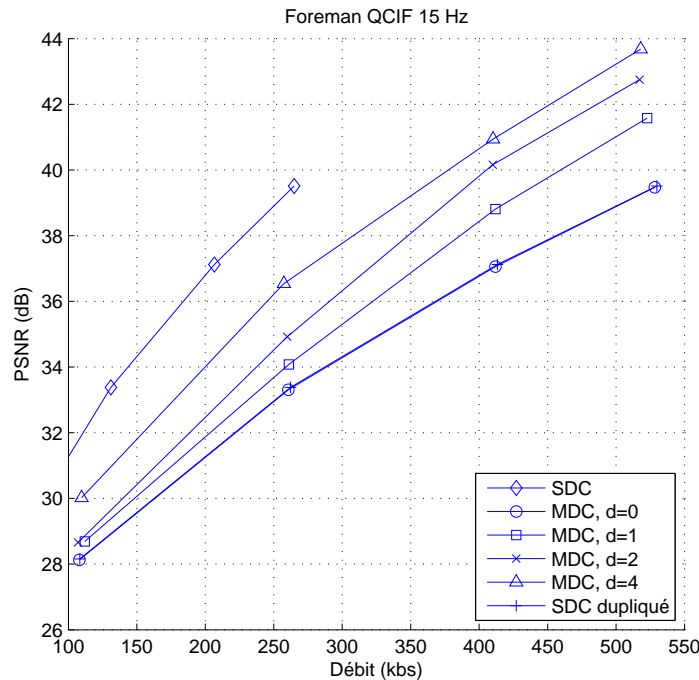


FIGURE 6.7 – Courbes débit-distorsion pour les trames clé des schémas SDC et MDC au décodeur central.

Les résultats obtenus pour les trames clé sont illustrés sur les figures 6.7 et 6.8 pour les décodeurs central et latéraux. Ils ont été obtenus en utilisant JPEG2000 (VM 5.2) dans un système de codage par deux descriptions progressives [152]. Ils montrent une augmentation progressive des performances débit-distorsion au décodeur central et une diminution aux décodeurs latéraux lorsque le nombre de diagonales dans la matrice d'assignation d'index augmente (le nombre de diagonales est  $2d + 1$ ), c'est-à-dire lorsque la redondance diminue.

La figure 6.9 montre les performances débit-distorsion obtenus pour les trames Wyner-Ziv au décodeur central des schémas sans décodage croisé utilisant des turbo codes comme code canal. Comme prévu, le schéma MDC avec  $d = 0$  est aussi performant que le schéma SDC lorsque le débit de ce dernier est compté deux fois. Les performances des schémas MDC avec  $d = 1, 2, 4$  s'approchent des performances atteintes par le schéma SDC. Cependant, on n'observe aucune différence significative entre les trois schémas MDC, qui obtiennent des performances quasiment identiques. Cela est dû au fait que, lorsque l'on réduit la redondance, on atteint un point ( $d = 1$ ) où plus de redondance n'augmenterait pas les performances centrales, car la corrélation entre l'information adjacente et les trames Wyner-Ziv serait trop faible, ce qui se traduirait par une augmentation du débit. On note que la réduction de débit obtenue en utilisant un schéma MDC avec  $d = 1$  par rapport à  $d = 0$  pour les trames Wyner-Ziv est plus importante que celle obtenue pour les trames clé et les mêmes valeurs de  $d$ . Les résultats aux décodeurs latéraux sont illustrés sur la figure 6.10. On

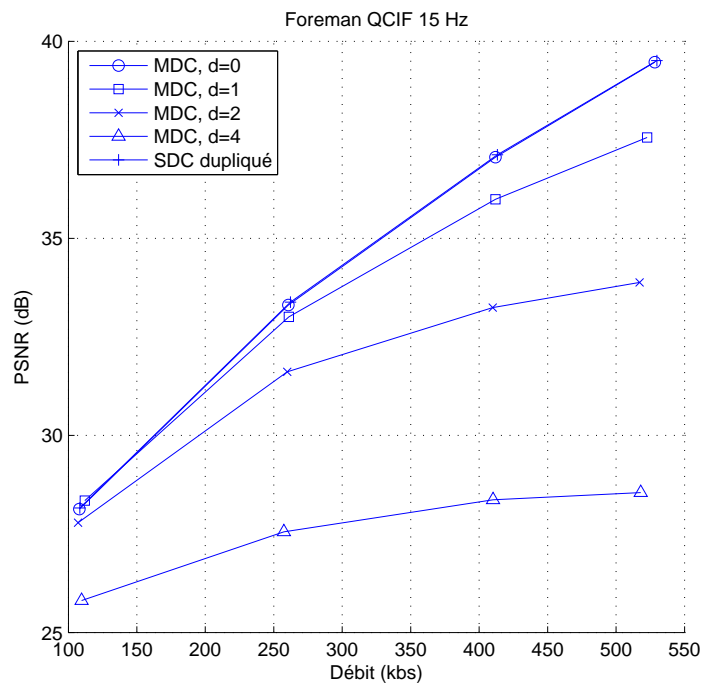


FIGURE 6.8 – Courbes débit-distorsion pour les trames clé des schémas SDC et MDC aux décodeurs latéraux.

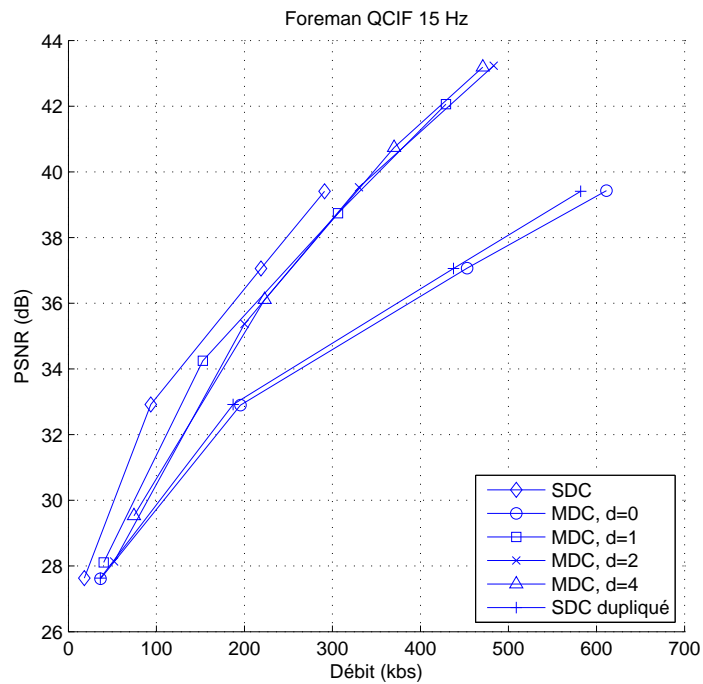


FIGURE 6.9 – Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés turbo au décodeur central.



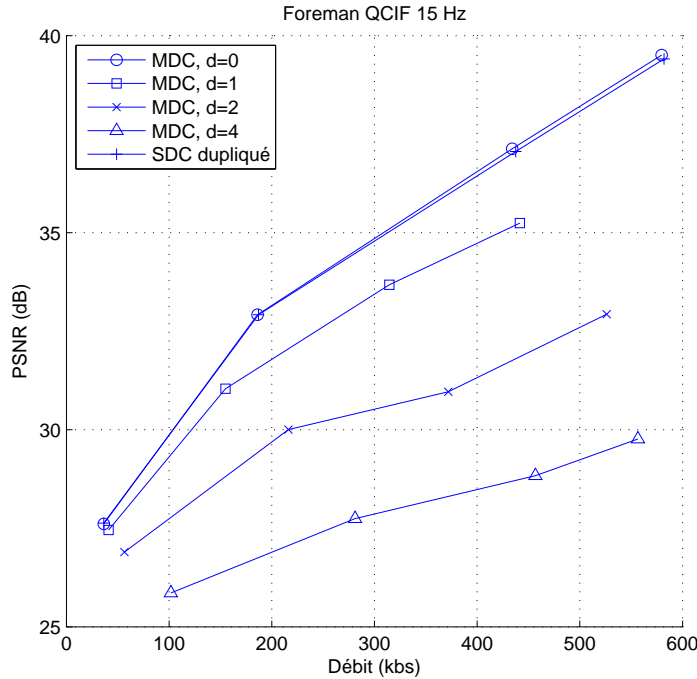


FIGURE 6.10 – Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés turbo aux décodeurs latéraux.

peut voir, en comparant les figures 6.8 et 6.10, que pour un débit donné, le PSNR obtenu pour les trames codées en Wyner-Ziv est très proche de celui obtenu aux décodeurs latéraux pour les trames clé. Malheureusement, la condition d'un PSNR stable entre les trames clé et les trames Wyner-Ziv n'est pas toujours satisfaite. La qualité visuelle reste cependant acceptable visuellement (la différence de PSNR ne dépassant pas 2 dB). Comme on peut le voir pour  $d = 1$ , en introduisant une légère diminution de la robustesse du schéma, les performances débit-distorsion au décodeur central se rapprochent très fortement de celles obtenues sans utiliser de MDC et restent relativement bonnes aux décodeurs latéraux par rapport au cas où les trames sont simplement dupliquées dans les deux descriptions.

Les figures 6.11 et 6.12 montrent les résultats débit-distorsion pour les trames Wyner-Ziv des schémas utilisant des codes LDPC comme code canal. Ils sont légèrement meilleurs que ceux obtenus avec des turbo codes.

Les figures 6.13 et 6.14 montrent les performances débit-distorsion pour toutes les trames aux décodeurs central et latéraux respectivement avec les schémas basés turbo. De la même manière que pour les trames clé, les performances augmentent progressivement au décodeur central et diminuent aux décodeurs latéraux lorsque  $d$  augmente. Sachant que les performances centrales pour les trames codées en Wyner-Ziv ne s'améliorent pas pour des valeurs de  $d$  supérieures à 1, on pourrait être tenté de choisir  $d = 1$  pour le codage Wyner-Ziv, même lorsque la valeur de  $d$  est plus élevée pour les trames clé. Néanmoins, nous devons garder à l'esprit que les trames Wyner-Ziv nécessitent une information adjacente obtenue par interpolation des trames clé pour être décodées; donc si l'on choisit une valeur de  $d$  supérieure pour les trames clé, aux décodeurs latéraux, l'information adjacente pourrait ne pas être de qualité suffisante pour décoder les trames Wyner-Ziv. Utiliser la même

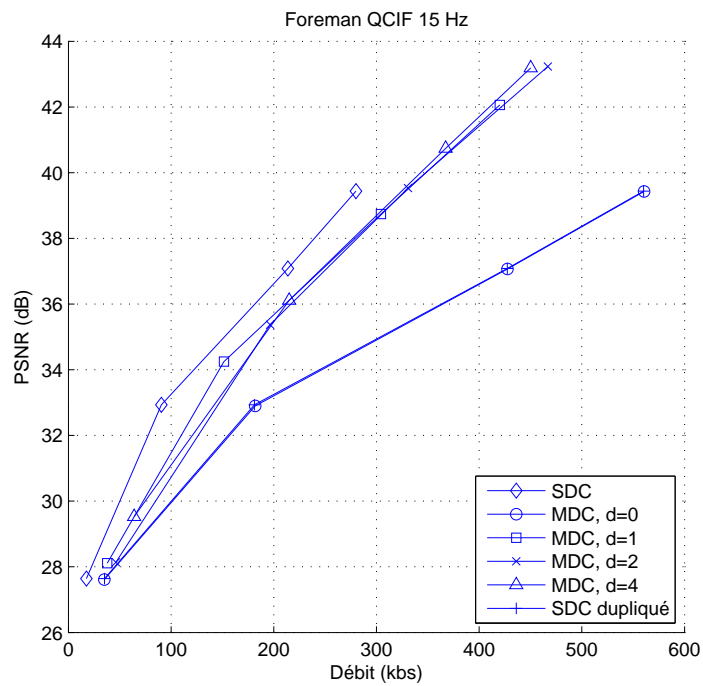


FIGURE 6.11 – Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés LDPC au décodeur central.

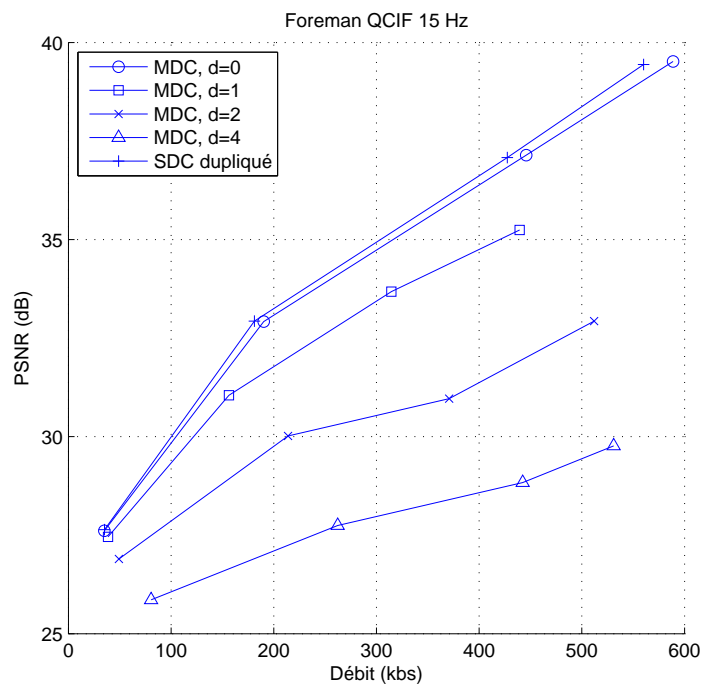


FIGURE 6.12 – Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés LDPC aux décodeurs latéraux.

