

UNIVERSITÉ DE LIMOGES
ÉCOLE DOCTORALE Sciences et Ingénierie pour l'Information
FACULTÉ des Sciences et Techniques
Laboratoire XLIM Département – C²S² UMR CNRS 6172

Thèse N° [38-2009]

Thèse
pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ DE LIMOGES

Discipline : Electronique des Hautes Fréquences, Photonique et Systèmes

Présentée et soutenue le 16 Octobre 2009 par

Stéphanie CASTAN - SAHUGUÈDE

**Codage de canal pour les communications
optiques**

Thèse dirigée par Anne JULIEN-VERGONJANNE et Jean-Pierre CANCES

Jury :

Président

Mme. Inbar FIJALKOW,

Professeur à l'ENSEA

Rapporteurs

M. Gilles BUREL,

Professeur à l'Université de Bretagne Occidentale

M. Daniel ROVIRAS,

Professeur au CNAM

Examineurs

M. Jean-Pierre CANCES,

Professeur à l'ENSIL

M. Dominique CHIARONI,

Ingénieur Recherche à Alcatel-Lucent

M. Philippe GALLION,

Professeur à Telecom ParisTech

Mme. Anne JULIEN-VERGONJANNE,

Professeur à l'ENSIL

Remerciements

Trois années passées dans le département C2S2 du laboratoire XLIM, au sein du groupe Etudes des Systèmes de Télécommunications de l'ENSIL, m'ont permis de mener à bien mes recherches et de concrétiser mes travaux à travers ce mémoire de thèse. A ce titre, je tiens à remercier l'ensemble de l'équipe de recherche et plus particulièrement son responsable, Monsieur Jean-Pierre CANCES, Professeur à l'ENSIL de m'avoir accueillie. Je le remercie également pour son rôle de co-directeur de thèse, il a su m'apporter soutien technique et moral afin de mener à bien cette thèse.

Mes remerciements s'adressent bien entendu à ma directrice de thèse, Madame Anne JULIEN-VERGONJANNE, Professeur à l'ENSIL, qui m'a guidée, conseillée et accompagnée durant ces trois ans. Son énergie, sa rigueur et sa disponibilité m'ont permis de réaliser l'ensemble de mes travaux dans un dynamisme de réussite et dans la bonne humeur.

Je remercie également mes collègues de bureau et bureaux voisins pour leur agréable compagnie journalière. Je pense aussi aux anciens doctorants ayant partagé mon bureau, qui m'ont accueillie au début de ma thèse, conseillée, et ont ensuite continué leur chemin.

Je remercie aussi l'ensemble des personnes qui ont accepté d'évaluer mes travaux, notamment messieurs Gilles BUREL, Professeur à l'Université de Bretagne Occidentale, et Daniel ROVIRAS, Professeur au Conservatoire National des Arts et Métiers, pour l'étude du rapport, mais aussi l'ensemble des membres du jury de ma thèse.

Enfin, je remercie ma famille, qui m'a toujours encouragée dans mes études et m'a permis de devenir qui je suis. Plus particulièrement, je pense à ma mère pour son soutien, mais aussi et surtout à Sébastien et Célestin qui partagent ma vie au quotidien.

Table des matières

Introduction Générale	xxi
1 Codes Correcteurs d'Erreurs	1
1.1 Introduction	2
1.2 Les différents schémas de codage correcteur d'erreurs	3
1.2.1 Définitions générales	3
1.2.1.1 Vocabulaire	3
1.2.1.2 Représentations polynomiales et matricielles	4
1.2.1.3 Codes convolutifs et codes en blocs	6
1.2.2 Codage	7
1.2.2.1 Codes convolutifs	7
1.2.2.2 Codes en blocs	9
1.2.2.3 Conclusions	10
1.2.3 Décodage	11
1.2.3.1 Décodage des codes convolutifs	11
1.2.3.2 Décodage des codes en blocs	13
1.2.3.3 Conclusions	14
1.2.4 Les codes correcteurs d'erreurs appliqués aux transmissions optiques	15
1.3 Codes de Reed Solomon	18
1.3.1 Introduction	18
1.3.2 Notions sur les corps de Galois	19
1.3.3 Les codes BCH	21
1.3.3.1 Paramètres caractéristiques	21
1.3.3.2 Polynômes générateurs	22
1.3.3.3 Codage	22
1.3.4 Les codes de Reed Solomon	24
1.3.4.1 Paramètres caractéristiques	24
1.3.4.2 Polynômes générateurs	25
1.3.4.3 Codage	25
1.3.5 Décodage des codes RS et BCH	27
1.3.5.1 Principe général	27