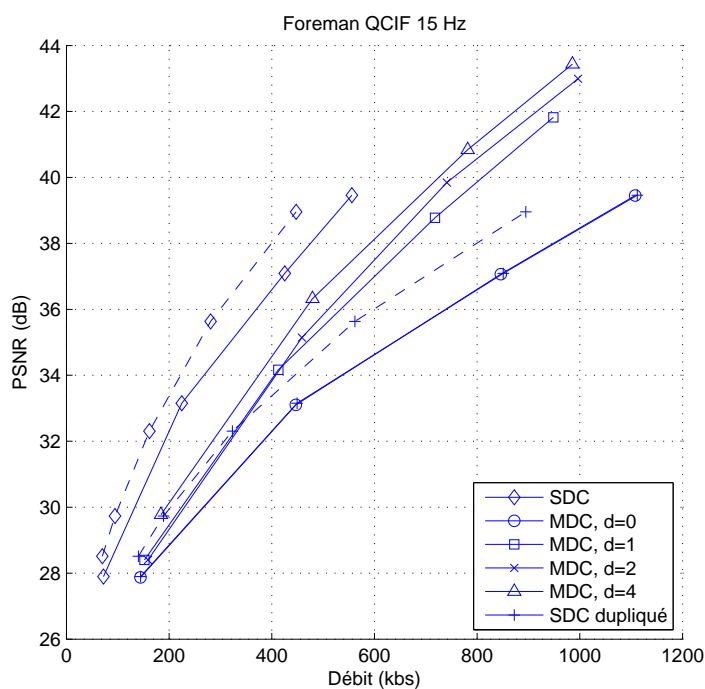


FIGURE 6.13 – Courbes débit-distorsion pour toutes les trames des schémas SDC, *Discover* et MDC basés turbo au décodeur central.

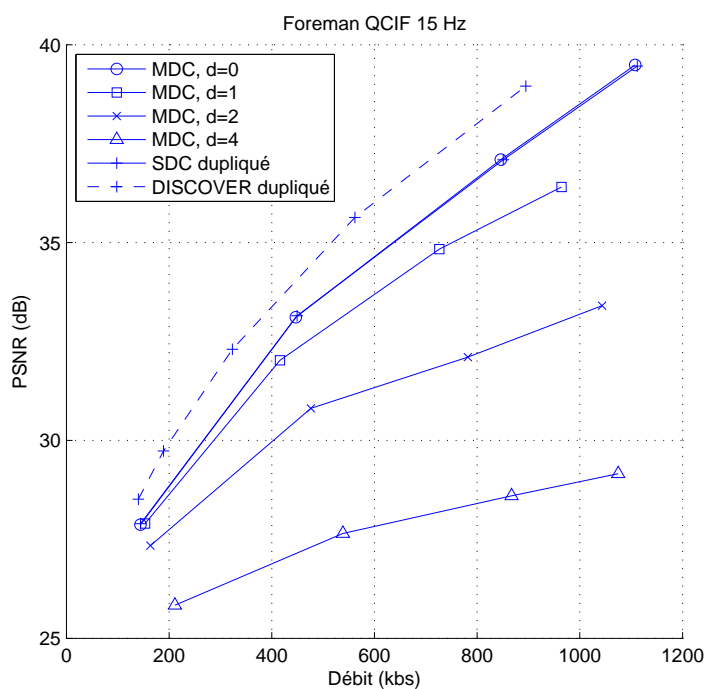


FIGURE 6.14 – Courbes débit-distorsion pour toutes les trames des schémas SDC, *Discover* et MDC basés turbo aux décodeur latéraux.

matrice d'assignation d'index permet également d'obtenir un PSNR à peu près stable entre les trames au cours du temps, quelque soit le nombre de descriptions reçues. Les courbes *Discover* ont été obtenues en utilisant le schéma de codage vidéo distribué du projet DISCOVER [107]. Ces bonnes performances proviennent principalement de l'utilisation de H.264/AVC en mode Intra pour coder les trames clé, qui est connu pour être plus efficace pour le codage de trames QCIF que JPEG2000 [154].

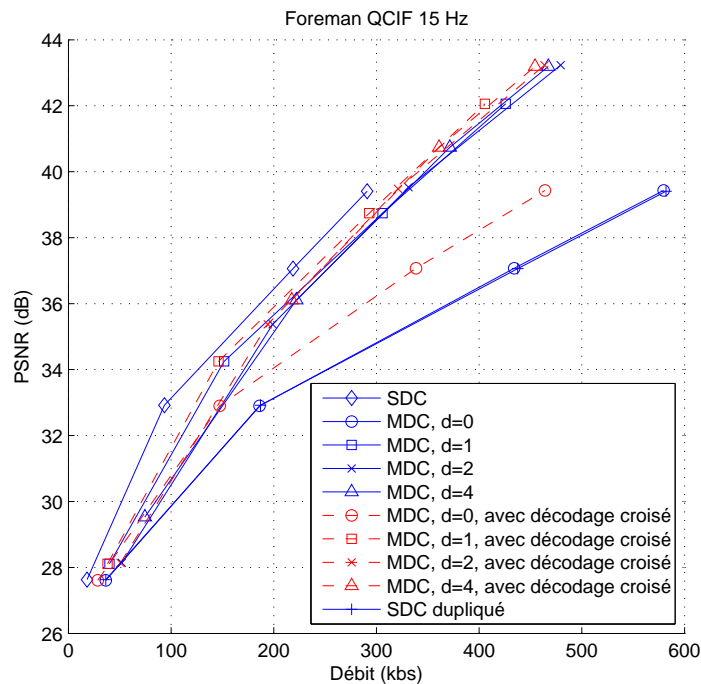


FIGURE 6.15 – Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés turbo au décodeur central avec et sans décodage croisé.

Sur les figures 6.15 et 6.16, nous comparons les performances débit-distorsion des schémas turbo et LDPC avec et sans décodage croisé. Le décodage croisé permet de gagner jusqu'à 2 dB pour  $d = 0$  mais n'améliorent pas les performances pour des valeurs plus élevées de  $d$ . Pour expliquer ce comportement, nous avons tracé sur les figures 6.17 et 6.18 la réduction de débit obtenue grâce au décodage croisé et le CSNR au cours du temps pour le schéma turbo avec  $d = 0$  et  $d = 1$  respectivement. Le débit des trames clé est fixé dans les deux cas à 523 kbs. Ces figures montrent que la réduction de débit est plus importante pour les trames qui correspondent à de faibles valeurs de CSNR, c'est-à-dire les trames pour lesquelles la qualité de l'information adjacente est la plus faible comme nous l'avions déjà constaté dans le chapitre précédent. En effet, l'information de décodage croisé peut être vue comme une deuxième information adjacente venant de l'autre description, et si une information adjacente n'apporte pas plus d'information que l'autre, alors elle ne peut contribuer à réduire le débit et reste sans effet. C'est ce qui se passe pour  $d = 1$  aux plus fortes valeurs de CSNR, où l'information de décodage croisé ne permet pas une réduction de débit. Une autre raison pour expliquer ces mauvaises performances pour des valeurs de  $d$  supérieures à 0 est le manque de précision dans l'estimation du modèle de distribution des coefficients dans les trames. En effet, le décodeur ne connaissant pas la trame d'ori-

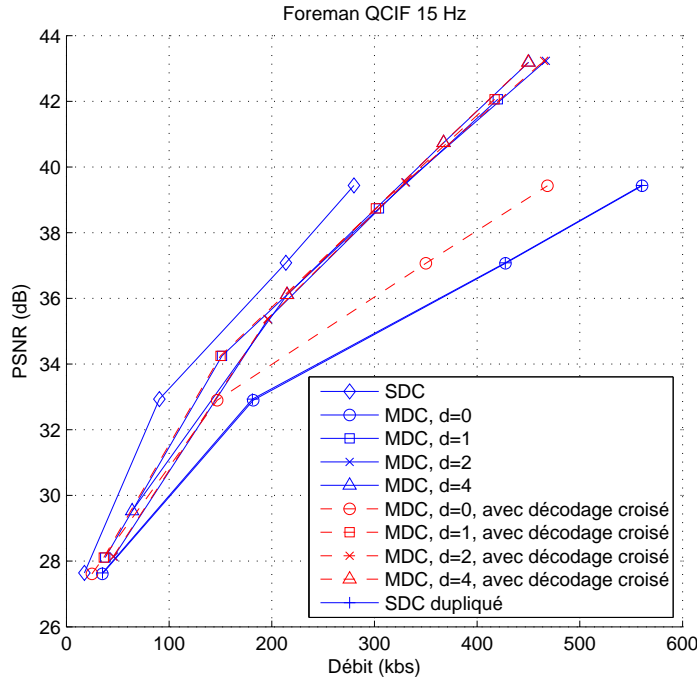


FIGURE 6.16 – Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés LDPC au décodeur central avec et sans décodage croisé.

gine est obligé d'effectuer son estimation du modèle de distribution des sous-bandes de la trame d'origine avec les coefficients de l'information adjacente au lieu de ceux de la trame d'origine.

## 6.4 Conclusion

Dans ce chapitre, nous avons présenté un schéma de codage vidéo robuste par descriptions multiples avec information adjacente, où la robustesse vient de la présence de deux descriptions pour représenter la vidéo d'origine. La méthode applique une MDSQ sur les trames clé et sur les trames codées en Wyner-Ziv pour générer deux descriptions qui contiennent chacune une information sur toutes les trames de la séquence. Il a été montré qu'en réduisant légèrement le taux de redondance entre les descriptions, les performances débit-distorsion au décodeur central pour les trames codées en Wyner-Ziv peuvent être très proches de celles obtenues avec un schéma de codage par description simple. Cependant, aucune réduction de débit n'est obtenue lorsque la corrélation entre les descriptions Wyner-Ziv et l'information adjacente générée à partir des trames clé devient trop faible. Notre codec vidéo possède la propriété intéressante que l'augmentation de débit due à l'introduction de redondance peut être atténuée lorsque la corrélation entre les descriptions est maximale, en effectuant un décodage croisé des deux descriptions au décodeur central. Malheureusement, lorsque la redondance est trop faible, la même méthode n'améliore pas les performances à cause de la plus faible corrélation entre les descriptions, de la redondance entre l'information provenant des descriptions Wyner-Ziv et l'information adjacente générée à partir des trames clé, et du manque de précision dans la modélisation de la

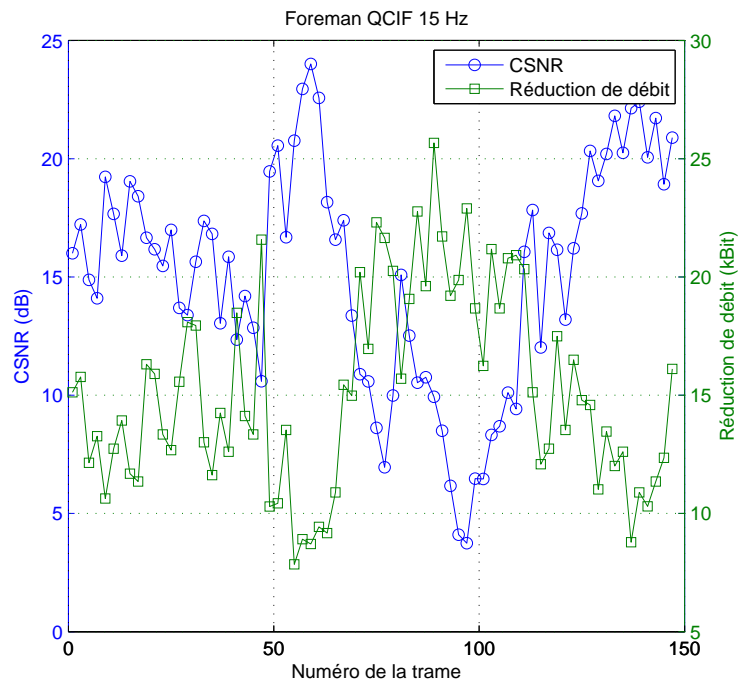


FIGURE 6.17 – Réduction de débit grâce au décodage croisé au décodeur central et CSNR pour le schéma MDC avec  $d = 0$  et un débit pour les trames clé fixé à 532 kbs.

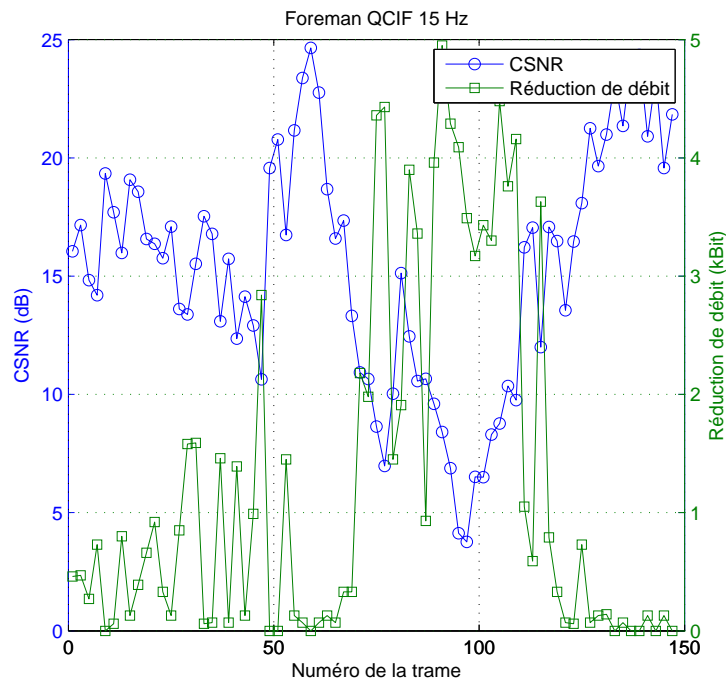


FIGURE 6.18 – Réduction de débit grâce au décodage croisé au décodeur central et CSNR pour le schéma MDC avec  $d = 1$  et un débit pour les trames clé fixé à 532 kbs.

distribution des coefficients dans les sous-bandes.