1.3.5.2	Détection d'erreurs	28
1.3.5.3	Localisation et évaluation d'erreurs	29
1.3.5.4	Correction des erreurs	30
1.3.5.5	Conclusions	31
1.3.6	Conclusion - Application des codes RS à l'optique	32
1.4	Codes LDPC	34
1.4.1	Introduction	34
1.4.2	Méthodes de construction	37
1.4.2.1	Principe général	37
1.4.2.2	Méthodes aléatoires	40
1.4.2.3	Méthodes déterministes	43
1.4.2.4	Conclusions	47
1.4.3	Décodage	48
1.4.3.1	Principe général	48
1.4.3.2	Décodage dur ou hard	49
1.4.3.3	Décodage souple ou soft	53
1.4.3.4	Conclusions	54
1.4.4	Conclusion - Application des codes LDPC aux transmissions optiques	54
1.5	Conclusion	56
2	Codage correcteur d'erreurs pour les transmissions sur fibre optique - Application au réseau métro/accès ECOFRAME	59
2.1	Introduction	61
2.2	Chaîne de transmission optique	62
2.2.1	Emetteur	62
2.2.2	Fibre optique	65
2.2.3	Amplificateurs	66
2.2.4	Circuit de réception	66
2.2.5	Conclusions	68
2.3	Modèles de canal	69
2.3.1	Modèle AWGN	69
2.3.1.1	Définition et modélisation du canal	69
2.3.1.2	Caractéristiques mathématiques	70
2.3.1.3	Paramètres de simulation	70
2.3.2	Modèle χ^2	72
2.3.2.1	Justification du modèle	72
2.3.2.2	Caractéristiques mathématiques	77
2.3.2.3	Modélisation du canal	82
2.3.2.4	Paramètres de simulation	83
2.4	Performances avec un modèle AWGN	86
2.4.1	Introduction	86

TABLE DES MATIÈRES

2.4.2	Performances dans le cas non codé	86
2.4.2.1	Expression de la probabilité d'erreur	86
2.4.2.2	Résultats de simulations	88
2.4.3	Performances avec les codes RS	90
2.4.3.1	Expression de la probabilité d'erreur	90
2.4.3.2	Résultats de simulations	94
2.4.4	Performances avec les codes LDPC	95
2.4.4.1	Contexte	95
2.4.4.2	Expressions des rapports logarithmiques de vraisemblance (LLR)	96
2.4.4.3	Analyse des performances	97
2.4.4.4	Conclusions	102
2.4.5	Conclusions	103
2.5	Performances avec le modèle χ^2	104
2.5.1	Cas non codé	104
2.5.1.1	Expression de la probabilité d'erreur	104
2.5.1.2	Résultats de simulations	109
2.5.2	Performances avec les codes RS	111
2.5.2.1	TEB sur canal χ^2	111
2.5.2.2	Comparaison performances simulées et théoriques	112
2.5.2.3	Comparaison performances RS sur canal χ^2 et AWGN	114
2.5.2.4	Conclusions	114
2.5.3	Performances avec les codes LDPC-BIBD	115
2.5.3.1	Expression des LLR	115
2.5.3.2	Analyse des performances	117
2.5.3.3	Conclusions	121
2.5.4	Conclusions	121
2.6	Conclusion	123
3	Codage correcteur d'erreurs pour des transmissions sur fibre optique utilisant la technique OCDMA	125
3.1	Introduction	127
3.2	Principe de l'OCDMA	129
3.2.1	OCDMA incohérent	129
3.2.2	Emission	131
3.2.2.1	Principe	131
3.2.2.2	Familles de codes	132
3.2.2.3	Présentation	132
3.2.2.4	Les codes 1D OOC	134
3.2.2.5	Les codes 2D MWOOC	136
3.2.3	Réception	137