---

## Conclusion et perspectives

Au cours de cette thèse, nous nous sommes intéressés à des systèmes de codage robuste utilisant à la fois des principes de codage de source distribué et des principes de codage par descriptions multiples.

Dans un premier temps, nous avons proposé plusieurs schémas de codage vidéo par descriptions multiples utilisant un sous-échantillonnage pour générer deux descriptions suivies d'un filtrage temporel compensé en mouvement. Deux manières de sous-échantillonner la séquence ont été envisagées. Dans la première, les trames paires et impaires sont séparées pour former les deux descriptions. Dans la seconde, les trames sont d'abord séparées en GOP (dont la taille est donnée par le facteur de sous-échantillonnage) avant d'être placées dans des descriptions. Des résultats expérimentaux ont été obtenus avec ces deux structures de codage pour des filtres de Haar 2-bandes et 3-bandes et trois et deux niveaux de composition temporels respectivement. Ces résultats ont montré qu'un regroupement des trames consécutives avant d'appliquer le filtrage donne de meilleurs résultats au décodeur central mais de moins bons résultats aux décodeurs latéraux. L'écart de performance dépend de la quantité de mouvement dans la séquence testée : l'écart est plus grand lorsque la séquence possède beaucoup de mouvement. Dans ces schémas de codage MD, lorsqu'une description est perdue, les trames manquantes sont obtenues par interpolation des trames reçues. Les trames ainsi reconstruites ne sont pas de suffisamment bonne qualité et on assiste alors à des variations brutales de qualité à travers la séquence. C'est pourquoi, dans une deuxième partie, nous avons considéré le codage de ces trames manquantes par un système de codage de Wyner-Ziv. Les trames reçues interpolées sont alors utilisées comme information adjacente pour le décodage Wyner-Ziv. Les résultats ont montré que les schémas qui utilisent un codage de Wyner-Ziv ont de meilleures performances moyennes que le schéma MDC 3-bandes aux décodeurs latéraux pour la plupart des débits considérés et que les variations en qualité entre les trames étaient réduites, conduisant à moins d'artefacts. Cependant, les performances débit-distorsion au décodeur central sont systématiquement pires que celles du schéma MDC 3-bandes, bien que les mêmes schémas sans information supplémentaire soient plus efficaces. Ceci est simplement dû au fait que, l'information Wyner-Ziv, systématiquement envoyée comme dans un mécanisme FEC, est ignorée au décodeur central, c'est-à-dire qu'elle ne contribue qu'à améliorer les performances aux décodeurs latéraux et constitue une redondance pure au central. C'est pourquoi, afin de réduire le débit utilisé pour le flux codé en Wyner-Ziv, tout en garantissant une amélioration de la qualité aux décodeurs latéraux, nous avons proposé un dernier schéma, qui ne code en Wyner-Ziv que les sous-bandes temporelles de basses fréquences et se débarrasse des hautes fréquences. Les résultats montrent que ce schéma peut s'avérer meilleur que le schéma MDC 3-bandes pour les débits les plus élevés et les indices de quantification les plus faibles. De leur côté, les décodeurs latéraux sont moins performants qu'avec les précédents schémas mais bénéficient tout de même de la présence de la sous-bande de basses fréquences des trames

---



manquantes en les substituant à celles des trames interpolées.

La deuxième contribution concerne le codage de source distribué robuste. Nous avons proposé un schéma de codage par descriptions multiples avec information adjacente basé sur la MDSQ. Ce schéma utilise soit des codes LDPC, soit des codes turbo pour coder les indices générés par une matrice d'assignation d'index pour la source  $X$ . L'information adjacente  $Y$  n'est pas codée et est directement disponible aux décodeurs. Si une seule description est reçue, le décodage des indices reçus s'effectue à l'aide de  $Y$  et les symboles sont reconstruits en fonction des indices parfaitement décodés et de  $Y$ . Si les deux descriptions sont reçues, le décodage Wyner-Ziv croisé permet à la fois de prendre en compte la corrélation entre  $X$  et  $Y$  et d'exploiter celle entre les indices. Les résultats de simulations sur des sources gaussiennes ont montré que ce système peut permettre un décodage conjoint source-canal efficace, notamment grâce au décodage croisé qui réduit efficacement le débit au décodeur central, et ainsi autorise une augmentation de la quantité de redondance, et donc de la robustesse, pour un coût raisonnable.

Nous avons ensuite introduit un système de codage vidéo distribué robuste. Son principe consiste à effectuer un codage par descriptions multiples sur les trames codées en Wyner-Ziv et sur les trames clé dans un codeur vidéo distribué de type Stanford. Deux descriptions prêtes à être transmises sur un canal sont ensuite créées en regroupant une description pour les trames clé avec une description pour les trames codées en Wyner-Ziv. Au décodeur, si une seule description est reçue, les trames clé reçues sont dégradées et, par voie de conséquence, la qualité des trames interpolées aussi. Cependant, les trames codées en Wyner-Ziv sont également dégradées et si leur qualité après reconstruction avec l'information adjacente a été choisie de manière à ce qu'elle corresponde à celle des trames clé, on obtient une qualité suffisamment stable tout au long de la séquence. Au décodeur central, les deux descriptions des trames codées en Wyner-Ziv sont décodées conjointement avec l'aide d'une même information adjacente en effectuant un décodage croisé. On obtient alors une meilleure qualité, aussi bien pour les trames clé que pour les trames codées en Wyner-Ziv. Les résultats de simulation ont montré qu'en réduisant la corrélation, on pouvait approcher les performances d'un codeur à description simple. Mais lorsque la redondance introduite par la MDSQ devient trop faible, la corrélation entre l'information adjacente et les trames codées en Wyner-Ziv devient également trop faible pour pouvoir observer une baisse de débit, les performances se mettent alors à stagner autour d'une même valeur débit-distorsion. Le décodage croisé est efficace au décodeur central lorsque la corrélation entre les deux descriptions est grande, mais devient de moins en moins nécessaire à mesure que la corrélation diminue.

Dans ces deux derniers chapitres, nous avons proposé une technique de décodage conjoint source-canal ; une autre voie à considérer serait la construction conjointe de codes canal et de quantificateurs à descriptions multiples. De plus, dans nos simulations nous avons toujours considéré une transmission sur des canaux sans pertes ou un canal peut être soit en état de marche, soit défaillant. Il serait préférable de considérer des transmissions dans des environnements bruités qui correspondent davantage à la réalité. Enfin, dans les simulations sur des sources gaussiennes, nous avons supposé que l'information adjacente est disponible au décodeur quelque soit la manière dont cette information a été obtenue. Avec cette hypothèse, il était évident que les performances de décodage allaient s'améliorer avec l'aide de l'information adjacente. Pour mieux correspondre à la réalité, il serait préférable de considérer le cas où l'information adjacente doit également être transmise au décodeur comme c'est le cas en vidéo.

Une des principales difficultés rencontrées au cours des tests de simulation a été le pa-

ramétrage des codeurs vidéo. En effet, dans un codeur vidéo distribué de type Stanford, la qualité des trames codées en Wyner-Ziv doit correspondre à celle des trames clé afin d'éviter des variations de qualité au cours du temps. Ainsi, il est nécessaire de déterminer, pour chaque séquence vidéo et pour plusieurs niveaux de qualité pour les trames clé, le nombre de niveaux de quantification de chaque bande de coefficients (DCT ou DWT) des trames codées en Wyner-Ziv. Dans le cas des codeurs MD du chapitre 6, ce paramétrage devient encore plus complexe : il dépend également de la matrice d'assignation d'index et de la qualité des trames clé à la sortie des trois décodeurs. Jusqu'à présent, il était seulement possible de réaliser ce paramétrage manuellement. Pour résoudre ce problème, nous avons tenté, dans le cadre du codeur vidéo basé ondelettes à description simple présenté dans le chapitre 6, d'utiliser l'algorithme PCRD (en anglais post-compression rate-distortion) de JPEG2000 [155] à la fois pour les trames clé et les trames codées en Wyner-Ziv. Malheureusement, les résultats de simulation ont montré que, trop souvent, l'algorithme privilégiait les sous-bandes de hautes fréquences là où on il aurait été préférable de privilégier les basses fréquences. On observait alors une augmentation du débit des trames codées en Wyner-Ziv. De plus, l'utilisation de l'information adjacente au moment de la reconstruction empêchait d'avoir une qualité stable, la qualité des trames codées en Wyner-Ziv devenant alors en général légèrement meilleure que celles des trames clé. La solution proposée n'était donc pas la bonne et ce problème reste à résoudre.

---



# Glossaire

**ARQ** : Automatic Repeat reQuest.  
**AVC** : Advanced Video Coding.  
**BCJR** : Bahl, Cocke, Jelinek et Raviv [92].  
**BER** : Bit Error Rate.  
**CIF** : Common Intermediate Format for images ( $352 \times 288$  pixels).  
**CRC** : Cyclic Redundancy Check.  
**CSNR** : Correlation-Signal to Noise Ratio.  
**DCT** : Discrete Cosine Transform.  
**DPCM** : Differential pulse code modulation.  
**DSC** : Distributed Source Coding.  
**DVC** : Distributed Video Coding.  
**DWT** : Discrete Wavelet Transform.  
**EDDD** : Expected Distorsion of DCT coefficients.  
**EQM** : Erreur Quadratique Moyenne.  
**FEC** : Forward Error Correction.  
**FMO** : Flexible Macroblock Ordering.  
**KF** : Key Frame.  
**IA** : Information Adjacente.  
**Inter** : Mode de codage avec utilisation de prédiction temporelle.  
**Intra** : Mode de codage sans utilisation de prédiction temporelle.  
**GOP** : Group Of Pictures.  
**LDPC** : Low Density Parity Check.  
**LLR** : Log-Likelihood Ratio.  
**MAP** : Maximum A Posteriori.  
**MCTF** : Motion Compensated Temporal Filter.  
**MD (codage)** : Codage par descriptions multiples.  
**MDC** : Multiple Description Coding.  
**MMSE** : Minimum of Mean Square Error.  
**MSB** : Most Significant Bit.  
**p.d.f.** : probability density function.  
**PSNR** : Peak Signal to Noise Ratio.  
**QCIF** : Quarter Common Intermediate Format for images ( $176 \times 144$  pixels).  
**QoS** : Quality of Service.  
**ROI** : Region Of Interest.  
**RSC** : Recursive Systematic Convolutional.  
**SISO** : Soft-Input-Soft-Output.  
**SD (codage)** : Codage par description simple.  
**SDC** : Single Description Coding.

---

**SNR** : Signal to Noise Ratio.

**SW** : Slepian-Wolf.

**UMTS** : Universal Mobile Telecommunications System.

**WZ** : Wyner-Ziv.

---

## Annexe A

# Les turbo codes

Les turbo codes ont été découverts en 1993 par Berrou *et al.* [153, 156]. Un turbo encodeur est constitué de deux codes élémentaires en parallèle séparés par un entrelaceur.

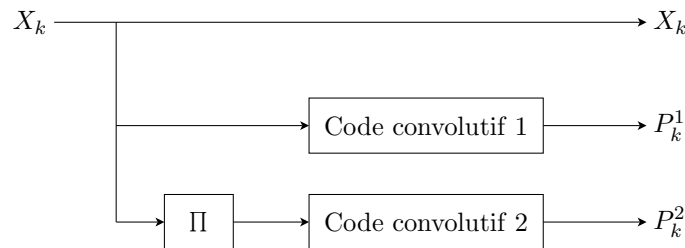


FIGURE A.1 – Schéma d'un turbo encodeur.

Prenons l'exemple du turbo code de taux  $\frac{1}{3}$  illustré sur la figure A.1. Les bits d'information de la source binaire  $X$  sont entrelacés avant d'être encodés par le deuxième code élémentaire. Le choix de l'entrelaceur joue un rôle fondamental, son rôle est de faire croire au décodeur que les deux encodeurs sont non corrélés. Dans cet exemple, les encodeurs sont des codes RSC qui pour un bit d'information  $X_k$  génère 1 bit de parité  $P_{1k}$  ou  $P_{2k}$ .

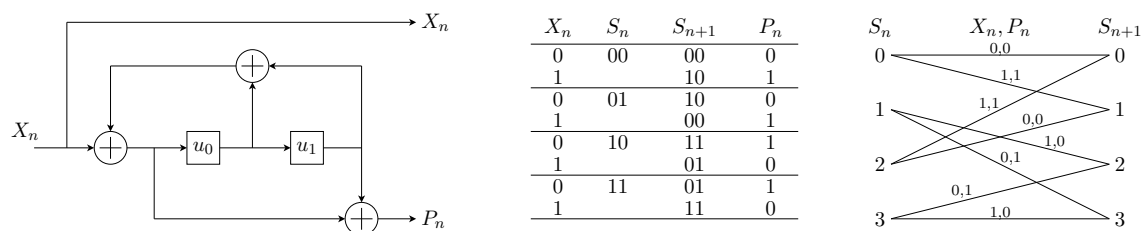


FIGURE A.2 – Schéma d'un code convolutif récursif systématique à 4 états donnés par la séquence de bits  $u_1u_0$  (gauche), sa table d'encodage (centre) et son treillis de transition (droite).

La figure A.2 (gauche) montre un exemple de code RSC à 4 états. L'état du code à un instant  $n$  est donné par  $S_n = \{u_0u_1\}$ . L'opération d'encodage de ce code RSC est résumée par sa table d'encodage au centre de la figure A.2 et par son treillis de transition à droite. On y voit les paires de bits en sortie ( $X_n, P_n$ ) et l'état du code après transition d'un état  $S_n$  en un état  $S_{n+1}$ .

Après turbo encodage, les bits de parité  $P_{1k}$  et  $P_{2k}$  et les bits d'information  $X_k$  sont convertis en valeurs symbole (modulation BPSK) avant d'être transmis sur un canal bruité au décodeur. Les valeurs symbole sont respectivement notées  $p_{1k}$ ,  $p_{2k}$  et  $x_k$ .

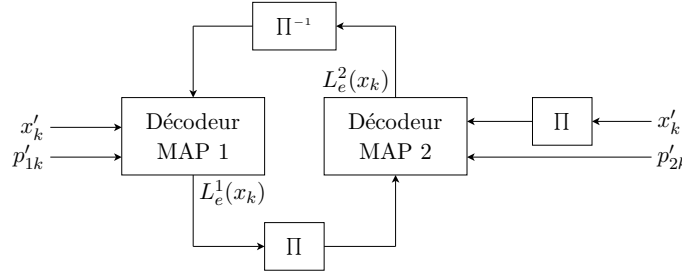


FIGURE A.3 – Schéma d'un turbo décodeur.

La structure du décodeur est illustrée sur la figure A.3. Il se compose d'une paire de décodeurs qui travaillent de façon coopérative pour raffiner l'estimation des bits d'information d'origine. Les décodeurs utilisent un algorithme MAP (maximum a posteriori) appelé BCJR [92] qui produit une information extrinsèque à partir des bits de parité. A la première itération, le premier décodeur n'a aucune information d'initialisation (l'information *a priori* est nulle). Dans les itérations suivantes, l'information extrinsèque d'un décodeur sert à initialiser l'autre décodeur. L'information circule de cette manière jusqu'à ce que la décision *soft* converge vers une valeur stable. Cette décision est ensuite convertie en une décision *hard* pour décoder la séquence d'origine.

L'algorithme MAP minimise la probabilité d'avoir un bit d'erreur en utilisant la séquence entière pour identifier le bit le plus probable à chaque stage du treillis. Il ne contraint pas l'ensemble des bits estimés à correspondre à un chemin valide dans le treillis. Les résultats peuvent donc différer de ceux générés par un décodeur de Viterbi [157] qui se contente d'identifier le chemin valide le plus probable dans un treillis.

On utilise les notations suivantes pour représenter les paires de symboles transmis :

$$\mathbf{y}_k = \{x_k, p_k\} \text{ et } \mathbf{y} = y_{1,K} = \{y_1, y_2, \dots, y_K\} \quad (\text{A.1})$$

et reçus :

$$\mathbf{y}'_k = \{x'_k, p'_k\} \text{ et } \mathbf{y}' = y'_{1,K} = \{y'_1, y'_2, \dots, y'_k\} \quad (\text{A.2})$$

Les décision *soft* à la sortie des décodeurs MAP est définie par le LLR suivant :

$$L_{\text{map}}(x_k) = \log \frac{\Pr(x_k = +1 | \mathbf{y}')}{\Pr(x_k = -1 | \mathbf{y}')} \quad (\text{A.3})$$

Pour un code convolutif, on peut exprimer les probabilités par les valeurs des états sur le treillis :

$$\frac{\Pr(x_k = +1 | \mathbf{y}')}{\Pr(x_k = -1 | \mathbf{y}')} = \frac{\sum_{(s',s) \in S^+} \Pr(s_{k-1} = s', s_k = s | \mathbf{y}')}{\sum_{(s',s) \in S^-} \Pr(s_{k-1} = s', s_k = s | \mathbf{y}')} = \frac{\sum_{(s',s) \in S^+} \Pr(s_{k-1} = s', s_k = s, \mathbf{y}')}{\sum_{(s',s) \in S^-} \Pr(s_{k-1} = s', s_k = s, \mathbf{y}')} \quad (\text{A.4})$$

Le numérateur est une somme sur toutes les transitions d'états possibles associées à un bit d'information "1", et le dénominateur est une somme sur toutes les transitions d'états possibles associées à un bit d'information "0".

La probabilité de passer d'un état particulier à un autre état particulier et d'observer une séquence d'informations bruitées particulière associée au treillis de transition peut s'exprimer, grâce au théorème de Bayes, par :

$$\begin{aligned} Pr(S_{k-1} = s', s_k = s, y'_{1,K}) &= Pr(s_{k-1} = s', y'_{1,k-1}) \cdot Pr(s_k = s, y'_k | s_{k-1} = s') \cdot Pr(y'_{k+1,K} | s_k = s) \quad (\text{A.5}) \\ &= \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s) \quad (\text{A.6}) \end{aligned}$$

Le terme  $\alpha_k(s')$  est la probabilité d'arriver à une branche dans un état particulier et d'avoir observé la séquence de données bruitées  $y'_{1,k} = y'_1, y'_2, \dots, y'_k$  qui conduit à cet état. En effectuant la somme sur tous les chemins menant à cet état, on peut calculer  $\alpha_k(s')$  par récursion en fonction des valeurs de  $\gamma_k(s', s)$  :

$$\alpha_k(s) = Pr(s_k = s, y'_{1,k}) = \sum_{s'} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \quad (\text{A.7})$$

Le terme  $\beta_k(s)$  est la probabilité de sortir d'une branche via un état particulier  $s$  et d'observer la séquence d'observations bruitées  $y'_{k+1,K} = y'_{k+1}, y'_{k+2}, \dots, y'_K$ . En effectuant la somme sur tous les chemins à la sortie de cet état, on peut calculer  $\beta_k(s)$  par récursion en fonction des valeurs de  $\gamma_k(s', s)$  :

$$\beta_k(s') = Pr(y'_{k+1,K} | s_k = s) = \sum_s \gamma_{k+1}(s', s) \cdot \beta_{k+1}(s) \quad (\text{A.8})$$

Pour une transition d'état donnée, le signal transmis est  $y_k$  et le prochain état est uniquement déterminé par la valeur du bit d'information. Le théorème de Bayes permet alors d'écrire :

$$\begin{aligned} \gamma_k(s', s) &= Pr(s_k = s, y'_k | s_{k-1} = s') = Pr(y'_k | s_{k-1} = s', s_k = s) \cdot Pr(s_k = s | s_{k-1} = s') \quad (\text{A.9}) \\ &= Pr(y'_k | y_k) \cdot Pr(x_k) \quad (\text{A.10}) \end{aligned}$$

La probabilité que le bit d'information prenne une valeur particulière peut s'exprimer en terme de LLR sur sa probabilité *a priori* ( $L_a(x_k) = \log \left[ \frac{Pr(x_k=+1)}{Pr(x_k=-1)} \right]$ ). La probabilité d'observer une certaine paire de données bruitées dépend du modèle de distribution du bruit sur le canal. Pour un bruit gaussien d'espérance nulle et de variance  $\sigma^2$ , la probabilité de transition peut s'exprimer ainsi :

$$\gamma_k(s', s) = A_k \cdot \exp \left[ \frac{1}{2} (x_k L_a(x_k) + x_k L_c x'_k + p_k L_c p'_k) \right], \quad L_c = \frac{2}{\sigma^2} \quad (\text{A.11})$$

où  $A_k$  est un facteur qui ne dépend pas de  $x_k$ . Revenons à l'équation du LLR du MAP (A.3) et remplaçons les expressions au numérateur et au dénominateur par celles que l'on vient de définir. Le LLR *a posteriori* peut alors s'écrire :

$$L_{\text{map}}(x_k) = L_a(x_k) + L_c x'_k + \log \left[ \frac{\sum_{(s',s) \in S^+} \alpha_{k-1}(s') \exp \left( \frac{p_k L_c p'_k}{2} \right) \beta_k(s)}{\sum_{(s',s) \in S^-} \alpha_{k-1}(s') \exp \left( \frac{p_k L_c p'_k}{2} \right) \beta_k(s)} \right] \quad (\text{A.12})$$

Le premier terme,  $L_a(x_k)$ , est l'information *a priori*. C'est l'estimation initiale de la valeur de  $x_k$  avant de lancer l'algorithme MAP. Le second terme,  $L_c x'_k$ , est l'information provenant de l'observation bruitée qui ne dépend pas des contraintes du code convolutif. Le troisième



terme,  $L_e(x_k)$ , est l'information que l'on a appris de la contrainte de parité. C'est cette information qu'on appelle information extrinsèque.

Dans un turbo décodeur, l'information extrinsèque d'un décodeur MAP est utilisée comme information *a priori* par l'autre décodeur MAP. Ce passage d'information (responsable de l'appellation «turbo») se poursuit jusqu'à ce que l'information *a posteriori* converge vers une valeur stable ou que le nombre d'itérations atteigne un certain seuil. La décision *hard* est alors prise selon la valeur de la décision *soft* en sortie du dernier décodeur MAP.

---

## Annexe B

# Les codes LDPC

Les codes Low Density Parity Check (LDPC), littéralement «contrôle de parité à faible densité», ont été découverts par Gallager [158] en 1962, puis réinventés récemment par MacKay et Neal [159]. Un code LDPC est un code en blocs  $(n, k)$  défini par une matrice  $\mathbf{H}$  de taille  $(n - k) \times n$  parcimonieuse (elle contient très peu de "1"). Si le nombre de "1" dans chaque ligne et chaque colonne est le même, le code LDPC est régulier, sinon il est irrégulier. Un exemple de matrice d'un code  $(8, 4)$  régulier est donnée par l'équation (B.1), son graphe biparti [160] associé est représenté sur la figure B.1. Dans la figure, les nœuds à gauche sont les nœuds de variable ( $c_i$ ), et ceux à droite, les nœuds de contrôle ( $f_i$ ). Chaque nœud de variable (respectivement contrôle) représente une ligne (resp. colonne) dans la matrice. Pour un code LDPC  $(n, k)$ , le graphe contient  $n$  nœuds de variable et  $n - k$  nœuds de contrôle. Le degré d'un nœud de variable (resp. contrôle) correspond au nombre de nœuds de contrôle (resp. variable) auxquels il est relié, il est donné par le nombre de 1 dans la ligne (resp. colonne) associée dans la représentation matricielle.

$$\mathbf{H} = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}. \quad (\text{B.1})$$

Le transmetteur envoie un mot de code  $\mathbf{c}$  tel que  $\mathbf{H}^T \mathbf{c} = \mathbf{0}$ . Le récepteur reçoit le vecteur  $\mathbf{y}$ . Le décodage d'un code LDPC est un problème NP-complet. Cependant, il est possible d'obtenir une bonne approximation avec des techniques de décodage sous-optimales basées sur la propagation de croyance. L'algorithme de propagation de croyance est équivalent à l'algorithme somme-produit appliqué au graphe factoriel correspondant à un réseau bayésien donné [161]. Si  $\mathbf{H}$  ne comporte pas de cycle, l'algorithme de propagation de croyance converge vers la solution exacte [162]. Le décodage sur un graphe biparti est illustré sur la figure B.2. Commençons par donner quelques notations :

- $P_i = Pr(c_i = 1 | y_i)$ .
- $q_{ij}$  est un message envoyé par le nœud de variable  $c_i$  vers le nœud de contrôle  $f_j$ . Chaque message contient la paire  $q_{ij}(0)$  et  $q_{ij}(1)$  qui représentent le niveau de croyance que  $y_i$  est un "0" ou un "1".
- $r_{ij}$  est un message envoyé par le nœud de contrôle  $f_j$  vers le nœud de variable  $c_i$ . Chaque message contient la paire  $r_{ji}(0)$  et  $r_{ji}(1)$  qui indiquent le niveau de croyance que  $y_i$  est un "0" ou un "1".

L'algorithme de propagation de croyance est le suivant :

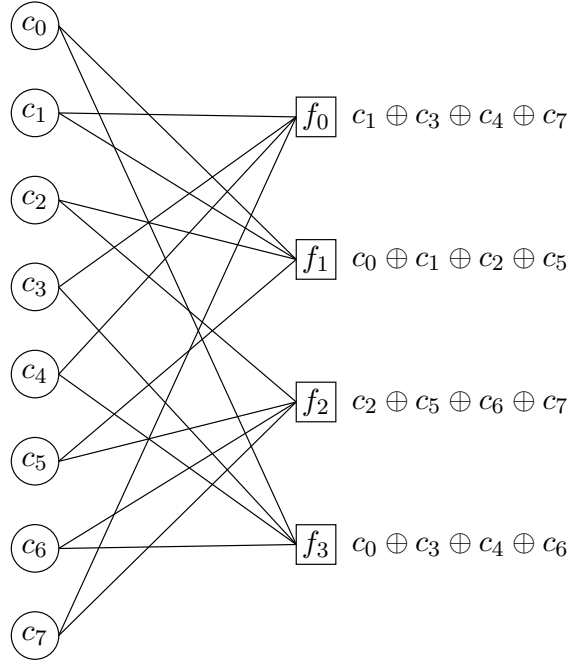


FIGURE B.1 – Graphe biparti de la matrice de parité donnée par l'équation B.1.

1. Tous les nœuds de variable envoient leurs messages  $q_{ij}$ . Comme aucune autre information n'est disponible à cette étape,  $q_{ij}(1) = P_i$  et  $q_{ij}(0) = 1 - P_i$ .
2. Les nœuds de variable calculent leurs messages de réponse  $r_{ji}$  :

$$r_{ji}(0) = \frac{1}{2} + \frac{1}{2} \prod_{i' \in C_j \setminus i} (1 - 2q_{i'j}(1)) \quad (\text{B.2})$$

et

$$r_{ji}(1) = 1 - r_{ji}(0) \quad (\text{B.3})$$

$C_j \setminus i$  est l'ensemble des nœuds de variable sauf  $c_i$ . Dans cette étape, on calcule la probabilité qu'il y ait un nombre pair de "1" parmi tous les nœuds de variable sauf  $c_i$ .