TABLE DES MATIÈRES

3.2.3.1	Introduction	137
3.2.3.2	Réception par corrélation	138
3.2.3.3	Les autres récepteurs	141
3.2.4	Application de l'OCDMA aux transmissions multimédia	142
3.3	Limitations	144
3.3.1	L'Interférence d'Accès Multiple	144
3.3.2	Le beat noise	148
3.3.2.1	Contexte	148
3.3.2.2	Premier modèle : approche temporelle	151
3.3.2.3	Deuxième modèle : approche spectrale et modèle statistique	159
3.3.2.4	Comparaison et conclusions	173
3.3.3	Les autres limitations	175
3.3.3.1	Photodétection idéale	175
3.3.3.2	Battements dus à la photodétection	176
3.3.4	Conclusions	179
3.4	Amélioration des performances avec codes correcteurs d'erreurs	180
3.4.1	Contexte	180
3.4.1.1	Présentation	180
3.4.1.2	Chaîne de transmission non codée	180
3.4.1.3	Chaîne de transmission codée	182
3.4.2	Application des codes LDPC à la chaîne de transmission OCDMA .	182
3.4.2.1	Présentation	182
3.4.2.2	Evaluation des performances en présence d'IAM (cas idéal)	184
3.4.2.3	Evaluation des performances en présence d'IAM et de bruit gaussien	187
3.4.2.4	Evaluation des performances en présence d'IAM et de beat noise	192
3.4.2.5	Conclusions	198
3.4.3	Comparaison des performances des codes LDPC avec les codes RS .	199
3.5	Application des LDPC à un réseau multiservice utilisant la technique OCDMA	201
3.5.1	Description du système	201
3.5.1.1	Principe	201
3.5.1.2	Analyse des performances	202
3.5.2	Performances du système multimédia avec codes LDPC	206
3.5.2.1	Adaptation des LLR au contexte multimédia OCDMA . . .	206
3.5.2.2	Gain en TEB et en S/B	206
3.5.2.3	Gain en nombre d'utilisateurs actifs et en débits	208
3.5.2.4	Impact d'un limiteur sur les performances	209
3.5.3	Conclusions	212
3.6	Conclusion	213

TABLE DES MATIÈRES

Conclusion générale et perspectives	215
Glossaire	221
Références	235
A Caractéristiques du code RS(255,239) et du corps GF(2⁸)	237
A.1 Propriétés du code RS(255,239)	237
A.2 Propriétés du corps GF(2 ⁸)	238
A.3 Codage et polynôme générateur du RS(255,239)	239
B Algorithmes de décodage des codes RS et BCH	243
B.1 Algorithme d'Euclide	243
B.2 Algorithme de Berlekamp-Massey	244
C Méthodes de construction des codes LDPC aléatoires	247
C.1 Construction de Gallager : régulier aléatoire	247
C.2 Construction de MacKay : régulier aléatoire	248
C.3 Construction de Luby : non-régulier aléatoire	250
D Algorithmes de décodage à propagation de croyance	251
D.1 Notations générales	251
D.2 Le décodeur somme-produit ou à propagation de croyance	252
D.3 Le décodeur LLR somme-produit ou à propagation de croyance	255
E Application du codage correcteur d'erreurs à l'anneau optique ECOFRAME	257
E.1 Présentation	257
E.2 Contraintes et caractéristiques liées au contexte	258
E.3 Dimensionnement des codes RS	261
E.4 Dimensionnement des codes LDPC	271
E.5 Adaptation au canal χ^2	279
F Obtention de la densité de probabilité de la variable décisionnelle avec le modèle statistique du beat noise	281
G Calcul de la densité de probabilité de la variable de bruit globale (IAM+AWGN)	285
H Calcul de Y_L	289
Liste des publications	293

Table des figures

1.1	Exemple de codeur convolutif	7
1.2	Exemple de codage d'information à travers un codeur convolutif	8
1.3	Exemple de codeur convolutif récursif	9
1.4	Exemple de diagramme en treillis, pour un code $R=1/2$	11
1.5	Evolution des gains de codage de différents codes	17
1.6	Table des polynômes générateurs des codes BCH	23
1.7	Représentation équivalente du codeur cyclique utilisée pour le codeur RS .	27
1.8	Représentation équivalente de la matrice de parité par le graphe de Tanner	35
1.9	Chaîne de transmission avec codage/décodage	36
1.10	Exemple de cycle dans la matrice de parité et dans le graphe de Tanner équivalent	42
1.11	Exemple de matrice de parité construite avec la méthode BIBD	45
1.12	Exemple de décodage à partir du graphe de Tanner	50
2.1	Caractéristique Puissance - Courant du laser	64
2.2	Modélisation du canal AWGN	70
2.3	Densité de probabilité $p_X(x)$ d'une loi gaussienne avec $\sigma^2 = 0.1$ et $m = 0.5$	71