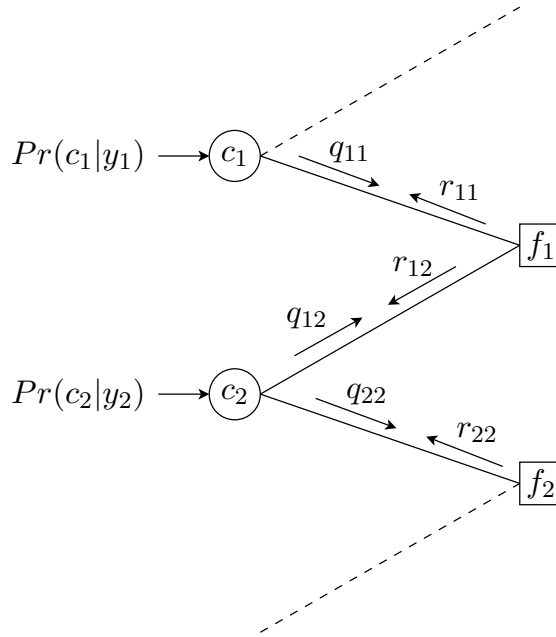
3. Les nœuds de variable mettent à jour leurs messages de réponse vers les nœuds de contrôle :

$$q_{ij}(0) = K_{ij}(1 - P_i) \prod_{j' \in F_i \setminus j} r_{j'i}(0) \quad (\text{B.4})$$

$$q_{ij}(1) = K_{ij}P_i \prod_{j' \in F_i \setminus j} r_{j'i}(1) \quad (\text{B.5})$$

où les constantes  $K_{ij}$  sont choisies de façon à ce que  $q_{ij}(0) + q_{ij}(1) = 1$  et  $F_i \setminus j$  est l'ensemble des nœuds de contrôle sauf  $f_j$ .

À cette étape, les nœuds de variable mettent également à jour leur estimation actuelle  $\hat{c}_i$  de leur variable  $c_i$  en calculant les probabilités qu'elle soit égale à "0" et "1" et

FIGURE B.2 – Propagation de croyance sur le graphe biparti  $\mathbf{H}$ .

en choisissant la valeur dont la probabilité est la plus grande. Les probabilités sont obtenues par les équations suivantes :

$$Q_i(0) = K_i(1 - P_i) \prod_{j \in F_i} r_{ji}(0) \quad (\text{B.6})$$

et

$$Q_i(1) = K_i P_i \prod_{j \in F_i} r_{ji}(1) \quad (\text{B.7})$$

Ces équations sont très proches de celles utilisées pour trouver  $q_{ij}$  sauf qu'ici, sont utilisés les messages provenant de chaque nœud de contrôle. L'estimation de la variable au nœud  $c_i$  est calculée par :

$$\hat{c}_i = \begin{cases} 1 & \text{si } Q_i(1) > Q_i(0) \\ 0 & \text{sinon} \end{cases} \quad (\text{B.8})$$

Si la valeur actuelle du mot de code estimé satisfait la condition  $\mathbf{H}^T \hat{\mathbf{c}} = \mathbf{0}$  ou si un nombre maximum d'itérations a été atteint, l'algorithme s'arrête.

#### 4. Aller à l'étape 2.

Pour éviter que l'algorithme converge vers zéro avec des blocs de grande taille, on peut faire les calculs dans le domaine logarithmique et remplacer les multiplications par des additions. Ainsi, on obtient un algorithme plus stable et plus efficace car les additions sont moins coûteuses. De plus, l'encodage peut être simplifié en utilisant  $\mathbf{H}$ , la matrice de parité d'un code LDPC  $(n, k)$ , dans sa forme systématique, telle que  $\mathbf{H} = [\mathbf{P}^T | \mathbf{I}_{n-k}]$ . Alors, la matrice génératrice  $\mathbf{G}$  est simplement une matrice de la forme  $\mathbf{G} = [\mathbf{I}_k | \mathbf{P}]$ , où  $\mathbf{P}$  est une matrice de taille  $k \times (n - k)$  et  $\mathbf{I}$  est la matrice identité. Ainsi, pour obtenir un mot de code  $\mathbf{c}$  de taille  $n$  à partir d'un vecteur  $\mathbf{x}$  de taille  $k$ , il suffit de calculer  $\mathbf{c} = \mathbf{G}\mathbf{x}$ .

La figure B.3 présente une comparaison des taux d'erreurs de décodage obtenus pour des sources binaires et un bruit blanc gaussien additif avec un code LDPC (3, 6) régulier, un turbo code et un code LDPC irrégulier optimisé. La comparaison est faite pour des blocs de taille  $10^6$ . Comme on peut le voir, le turbo code est plus performant que le code LDPC (3, 6) régulier mais moins performant que le code LDPC irrégulier optimisé.

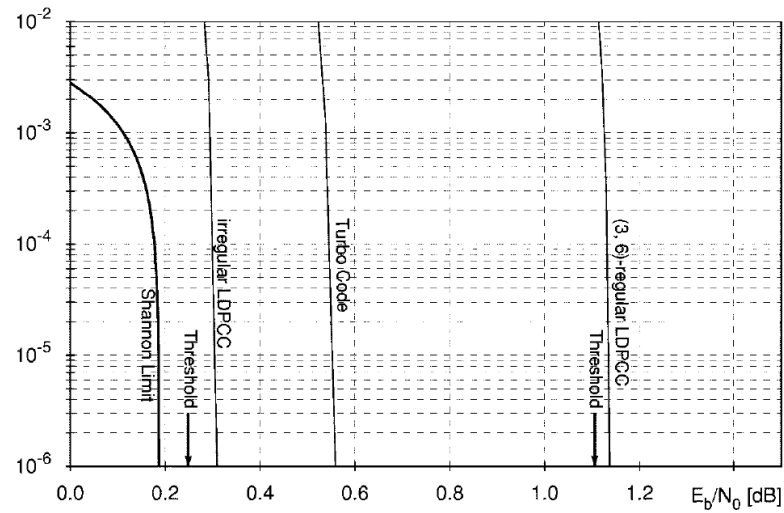


FIGURE B.3 – Comparaison des taux d'erreurs de décodage obtenus sur des sources binaires avec un code LDPC régulier (3, 6), un turbo code et un code LDPC irrégulier optimisé. Le canal est perturbé par un un bruit blanc gaussien additif (Extrait de [163]).

# Publications

## Articles de revues publiés

1. O. Crave, C. Tillier, C. Guillemot, B. Pesquet-Popescu, "Distributed temporal multiple description coding for robust video transmission", *EURASIP Journal on Wireless Communications and Networking*, Special Issue on Multimedia over Wireless Networks, January 2008.

## Articles de revues soumis

2. O. Crave, C. Guillemot, B. Pesquet-Popescu, "Multiple description coding with side information : Practical scheme and iterative decoding", *EURASIP Journal on Advances in Signal Processing*.
3. O. Crave, B. Pesquet-Popescu, C. Guillemot, "Robust video coding based on multiple description scalar quantization with side information", *IEEE Transactions on Circuits and Systems for Video Technology*.

## Articles de conférences publiés

4. O. Crave, C. Guillemot, B. Pesquet-Popescu, "Multiple description video coding and iterative decoding of LDPCA codes with side information", *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2009*, Taipei, April 19-24, 2009.
  5. O. Crave, C. Guillemot, B. Pesquet-Popescu, "Multiple description source coding with side information", *European Signal Processing Conference (EUSIPCO)*, Lausanne, August 25-29, 2008.
  6. O. Crave, C. Guillemot, B. Pesquet-Popescu, C. Tillier, "Robust video transmission based on distributed multiple description coding", *European Signal Processing Conference (EUSIPCO)*, Poznań, September 4-8, 2007.
  7. C. Tillier, O. Crave, B. Pesquet-Popescu, C. Guillemot, "A comparison of four video multiple description coding schemes", *European Signal Processing Conference (EUSIPCO)*, Poznań, September 4-8, 2007.
-



---

# Bibliographie

- [1] A. E. Gamal and T. Cover, "Achievable rates for multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 28, pp. 851–857, Nov 1982.
  - [2] J. Wolf, A. Wyner, and J. Ziv, "Source coding for multiple descriptions," *AT & T Technical Journal*, vol. 59, pp. 1417–1426, oct 1980.
  - [3] L. Ozarow, "On a source coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, pp. 1909–1921, Dec. 1980.
  - [4] V. K. Goyal and J. Kovacevic, "Generalized multiple description coding with correlating transforms," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2199–2224, Sept. 2001.
  - [5] V. K. Goyal, *Beyond Traditional Transform Coding*. PhD thesis, University of California, Berkeley, 1998.
  - [6] R. Zamir, "Gaussian codes and Shannon bounds for multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 45, pp. 2629–2636, Nov. 1999.
  - [7] H. Witsenhausen, "On source networks with minimal breakdown degradation," *Bell Syst. Tech. J.*, vol. 59, no. 6, pp. 1083–1087, 1980.
  - [8] H. Witsenhausen, "Minimizing the worst-case distortion in channel splitting," *Bell Syst. Tech. J.*, vol. 60, no. 8, pp. 1979–1983, 1981.
  - [9] Z. Zhang and T. Berger, "New results in binary multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 33, pp. 502–521, Jul 1987.
  - [10] R. Venkataramani, G. Kramer, and V. K. Goyal, "Bounds on the achievable region for certain multiple description coding problems," in *Proc. IEEE International Symposium on Information Theory*, p. 148, 24–29 June 2001.
  - [11] R. Puri, S. S. Pradhan, and K. Ramchandran, "n-channel symmetric multiple descriptions : new rate regions," in *Proc. IEEE International Symposium on Information Theory*, p. 93, 2002.
  - [12] R. Puri, S. S. Pradhan, and K. Ramchandran, "n-channel multiple descriptions : theory and constructions," in *Proc. Data Compression Conference DCC 2002*, pp. 262–271, 2–4 April 2002.
  - [13] S. S. Pradhan, R. Puri, and K. Ramchandran, "(n,k) source channel erasure codes : can parity bits also refine quality?," in *Proc. Conference on Information Sciences and Systems (CISS)*, March 2001.
  - [14] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1737–1744, Nov. 1996.
  - [15] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, Jan 1976.
-



- 
- [16] S. S. Pradhan, R. Puri, and K. Ramchandran, "n-channel symmetric multiple descriptions - part i : (n, k) source-channel erasure codes," *IEEE Trans. Inform. Theory*, vol. 50, pp. 47–61, Jan. 2004.
- [17] R. Puri, S. S. Pradhan, and K. Ramchandran, "n-channel symmetric multiple descriptions-part ii :an achievable rate-distortion region," *IEEE Trans. Inform. Theory*, vol. 51, pp. 1377–1392, April 2005.
- [18] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 39, pp. 821–834, May 1993.
- [19] T. Y. Berger-Wolf and E. M. Reingold, "Index assignment for multichannel communication under failure," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2656–2668, Oct. 2002.
- [20] V. A. Vaishampayan and J. Domaszewicz, "Design of entropy-constrained multiple-description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 40, pp. 245–250, Jan. 1994.
- [21] L. Buzi, V. A. Vaishampayan, and R. Laroia, "Design and asymptotic performance of a structured multiple description vector quantizer," in *Proc. IEEE International Symposium on Information Theory*, p. 459, 27 June–1 July 1994.
- [22] S. D. Servetto, V. A. Vaishampayan, and N. J. A. Sloane, "Multiple description lattice vector quantization," in *Proc. Data Compression Conference DCC '99*, pp. 13–22, 29–31 March 1999.
- [23] V. A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto, "Multiple-description vector quantization with lattice codebooks : design and analysis," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1718–1734, July 2001.
- [24] H. S. Malvar and D. H. Staelin, "The lot : transform coding without blocking effects," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, pp. 553–559, April 1989.
- [25] D.-M. Chung and Y. Wang, "Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 895–908, Sept. 1999.
- [26] J. Lebrun and M. Vetterli, "Balanced multiwavelets theory and design," *IEEE Trans. Signal Processing*, vol. 46, pp. 1119–1125, April 1998.
- [27] J. Lebrun and M. Vetterli, "High-order balanced multiwavelets : theory, factorization, and design," *IEEE Trans. Signal Processing*, vol. 49, pp. 1918–1930, Sept. 2001.
- [28] Z. Chen, C. Guillemot, and R. Ansari, "Multiple description coding using multiwavelet transforms," in *Proc. Picture Coding Symposium*, 2001.
- [29] Y. Wang, M. T. Orchard, and A. R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. IEEE First Workshop on Multimedia Signal Processing*, pp. 419–424, 23–25 June 1997.
- [30] M. T. Orchard, Y. Wang, V. Vaishampayan, and A. R. Reibman, "Redundancy rate-distortion analysis of multiple description coding using pairwise correlating transforms," in *Proc. International Conference on Image Processing*, vol. 1, pp. 608–611, 26–29 Oct. 1997.
-

- 
- [31] V. K. Goyal and J. Kovacevic, "Optimal multiple description transform coding of gaussian vectors," in *Proc. Data Compression Conference DCC '98*, pp. 388–397, 30 March–1 April 1998.
- [32] V. K. Goyal, M. Vetterli, and J. Kovacevic, "Multiple description transform coding : robustness to erasures using tight frame expansions," in *Proc. IEEE International Symposium on Information Theory*, p. 408, 16–21 Aug. 1998.
- [33] P. L. Dragotti, S. D. Servetto, and M. Vetterli, "Analysis of optimal filter banks for multiple description coding," in *Proc. Data Compression Conference DCC 2000*, pp. 323–332, 28–30 March 2000.
- [34] P. L. Dragotti, S. D. Servetto, and M. Vetterli, "Optimal filter banks for multiple description coding : analysis and synthesis," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2036–2052, July 2002.
- [35] X. Yang and K. Ramchandran, "Optimal multiple description subband coding," in *Proc. International Conference on Image Processing ICIP 98*, vol. 1, pp. 654–658, 4–7 Oct. 1998.
- [36] X. Yang and K. Ramchandran, "Optimal subband filter banks for multiple description coding," *IEEE Trans. Inform. Theory*, vol. 46, pp. 2477–2490, Nov. 2000.
- [37] V. K. Goyal, J. Kovacevic, and J. A. Kelner, "Quantized frame expansions with erasures," *Applied and Computational Harmonic Analysis*, vol. 10, pp. 203–233, 2001.
- [38] R. Duffin and A. Schaffer, "A class of nonharmonic fourier series," *Trans. Amer. Math. Soc.*, vol. 72, pp. 341–366, 1952.
- [39] B. Pesquet-Popescu and J.-C. Pesquet, *Le Traitement des Images*, ch. Ondelettes et traitement d'images, pp. 193–212. Hermes-Lavoisier, 2003.
- [40] J. Kovacevic, P. L. Dragotti, and V. K. Goyal, "Filter bank frame expansions with erasures," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1439–1450, June 2002.
- [41] R. Motwani and C. Guillemot, "2-channel oversampled filter banks as joint source-channel codes for erasure channels," in *Proc. International Conference on Image Processing ICIP 2003*, vol. 1, pp. 1–69–72, 14–17 Sept. 2003.
- [42] G. M. Davis and J. M. Danskin, "Joint source and channel coding for image transmission over lossy packet networks," in *Applications of Digital Image Processing XIX* (A. G. Tescher, ed.), vol. 2847, pp. 376–387, SPIE, 1996.
- [43] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Graceful degradation over packet erasure channels through forward error correction," in *Proc. Data Compression Conference DCC '99*, pp. 92–101, 29–31 March 1999.
- [44] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Generalized multiple description coding through unequal loss protection," in *Proc. International Conference on Image Processing ICIP 99*, vol. 1, pp. 411–415, 1999.
- [45] R. Puri and K. Ramchandran, "Multiple description source coding using forward error correction codes," in *Proc. Asilomar Conference on Signals, Systems, and Computers*, vol. 1, pp. 342–346, 24–27 Oct. 1999.
- [46] H. Coward, R. Knopp, and S. D. Servetto, "On the performance of multiple description codes over bit error channels," in *Proc. IEEE International Symposium on Information Theory*, p. 240, 24–29 June 2001.
-

- 
- [47] J. N. Laneman, E. Martinian, G. W. Wornell, and J. G. Apostolopoulos, "Source-channel diversity for parallel channels," *IEEE Trans. Inform. Theory*, vol. 51, pp. 3518–3539, Oct. 2005.
- [48] M. Srinivasan, "Iterative decoding of multiple descriptions," in *Proc. Data Compression Conference DCC '99*, pp. 463–472, 29–31 March 1999.
- [49] J. Barros, J. Hagenauer, and N. Gortz, "Turbo cross decoding of multiple descriptions," in *Proc. IEEE International Conference on Communications ICC 2002*, vol. 3, pp. 1398–1402, 28 April–2 May 2002.
- [50] T. Guionnet, C. Guillemot, and E. Fabre, "Soft decoding of multiple descriptions," in *Proc. IEEE International Conference on Multimedia and Expo ICME '02*, vol. 2, pp. 601–604, 26–29 Aug. 2002.
- [51] M. Alasti, K. Sayrafian-Pour, A. Ephremides, and N. Farvardin, "Multiple description coding in networks with congestion problem," *IEEE Trans. Inform. Theory*, vol. 47, pp. 891–902, March 2001.
- [52] G. Barrenechea, B. Beferull-Lozano, V. Abhishek, P. L. Dragotti, and M. Vetterli, "Multiple description source coding and diversity routing : A joint source channel coding approach to real-time services over dense networks," in *International Packet Video Workshop*, IEEE, 2003.
- [53] Y. Wang, A. R. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proc. IEEE*, vol. 93, pp. 57–70, Jan. 2005.
- [54] A. Ingle and V. A. Vaishampayan, "Dpcm system design for diversity systems with applications to packetized speech," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 48–58, Jan. 1995.
- [55] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," *IEEE Signal Processing Magazine*, vol. 17, pp. 61–82, July 2000.
- [56] J. G. Apostolopoulos, "Error-resilient video compression through the use of multiple states," in *Proc. International Conference on Image Processing*, vol. 3, pp. 352–355, 10–13 Sept. 2000.
- [57] V. A. Vaishampayan and S. John, "Balanced interframe multiple description video compression," in *Proc. International Conference on Image Processing ICIP 99*, vol. 3, pp. 812–816, 24–28 Oct. 1999.
- [58] A. R. Reibman, H. Jafarkhani, Y. Wang, M. T. Orchard, and R. Puri, "Multiple description coding for video using motion compensated prediction," in *Proc. International Conference on Image Processing ICIP 99*, vol. 3, pp. 837–841, 24–28 Oct. 1999.
- [59] A. R. Reibman, H. Jafarkhani, Y. Wang, M. T. Orchard, and R. Puri, "Multiple-description video coding using motion-compensated temporal prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 193–204, March 2002.
- [60] C.-S. Kim, R.-C. Kim, and S.-U. Lee, "Robust transmission of video sequence using double-vector motion compensation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 1011–1021, Sept. 2001.
- [61] Y. Wang and S. Lin, "Error-resilient video coding using multiple description motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 438–452, June 2002.
-

- 
- [62] G. Wang, S. Futemma, and E. Itakura, "Fec-based scalable multiple description coding for overlay network streaming," in *Proc. CCNC Consumer Communications and Networking Conference 2005 Second IEEE*, pp. 406–410, 3–6 Jan. 2005.
- [63] A. El Essaili, S. Khan, W. Kellerer, and E. Steinbach, "Multiple description video transcoding," in *Proc. IEEE International Conference on Image Processing ICIP 2007*, vol. 6, pp. VI–77–VI–80, Sept. 16 2007–Oct. 19 2007.
- [64] M. van der Schaar and D. S. Turaga, "Multiple description scalable coding using wavelet-based motion compensated temporal filtering," in *Proc. International Conference on Image Processing ICIP 2003*, vol. 3, pp. III–489–92, 14–17 Sept. 2003.
- [65] C. Tillier, B. Pesquet-Popescu, and M. van der Schaar, "Multiple descriptions scalable video coding," in *Proc. European Signal Processing Conference*, 2004.
- [66] S. Cho and W. A. Pearlman, "Error resilient compression and transmission of scalable video," in *Applications of Digital Image Processing XXIII*, p. 396–405, Août 2000.
- [67] J. Kim, R. M. Mersereau, and Y. Altunbasak, "Network-adaptive video streaming using multiple description coding and path diversity," in *Proc. International Conference on Multimedia and Expo ICME '03*, vol. 2, pp. II–653–6, 6–9 July 2003.
- [68] N. Franchi, M. Fumagalli, G. Gatti, and R. Lancini, "A novel error-resilience scheme for a 3-d multiple description video coder," in *Proc. Picture Coding Symposium*, Décembre 2004.
- [69] S. McCanne and M. Vetterli, "Joint source/channel coding for multicast packet video," in *Proc. International Conference on Image Processing*, vol. 1, pp. 25–28, 23–26 Oct. 1995.
- [70] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 983–1001, Aug. 1997.
- [71] J. G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," *Visual Communications and Image Processing 2001*, vol. 4310, no. 1, pp. 392–409, 2000.
- [72] M. Pereira, M. Antonini, and M. Barlaud, "Multiple description image and video coding for wireless channels," *EURASIP : Image Commun., Special Issue on Recent Advances in Wireless Video*, vol. 18, pp. 925–945, nov 2003.
- [73] M. U. Pereira, M. Antonini, and M. Barlaud, "Channel adapted multiple description coding scheme using wavelet transform," in *Proc. International Conference on Image Processing 2002*, vol. 2, pp. II–197–II–200, 22–25 Sept. 2002.
- [74] C. Parisot, M. Antonini, and M. Barlaud, "3d scan based wavelet transform for video coding," in *Proc. IEEE Fourth Workshop on Multimedia Signal Processing*, pp. 403–408, 3–5 Oct. 2001.
- [75] V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to-peer streaming," in *Proc. IEEE International Conference on Network Protocols*, pp. 16–27, 4–7 Nov. 2003.
- [76] E. Akyol, A. M. Tekalp, and M. R. Civanlar, "A flexible multiple description coding framework for adaptive peer-to-peer video streaming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, pp. 231–245, Aug. 2007.
- [77] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, pp. 471–480, Jul 1973.
-

- 
- [78] T. Cover and J. Thomas, *Elements of Information Theory*. New York, NY, USA, 1991.
- [79] S. S. Pradhan, J. Chou, and K. Ramchandran, "Duality between source coding and channel coding and its extension to the side information case," *IEEE Transactions on Information Theory*, vol. 49, pp. 1181–1203, May 2003.
- [80] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (discus) : design and construction," in *Proc. Data Compression Conference DCC '99*, pp. 158–167, 29–31 March 1999.
- [81] T. Berger, *The Information Theory Approach to Communications (CISM Courses and Lectures, no. 229)*, ch. Multiterminal source coding, pp. 171–231. G. Longo, 1978.
- [82] Y. Oohama, "Gaussian multiterminal source coding," *IEEE Trans. Inform. Theory*, vol. 43, pp. 1912–1923, Nov. 1997.
- [83] J. Chen, X. Zhang, T. Berger, and S. Wicker, "The sum-rate distortion function and optimal rate allocation for the quadratic gaussian ceo problem," *IEEE Journal on Selected Areas in Communications*, vol. 22, pp. 977–987, 2004.
- [84] Y. Oohama, "Multiterminal source coding problem with several side informations at the decoder," in *Proc. IEEE International Symposium on Information Theory ISIT 2008*, pp. 687–691, 6–11 July 2008.
- [85] A. K. Al Jabri and S. Al-Issa, "Zero-error codes for correlated information sources," in *Proc. of Cryptography*, 1997.
- [86] Q. Zhao and M. Effros, "Optimal code design for lossless and near lossless source coding in multiple access networks," in *Proc. DCC 2001. Data Compression Conference*, pp. 263–272, 27–29 March 2001.
- [87] Q. Zhao and M. Effros, "Lossless and near-lossless source coding for multiple access networks," *IEEE Transactions on Information Theory*, vol. 49, pp. 112–128, Jan. 2003.
- [88] J. Garcia-Frias, "Compression of correlated binary sources using turbo codes," *IEEE Communications Letters*, vol. 5, pp. 417–419, Oct. 2001.
- [89] J. Garcia-Frias and Y. Zhao, "Data compression of unknown single and correlated binary sources using punctured turbo codes," in *Proc. of the 39th Annual Allerton Conference on Communication, Control, and Computing*, 2001.
- [90] Y. Zhao and J. Garcia-Frias, "Joint estimation and compression of correlated nonbinary sources using punctured turbo codes," *IEEE Transactions on Communications*, vol. 53, pp. 385–390, March 2005.
- [91] J. Bajcsy and P. Mitran, "Coding for the slepian-wolf problem with turbo codes," in *Proc. IEEE Global Telecommunications Conference GLOBECOM '01*, vol. 2, pp. 1400–1404, 25–29 Nov. 2001.
- [92] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, Mar 1974.
- [93] A. Aaron and B. Girod, "Compression with side information using turbo codes," in *Proc. Data Compression Conference DCC 2002*, pp. 252–261, 2–4 April 2002.
-

- 
- [94] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using ldpc codes," *IEEE Communications Letters*, vol. 6, pp. 440–442, Oct. 2002.
- [95] J. Garcia-Frias and W. Zhong, "Ldpc codes for compression of multi-terminal sources with hidden markov correlation," *IEEE Communications Letters*, vol. 7, pp. 115–117, March 2003.
- [96] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (discus) : design and construction," *IEEE Trans. Inform. Theory*, vol. 49, pp. 626–643, March 2003.
- [97] H. Witsenhausen and A. Wyner, "Interframe coder for video signals." United States Patent 4191970, March 1980.
- [98] R. Puri and K. Ramchandran, "Prism : A new robust video coding architecture based on distributed compression principles," in *40th Allerton Conference on Communication, Control and Computing*, (Allerton), October 2002.
- [99] A. Aaron, R. Zhang, and B. Girod, "Wyner-ziv coding of motion video," in *Proc. Asilomar Conference on Signals and Systems*, 2002.
- [100] R. Puri, A. Majumdar, and K. Ramchandran, "Prism : A video coding paradigm with motion estimation at the decoder," *IEEE Trans. Image Processing*, vol. 16, pp. 2436–2448, Oct. 2007.
- [101] F. J. Macwilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes*. North-Holland, 1977.
- [102] M. P. C. Fossorier and S. Lin, "Soft decision decoding of linear block codes based on ordered statistics," in *Proc. IEEE International Symposium on Information Theory*, p. 395, 27 June–1 July 1994.
- [103] K. Lengwehasarit and A. Ortega, "Probabilistic partial-distance fast matching algorithms for motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 139–152, Feb 2001.
- [104] A. Aaron, E. Setton, and B. Girod, "Towards practical wyner-ziv coding of video," in *Proc. International Conference on Image Processing ICIP 2003*, vol. 3, pp. III–869–72, 14–17 Sept. 2003.
- [105] A. Aaron, S. D. Rane, E. Setton, and B. Girod, "Transform-domain wyner-ziv codec for video," *Visual Communications and Image Processing 2004*, vol. 5308, no. 1, pp. 520–528, 2004.
- [106] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, pp. 71–83, Jan. 2005.
- [107] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The discover codec : Architecture, techniques and evaluation," in *Proc. Picture Coding Symposium*, (Lisboa, Portugal), IEEE, 2007.
- [108] K. Lajnef, *Étude du codage de sources distribuées pour de nouveaux concepts en compression vidéo*. PhD thesis, Université de Rennes 1, 2006.
- [109] M. Tagliasacchi, J. Pedro, F. Pereira, and S. Tubaro, "An efficient request stopping method at the turbo decoder in distributed video coding," in *Proc. European Signal Processing Conference*, 2007.
-