2.4	Densité de probabilité $p_Y(y)$ d'une variable à statistique χ^2 avec $\sigma^2 = 0.05$ et $s^2 = 0.5$	79
2.5	Densité de probabilité $p_Y(y)$ d'une variable à statistique χ^2 avec $\sigma^2 = 0.1$ et $s^2 = 0.4$	80
2.6	Densité de probabilité $p_Y(y)$ d'une variable à statistique χ^2 avec $\sigma_Y^2 = 0.5$ et $s^2 = 0.5$	82
2.7	Modélisation du canal χ^2	84
2.8	Exemple de densité de probabilité pour la variable reçue, avec $\sigma^2 = 0.1$ pour un canal AWGN	87
2.9	Probabilité d'erreur théorique sur canal AWGN	89
2.10	TEB obtenus par simulation sur canal AWGN	90
2.11	Performances théoriques sur canal AWGN des codes RS(255,K)	93
2.12	Comparaison des TEB simulés avec la probabilité d'erreur théorique pour des codes RS (255,K)	95
2.13	TEB obtenus par simulation avec les codes LDPC construits selon la méthode BIBD	98
2.14	TEB pour différents nombres maximaux d'itérations avec le LDPC(6176,5983)	99
2.15	Comparaison des performances avec les codes LDPC(2715,2535) obtenus avec la méthode BIBD et la méthode aléatoire	100
2.16	Comparaison des performances des codes LDPC(6176,5983) et RS(511,489)	102
2.17	Comparaison des performances des codes LDPC(2034,1921) et RS(255,241)	103
2.18	Exemple de densité de probabilité pour la variable reçue, avec $\sigma^2 = 0.03$ $ER_{in} = 7dB$ dans le cas d'un canal χ^2 avec $n = 4$	105

TABLE DES FIGURES

2.19	TEB en fonction du facteur de qualité Q sur un canal χ^2 avec $n = 4$ pour différentes valeurs de ER_{in}	107
2.20	TEB en fonction du facteur de qualité Q sur un canal χ^2 pour différentes valeurs de n , avec ER_{in} infini	108
2.21	TEB obtenus par simulation sur canal χ^2 , $n = 2$	110
2.22	TEB obtenus sur canal χ^2 idéal (ER_{in} infini et $n = 2$) pour différents codes RS	112
2.23	TEB obtenus sur canal χ^2 idéal (ER_{in} infini), $n = 2$ pour différents codes RS	113
2.24	TEB obtenus avec le code RS(255,239) sur canal χ^2 (ER_{in} infini, $n = 2$) avec le seuil optimal et avec le seuil gaussien de 0,5	114
2.25	TEB obtenus avec le code RS(511,495) sur canal χ^2 (ER_{in} infini, $n = 2$) et canal AWGN	115
2.26	Performances obtenues avec les codes LDPC(2715,2535) et LDPC(1810,1630) sur un canal χ^2 avec $ER_{in}=10$ dB et $n=4$	118
2.27	Comparaison des performances obtenues avec le code LDPC(6176,5983) sur canal χ^2 et canal AWGN pour $n = 4$	119
2.28	Performances avec le code LDPC (6176,5983) sur canal χ^2 avec ER_{in} infini et $n = 4$, avec calcul du LLR adapté au canal χ^2 et avec le modèle classique AWGN	120
3.1	Principe d'émission OCDMA	132
3.2	Exemple de séquence de code en 1D	134
3.3	Exemple de séquence de code en 2D	137
3.4	Réception par corrélation CCR	138
3.5	Récepteur CCR avec limiteur	141