- 
- [110] C. Brites and F. Pereira, "Encoder rate control for transform domain wyner-ziv video coding," in *Proc. IEEE International Conference on Image Processing ICIP 2007*, vol. 2, pp. II-5-II-8, Sept. 16 2007–Oct. 19 2007.
- [111] J. Areia, J. Ascenso, C. Brites, and F. Pereira, "Low complexity hybrid rate control for lower complexity wyner-ziv video decoding," in *Proc. European Signal Processing Conference*, 2008.
- [112] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain wyner&iv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, pp. 1177–1190, Sept. 2008.
- [113] J. Ascenso, C. Brites, and F. Pereira, "Content adaptive wyner-ziv video coding driven by motion activity," in *Proc. IEEE International Conference on Image Processing*, pp. 605–608, 8–11 Oct. 2006.
- [114] C. Yaacoub, J. Farah, and B. Pesquet-Popescu, "New adaptive algorithms for gop size control with return channel suppression in wyner-ziv video coding." accepted for publication in the International Journal of Digital Multimedia Broadcasting, special issue on "Advances in Video Coding for Broadcast Applications", October 2008.
- [115] J. Ascenso, C. Brites, and F. Pereira, "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding," in *Proc. of the 5th EURASIP Conf. on Speech, Image Processing, Multimedia Communications and Services*, July 2005.
- [116] L. Alparone, M. Barni, F. Bartolini, and V. Cappellini, "Adaptively weighted vector-median filters for motion-fields smoothing," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP-96*, vol. 4, pp. 2267–2270, 7–10 May 1996.
- [117] "The DISCOVER codec evaluation."  
<http://www.img.lx.it.pt/~discover/home.html>.
- [118] J. Pedro, L. Soares, C. Brites, J. Ascenso, F. Pereira, C. Bandeirinha, S. Ye, F. Dufaux, and T. Ebrahimi, "Studying error resilience performance for a feedback channel based transform domain wyner-ziv video codec," in *Proc. Picture Coding Symposium*, 2007.
- [119] C. Tonoli, M. Dalai, P. Migliorati, and R. Leonardi, "Error resilience performance evaluation of a distributed video codec," in *Proc. Picture Coding Symp.*, 2007.
- [120] Q. Xu, V. Stankovic, and Z. Xiong, "Distributed joint source-channel coding of video using raptor codes," *IEEE Journal on Selected Areas in Communications*, vol. 25, pp. 851–861, May 2007.
- [121] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, vol. 52, pp. 2551–2567, June 2006.
- [122] M. Luby, M. Watson, T. Gasiba, T. Stockhammer, and W. Xu, "Raptor codes for reliable download delivery in wireless broadcast systems," in *Proc. 3rd IEEE Consumer Communications and Networking Conference CCNC 2006*, vol. 1, pp. 192–197, 8–10 Jan. 2006.
- [123] M. Mitzenmacher, "Digital fountains : a survey and look forward," in *Proc. IEEE Information Theory Workshop*, pp. 271–276, 24–29 Oct. 2004.
- [124] S. Shamai, S. Verdú, and R. Zamir, "Systematic lossy source/channel coding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 564–579, March 1998.
-

- 
- [125] S. Rane, A. Aaron, and B. Girod, "Systematic lossy forward error protection for error-resilient digital video broadcasting - a wyner-ziv coding approach," in *Proc. International Conference on Image Processing ICIP '04*, vol. 5, pp. 3101–3104, 24–27 Oct. 2004.
- [126] J. Wang, A. Majumdar, K. Ramchandran, and H. Garudadri, "Robust video transmission over a lossy network using a distributed source coded auxiliary channel," in *Proc. Picture Coding Symposium*, (San Francisco, CA, USA), December 2004.
- [127] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-ziv coding of video : an error-resilient compression framework," *IEEE Trans. Multimedia*, vol. 6, pp. 249–258, April 2004.
- [128] S. Rane, P. Baccichet, and B. Girod, "Modelling and optimization of a systematic lossy error protection system based on h.264/avc redundant slices," in *Proc. Picture Coding Symposium*, 2006.
- [129] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 966–976, June 2000.
- [130] M. Fumagalli, M. Tagliasacchi, and S. Tubaro, "Improved bit allocation in an error-resilient scheme based on distributed source coding," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2006*, vol. 2, pp. II–II, 14–19 May 2006.
- [131] A. Majumdar, R. Puri, P. Ishwar, and K. Ramchandran, "Complexity/performance trade-offs for robust distributed video coding," in *Proc. IEEE International Conference on Image Processing ICIP 2005*, vol. 2, pp. II–678–81, 11–14 Sept. 2005.
- [132] Y. Steinberg and N. Merhav, "On successive refinement for the wyner-ziv problem," *IEEE Transactions on Information Theory*, vol. 50, pp. 1636–1654, Aug. 2004.
- [133] Q. Xu and Z. Xiong, "Layered wyner&ziv video coding," *IEEE Trans. Image Processing*, vol. 15, pp. 3791–3803, Dec. 2006.
- [134] M. Tagliasacchi, A. Majumdar, and K. Ramchandran, "A distributed source coding based spatio-temporal scalable video codec," in *Proc. Picture Coding Symp.*, (San Francisco, CA), December 2004.
- [135] H. Wang, N.-M. Cheung, and A. Ortega, "A framework for adaptive scalable video coding using wyner-ziv techniques," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 267 – 267, 2006.
- [136] M. Ouaret, F. Dufaux, and T. Ebrahimi, "Codec-independent scalable distributed video coding," in *Proc. IEEE International Conference on Image Processing ICIP 2007*, vol. 3, pp. III–9–III–12, Sept. 16 2007–Oct. 19 2007.
- [137] P. Ishwar, R. Puri, S. Sandeep Pradhan, and K. Ramchandran, "On compression for robust estimation in sensor networks," in *Proc. IEEE International Symposium on Information Theory*, p. 193, 29 June–4 July 2003.
- [138] J. Chen and T. Berger, "Robust coding schemes for distributed sensor networks with unreliable sensors," in *Proc. International Symposium on Information Theory ISIT 2004*, p. 116, 27 June–2 July 2004.
- [139] J. Chen and T. Berger, "Robust distributed source coding," *IEEE Trans. Inform. Theory*, vol. 54, pp. 3385–3398, Aug. 2008.
-



- 
- [140] A. Saxena, J. Nayak, and K. Rose, "On efficient quantizer design for robust distributed source coding," in *Proc. Data Compression Conference DCC 2006*, pp. 63–72, 28–30 March 2006.
- [141] K. Rose, "Deterministic annealing for clustering, compression, classification, regression, and related optimization problems," *Proc. IEEE*, vol. 86, pp. 2210–2239, Nov. 1998.
- [142] S. N. Diggavi and V. A. Vaishampayan, "On multiple description source coding with decoder side information," in *Proc. IEEE Information Theory Workshop*, pp. 88–93, 24–29 Oct. 2004.
- [143] J. Wang, X. Wu, S. Yu, and J. Sun, "Multiple descriptions in the Wyner-Ziv setting," in *Proc. IEEE International Symposium on Information Theory*, pp. 1584–1588, July 2006.
- [144] J. Wang, X. Wu, S. Yu, and J. Sun, "Multiple descriptions with side informations also known at the encoder," in *Proc. IEEE International Symposium on Information Theory ISIT 2007*, pp. 1771–1775, 24–29 June 2007.
- [145] M. Wu, A. Vetro, and C. Wen, "Multiple-description image coding with distributed source coding and side information," *Multimedia Systems and Applications VII*, vol. 5600, pp. 120–127, December 2004.
- [146] C. Tillier and B. Pesquet-Popescu, "3d, 3-band, 3-tap temporal lifting for scalable video coding," in *Proc. International Conference on Image Processing ICIP 2003*, vol. 2, pp. II–779–82, 14–17 Sept. 2003.
- [147] C. Tillier, B. Pesquet-Popescu, and M. van der Schaar, "Improved update operators for lifting-based motion-compensated temporal filtering," *IEEE Signal Processing Lett.*, vol. 12, pp. 146–149, Feb. 2005.
- [148] P. Chen and J. W. Woods, "Bidirectional mc-ezbc with lifting implementation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, pp. 1183–1194, Oct. 2004.
- [149] M. Valipour and F. Lahouti, "Design of distributed channel optimized multiple description vector quantizer," in *Proc. 24th Biennial Symposium on Communications*, pp. 336–339, 24–26 June 2008.
- [150] V. M. Stankovic, S. Cheng, and Z. Xiong, "On dualities in multiterminal coding problems," *IEEE Trans. Inform. Theory*, vol. 52, pp. 307–315, Jan. 2006.
- [151] I. Bahceci, Y. Altunbasak, and T. M. Duman, "A turbo coded multiple description system for multiple antennas," in *Proc. IEEE Global Telecommunications Conference GLOBECOM '03*, vol. 7, pp. 4011–4015, 1–5 Dec. 2003.
- [152] T. Guionnet, C. Guillemot, and S. Pateux, "Embedded multiple description coding for progressive image transmission over unreliable channels," in *Proc. International Conference on Image Processing*, vol. 1, pp. 94–97, 7–10 Oct. 2001.
- [153] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding : Turbo-codes. 1," in *Proc. IEEE International Conference on Communications*, vol. 2, pp. 1064–1070, 23–26 May 1993.
- [154] M. Oualet, F. Dufaux, and T. Ebrahimi, "On Comparing JPEG 2000 and Intraframe AVC," in *SPIE Applications of Digital Image Processing XXIX*, Lecture Notes in Computer Science, IEEE, 2006.
- [155] D. Taubman, "High performance scalable image compression with ebcot," *IEEE Trans. Image Processing*, vol. 9, pp. 1158–1170, July 2000.
-

- [156] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding : turbo-codes," *IEEE Transactions on Communications*, vol. 44, pp. 1261–1271, Oct. 1996.
  - [157] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, pp. 260–269, Apr 1967.
  - [158] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. 8, pp. 21–28, January 1962.
  - [159] D. MacKay and R. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 33, pp. 457–458, Mar 1997.
  - [160] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.
  - [161] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 219–230, Feb. 1998.
  - [162] J. Pearl, *Probabilistic Reasoning in Intelligent Systems : Networks of Plausible Inference*. Santa Mateo, CA, USA : Morgan Kaufmann, September 1988.
  - [163] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, pp. 619–637, Feb 2001.
-



# Table des figures

1.1	Schéma générique de codage MD avec deux descriptions. . . . .	20
1.2	Borne inférieure (1.12) des distorsions latérales $D_1 = D_2$ lorsque le débit de base est $r$ , c'est-à-dire lorsque la distorsion centrale est $D_0 = 2^{-2r}$ . Pour des redondances élevées, les courbes se rejoignent car la borne est indépendante de $r$ (Extrait de [4]). . . . .	23
1.3	Région débit-distorsion pour une source gaussienne sans mémoire avec $D_1 = D_2$ et $R_1 = R_2$ et pour une mesure de distorsion quadratique (Extrait de [5]).	24
1.4	Codage de canal par descriptions multiples utilisant une concaténation de codes correcteur $(3, 1)$ , $(3, 2)$ et $(3, 3)$ . . . . .	25
1.5	Exemple de MDSQ avec des cellules entrelacées. Représentation <i>naturelle</i> des cellules (gauche) et représentation matricielle (droite). . . . .	26
1.6	Exemple de MDSQ qui minimise $D_0$ à $R$ fixé. Représentation <i>naturelle</i> des cellules (gauche) et représentation matricielle (droite) . . . . .	27
1.7	Exemple de MDSQ basée sur une indexation entrelacée modifiée décrite dans [18]. Cette indexation permet d'obtenir un compromis entre les distorsions centrale et latérale : $D_0$ est plus faible que dans le cas décrit par la figure 1.5, mais $D_1$ et $D_5$ sont plus élevées. Représentation <i>naturelle</i> des cellules (gauche) et représentation matricielle (droite). . . . .	27
1.8	Exemple de treillis pour la MDLVQ : le treillis hexagonal (gauche), et l'indexation optimale associée (droite). . . . .	28
1.9	Schéma MDTC pour une paire de variables gaussiennes indépendantes. . . . .	29
1.10	Schéma de codage MD basé sur une décomposition en sous-bandes. Les traits pointillés représentent une estimation linéaire. . . . .	31
1.11	Schéma de paquetisation d'un flux hiérarchique binaire. (1) Le flux est segmenté en $N$ couches successives. $R_i$ est le nombre de bits des $i$ premières couches. (2) la $i$ -ème couche est divisée en $i$ parties égales. (3) Un code en bloc (FEC) de redondance $(N, i)$ est appliqué à la $i$ -ème couche, qui est ensuite répartie dans les $N$ paquets. . . . .	33
2.1	Problème posé par Slepian et Wolf [77]. Deux sources corrélées sont codées séparément et décodées conjointement. . . . .	40
2.2	Région de Slepian et Wolf [77] des débits atteignables lorsque deux sources $X$ et $Y$ sont codées séparément et décodées conjointement. . . . .	41
2.3	Codage de source avec information adjacente. . . . .	42
2.4	Codage d'une source à valeurs continues avec information adjacente. . . . .	42
2.5	Illustration des relations entre $X$ , $Y$ et $Z$ . . . . .	43
2.6	Codage multiterminal direct et indirect de deux sources corrélées. . . . .	44
2.7	Schéma de codage distribué de Al Jabri et Al-Issa [85]. . . . .	45

2.8	Arbre de partition (gauche), étiquettes (centre) et mots de code associés (droite). . . . .	47
2.9	L'information adjacente $Y$ peut être vue comme une observation bruitée de $X$ . Exemple d'un canal binaire symétrique (gauche) et d'un canal AWGN (droite). . . . .	48
2.10	Codage des cosets dans DISCUS. Les sommets de la même couleur appartiennent au même coset. . . . .	49
2.11	Exemple d'une machine à états finis avec $k = 2$ et $n = 1$ . . . . .	51
2.12	Schéma de codage de Slepian-Wolf présenté par Aaron et Girod [93]. . . . .	51
2.13	Schéma d'un codeur de Wyner-Ziv. . . . .	53
2.14	Schéma de codage DISCUS. . . . .	54
2.15	Schéma de codage de Wyner-Ziv présenté par Aaron et Girod [93]. . . . .	55
2.16	Graphe biparti d'un code LDPC avec accumulation de syndromes (LDPCA). . . . .	56
2.17	Architecture de l'encodeur PRISM [100]. . . . .	58
2.18	Plans de bit d'un bloc de 64 coefficients. Le plan de bit 0 correspond au bit le moins significatif. Chaque couleur correspond à un codage différent : pas de codage (blanc), codage de Wyner-Ziv (gris), codage entropique (noir). . . . .	59
2.19	Architecture du décodeur de PRISM [100]. . . . .	60
2.20	Architecture du codeur de Stanford [105]. La partie grisée correspond à un codage/décodage par bande de fréquence $k$ . . . . .	61
2.21	Architecture de l'interpolateur de trame proposé dans (Extrait de [113]). . . . .	65
2.22	Comparaison des performances débit-distorsion du codeur vidéo distribué du projet DISCOVER [107], de H.263+, de H.264/AVC en mode Intra et de H.264/AVC sans estimation de mouvement (Hall Monitor, QCIF 15 fps) (Extrait de [117]). . . . .	67
2.23	Comparaison des performances débit-distorsion du codeur vidéo distribué du projet DISCOVER [107], de H.263+, de H.264/AVC en mode Intra et de H.264/AVC sans estimation de mouvement (Soccer, QCIF 15 fps) (Extrait de [117]). . . . .	67
3.1	Architecture de codage SLEP [125]. . . . .	73
3.2	Schéma de codage de Wyner-Ziv scalable avec codage prédictif de la couche de base. . . . .	74
3.3	Schéma générique de codage de source distribué robuste avec deux descriptions. . . . .	75
4.1	MDCT-T Haar : les trames vont deux à deux dans les descriptions puis un MCTF de type Haar est appliqué sur chacune des descriptions. . . . .	82
4.2	MDCT-G Haar : un MCTF de type Haar est appliqué sur des groupes de deux trames dans chaque description. . . . .	82
4.3	MDCT-T 3-bandes : les trames paires et impaires sont séparées et un MCTF 3-bandes est appliqué sur chaque description. . . . .	83
4.4	MDCT-G 3-bandes : un MCTF 3-bandes est appliqué sur des groupes de trois trames dans chaque description. . . . .	84
4.5	Implémentation de l'encodeur de description systématique avec pertes dans le domaine pixel. . . . .	86
4.6	Implémentation du décodeur de description systématique avec pertes dans le domaine pixel. . . . .	86

---

4.7	MDCD-T 3-bandes : La séquence est séparée en trames paires et impaires. Une sous-séquence est codée classiquement tandis que l'autre est codée en Wyner-Ziv. . . . .	87
4.8	MDCD-G 3-bandes : La séquence est séparée par groupes de trois trames consécutives. Une sous-séquence est codée classiquement tandis que l'autre est codée en Wyner-Ziv. . . . .	87
4.9	Implémentation du schéma d'encodage de description systématique avec pertes dans le domaine MCTF. . . . .	88
4.10	Implémentation du schéma de décodage de description systématique avec pertes dans le domaine MCTF. . . . .	89
4.11	Schéma MDCD-G 3-bandes dans le domaine MCTF : La séquence est séparée en groupes de trois trames. Une sous-séquence est codée à l'aide d'un codeur conventionnel tandis que l'autre est filtrée temporellement et seules les sous-bandes de basse fréquence sont codées en Wyner-Ziv. . . . .	89
4.12	Comparaison des performances des schémas MDCT-T Haar et MDCT-G Haar (Foreman, QCIF, 15 Hz). . . . .	90
4.13	Comparaison des performances des schémas MDCT-T Haar et MDCT-G Haar (Hall Monitor, QCIF 15 Hz). . . . .	91
4.14	Comparaison des performances des schémas MDCT-T 3-bandes et MDCT-G 3-bandes (Foreman, QCIF, 15 Hz). . . . .	92
4.15	Comparaison des performances des schémas MDCT-T 3-bandes et MDCT-G 3-bandes (Hall Monitor, QCIF 15 Hz). . . . .	92
4.16	Quatre matrices de quantification correspondants à différentes performances débit-distorsion. . . . .	93
4.17	Comparaison des distorsions centrales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	94
4.18	Comparaison des distorsions centrales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps). . . . .	94
4.19	Comparaison des distorsions latérales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	95
4.20	Comparaison des distorsions latérales du schéma MDCD-T 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps). . . . .	95
4.21	Variations centrales et latérales du PSNR entre la 50ème et la 100ème trame de la séquence Foreman (QCIF, 15 fps) at 307 kbs. . . . .	96
4.22	Comparaison des variations du PSNR au décodeur central pour le schéma MDCD-T 3-bandes et le codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	97
4.23	Comparaison des variations du PSNR aux décodeurs latéraux pour le schéma MDCD-T 3-bandes et le codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	97
4.24	Comparaison des distorsions centrales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	98
4.25	Comparaison des distorsions centrales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps). . . . .	98
4.26	Comparaison des distorsions latérales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	99
4.27	Comparaison des distorsions latérales du schéma MDCD-G 3-bandes et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps). . . . .	99
4.28	Comparaison des distorsions centrales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Foreman, QCIF 15 fps). . . . .	100

---

---

4.29	Comparaison des distorsions centrales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).	101
4.30	Comparaison des distorsions latérales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Foreman, QCIF 15 fps).	101
4.31	Comparaison des distorsions latérales du schéma MDCD-G 3-bandes dans le domaine temporel et du codec MDC 3-bandes (Hall Monitor, QCIF 15 fps).	102
5.1	Codage de source à deux descriptions avec information adjacente commune connue uniquement du décodeur.	104
5.2	Région des débits atteignables pour le problème de codage de deux descriptions avec information adjacente.	106
5.3	Assignation d'index MDSQ pour un dictionnaire central de dimension $Q = 32$ , avec (a) 1 diagonale ( $d = 0$ ), (b) 3 diagonales ( $d = 1$ ), (c) 5 diagonales ( $d = 2$ ), où $2d + 1$ est le nombre de diagonales couvertes par l'assignation d'index.	108
5.4	Implémentation de la MDSQ avec information adjacente.	108
5.5	Décodage croisé de deux descriptions avec information adjacente.	109
5.6	Schéma de codage par deux descriptions et décodage croisé au décodeur central.	109
5.7	Comparaison des débits obtenus par les schémas de codage SD et MD.	114
5.8	Comparaison des SNR obtenus par les schémas de codage SD et MD.	115
5.9	SNR atteignables par les schémas de codage SD et MD.	116
5.10	Comparaison des débits centraux obtenus par les schémas de codage MD avec et sans décodage croisé pour différentes valeurs de $d$ .	117
5.11	Comparaison des performances débit-distorsion obtenues par les schémas de codage SD et MD pour une valeur de CSNR de 10 dB au décodeur central.	118
5.12	Comparaison des performances débit-distorsion obtenues par les schémas de codage SD et MD pour une valeur de CSNR de 10 dB aux décodeurs latéraux.	118
6.1	Codage par deux descriptions avec information adjacente pour chaque décodeur.	120
6.2	Codec vidéo à deux descriptions avec information adjacente.	122
6.3	Encodeur vidéo distribué par descriptions multiples.	123
6.4	Décodeur vidéo distribué central.	123
6.5	Décodeur vidéo distribué latéral de la description 1.	124
6.6	Décodeur vidéo distribué latéral de la description 2.	124
6.7	Courbes débit-distorsion pour les trames clé des schémas SDC et MDC au décodeur central.	126
6.8	Courbes débit-distorsion pour les trames clé des schémas SDC et MDC aux décodeurs latéraux.	127
6.9	Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés turbo au décodeur central.	127
6.10	Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés turbo aux décodeurs latéraux.	128
6.11	Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés LDPC au décodeur central.	129
6.12	Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés LDPC aux décodeurs latéraux.	129

---

---

6.13	Courbes débit-distorsion pour toutes les trames des schémas SDC, <i>Discover</i> et MDC basés turbo au décodeur central. . . . .	130
6.14	Courbes débit-distorsion pour toutes les trames des schémas SDC, <i>Discover</i> et MDC basés turbo aux décodeur latéraux. . . . .	130
6.15	Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés turbo au décodeur central avec et sans décodage croisé. . . . .	131
6.16	Courbes débit-distorsion pour les trames Wyner-Ziv des schémas SDC et MDC basés LDPC au décodeur central avec et sans décodage croisé. . . . .	132
6.17	Réduction de débit grâce au décodage croisé au décodeur central et CSNR pour le schéma MDC avec $d = 0$ et un débit pour les trames clé fixé à 532 kbs. . . . .	133
6.18	Réduction de débit grâce au décodage croisé au décodeur central et CSNR pour le schéma MDC avec $d = 1$ et un débit pour les trames clé fixé à 532 kbs. . . . .	133
A.1	Schéma d'un turbo encodeur. . . . .	141
A.2	Schéma d'un code convolutif récuratif systématique à 4 états donnés par la séquence de bits $u_1u_0$ (gauche), sa table d'encodage (centre) et son treillis de transition (droite). . . . .	141
A.3	Schéma d'un turbo décodeur. . . . .	142
B.1	Graphe biparti de la matrice de parité donnée par l'équation B.1. . . . .	146
B.2	Propagation de croyance sur le graphe biparti <b>H</b> . . . . .	147
B.3	Comparaison des taux d'erreurs de décodage obtenus sur des sources binaires avec un code LDPC régulier (3, 6), un turbo code et un code LDPC irrégulier optimisé. Le canal est perturbé par un un bruit blanc gaussien additif (Extrait de [163]). . . . .	148

---



## Abstract

In this thesis, new robust video compression techniques are proposed for multi-terminal coding.

We first compare four video multiple description coding schemes based on different time splitting patterns and temporal two- or three-band motion-compensated temporal filtering (MCTF). Then, the latter schemes are extended with systematic lossy description coding where the original sequence is separated into two sub-sequences, one being coded as in the latter schemes, and the other being coded with a Wyner-Ziv encoder. This amounts to having a systematic lossy Wyner-Ziv coding of every other frame of each description. This error control approach can be used as an alternative to Automatic Repeat Request (ARQ) or Forward Error Correction (FEC), i.e. the additional bitstream can be systematically sent to the decoder or can be requested, as in ARQ. When used as a FEC mechanism, the amount of redundancy is mostly controlled by the quantization of the Wyner-Ziv data. In this context, this approach leads to satisfactory rate-distortion performance at the side decoders, however suffers from high redundancy which penalizes the central description. To cope with this problem, the approach is then extended to the use of MCTF for the Wyner-Ziv frames, in which case only the low-frequency subbands are Wyner-Ziv encoded and sent in the descriptions.

Next, we propose a multiple description coding scheme with side information at the receiver. It builds upon both multiple description coding principles and Slepian-Wolf coding principles. The input source is first quantized with a multiple description scalar quantizer (MDSQ) which introduces redundancy or correlation in the transmitted streams in order to take advantage of the path diversity. The resulting sequences of indexes are Slepian-Wolf encoded, that is separately encoded and jointly decoded. While the first step (MDSQ) plays the role of a channel code, the second one (Slepian-Wolf coding) plays the role of a source code, compressing the sequences of quantized indexes. In a second step, the cross-decoding of the two descriptions is proposed. This allows us to account for both the correlation with the side information as well as the correlation between the two descriptions.

Then, we propose a way to improve the robustness of distributed video coding while keeping good rate-distortion performance by structuring the input sequence into groups of picture, each containing one key frame and one Wyner-Ziv frame. For each frame, two descriptions are generated by a MDSQ. This provides two balanced descriptions that contain partial information about every frame in the video input. At the central decoder, iterative decoding of the two Wyner-Ziv descriptions along with side information is performed. We study the influence of the cross-decoding and of the amount of redundancy on the performance.

---

## Résumé

Cette thèse propose des techniques de compression vidéo robuste dans un contexte de codage multi-terminal.

Nous proposons des schémas de codage par descriptions multiples basés sur des filtres temporels dyadiques et 3-bandes appliqués sur des séquences sous-échantillonnées. Nous commençons par présenter plusieurs méthodes pour construire des descriptions corrélées à partir de ces bancs de filtres. Puis, pour augmenter leur robustesse et leur stabilité au cours du temps, nous proposons d'introduire un codec Wyner-Ziv au sein de ces schémas redondants : les trames qui en principe ne sont pas transmises dans une description sont ici codées en Wyner-Ziv. Pour réduire la redondance, nous proposons d'appliquer un filtrage temporel compensé en mouvement sur les trames codées en Wyner-Ziv pour ne transmettre que les sous-bandes de basses fréquences.

Ensuite, nous proposons un schéma de codage par descriptions multiples avec information adjacente basé sur une quantification scalaire à descriptions multiples (MDSQ). Dans un premier temps, les indices générés par la MDSQ sont codés séparément par un codeur de Slepian-Wolf et décodés séparément à l'aide d'une information adjacente qui sert également à la reconstruction des coefficients. Dans un deuxième temps, un décodage croisé est proposé pour décoder conjointement les indices et ainsi profiter de la redondance entre les descriptions.

Enfin, nous proposons d'améliorer la robustesse en codage vidéo distribué tout en conservant de bonnes performances débit-distorsion en structurant la séquence d'origine en groupes d'images, chacun contenant une trame clé et une trame codée en Wyner-Ziv. Pour chaque trame, deux descriptions sont générées par un quantificateur scalaire à descriptions multiples. Cette technique permet d'obtenir deux descriptions équilibrées contenant chacune une information partielle sur chaque trame dans la vidéo d'origine. Au décodeur central, le décodage itératif de deux descriptions avec information adjacente est utilisé. Nous étudions alors l'influence du décodage croisé et du taux de redondance sur les performances.

---