3.6	Exemple de partitionnement pour un code MWOOC avec $L = 7$	143
3.7	Exemple de densité de probabilité de la variable décisionnelle en présence d'IAM pour la famille OOC(70,3) avec 10 utilisateurs	146
3.8	Modèle de beat noise, approche temporelle	153
3.9	Performances obtenues avec la formule théorique (3.42), comparées à celles issues de [82] en fonction du nombre d'utilisateurs U pour $M = 50$	160
3.10	Densités de probabilité conditionnelles $p^{(1)}(Z_d)$ et $p^{(0)}(Z_d)$ pour $n = 4$ et $n = 30$	167
3.11	Performances obtenues avec la famille OOC($F=534, W=3$) avec (3.68) (trait plein) et par simulation (cercles) (3.43)	171
3.12	Influence du seuil de décision S sur le TEB pour différentes valeurs de M .	172
3.13	Comparaison des performances obtenues avec beat noise en fonction de M pour la famille ($F=70, W=3$) avec le premier modèle en trait plein, et le deuxième modèle avec les ronds	173
3.14	Comparaison des performances obtenues avec beat noise en fonction de M pour la famille ($F=534, W=3$) avec le premier modèle en trait plein, et le deuxième modèle avec les ronds	174
3.15	Performances obtenues par simulation avec la famille OOC($F=534, W=3$) et $U = 20$ utilisateurs, en présence de bruit AWGN et de beat noise dû à la photodétection	178
3.16	Chaîne de transmission complète OCDMA	181
3.17	Chaîne de transmission complète OCDMA avec codage correcteur d'erreurs	183
3.18	Comparaison des densités de probabilité entre le modèle approché et la loi de répartition binomiale pour la famille MWOOC($67 \times 8, 7$) et différents nombres d'utilisateurs U	185

TABLE DES FIGURES

3.19	TEB en fonction du nombre d'utilisateurs U pour la famille MWOOC($67 \times 8, 8$) dans le cas non codé et avec le code LDPC($905, 725$)	188
3.20	Densité de probabilité globale $f_T(x)$ pour la famille MWOOC($67 \times 8, 8$) avec différents nombres d'utilisateurs actifs U et deux valeurs de σ différentes	189
3.21	TEB obtenus pour la famille MWOOC($67 \times 8, 8$) avec $U = 60$ utilisateurs dans le cas non codé et avec le code LDPC($905, 725$) en fonction du rapport signal à bruit	191
3.22	TEB en fonction du rapport $M = B_o/B_e$ pour la famille OOC($534, 3$) dans le cas non codé et avec le code LDPC($2715, 2535$) pour $U = 45$ et $U = 50$ utilisateurs	197
3.23	Comparaison des performances du code RS($255, 239$) et du code LDPC($2715, 2535$)	200
3.24	TEB pour les trois classes de services avec $M_1 = M_2 = M_3 = 16$	205
3.25	TEB pour les trois classes de services avec $M_1 = M_2 = M_3 = 16$ avec le code LDPC($2715, 2535$)	207
A.1	Puissances de α de 0 à 127	241
A.2	Puissances de α de 128 à 254	242
E.1	Structure du container Ecoframe	259
E.2	Découpage du container en trame FEC	260
E.3	Régression gaussienne et χ^2	261
E.4	TEB obtenus par simulation pour les codes RS de classe $m = 7$	263
E.5	TEB obtenus par simulation pour les codes RS de classe $m = 8$	264
E.6	TEB obtenus par simulation pour les codes RS de classe $m = 9$	265
E.7	TEB obtenus par simulation pour les codes RS de classe $m = 10$	266

E.8 TEB en fonction du S/B pour une redondance de 6,3% - $R = 0,937$ 268

E.9 Gain de codage en fonction de la taille de la trame FEC (en bits) N_{trame}
et de la redondance 269

E.10 Variation du débit en ligne en fonction de la durée du container autour de
 $10\mu s$ 270

E.11 Rendement du code LDPC en fonction de ns 272

E.12 Nombre de bits par trame FEC N_{trame} en fonction de q et de ns 273

E.13 Performances des codes LDPC(6176,5983) et LDPC(12742,12465) 277

Liste des tableaux

1.1	Polynômes primitifs les plus courants	20
1.2	Exemple : mots de code du BCH(7,4;1)	24
3.1	TEB, débit, et nombre d'utilisateurs U_i pour chaque classe de service, récepteur CCR, sans codage	205
3.2	Améliorations du système avec le code LDPC(2715, 2535), récepteur CCR	209
3.3	TEB, débit, et nombre d'utilisateurs U_i pour chaque classe de service pour le cas avec limiteur, sans bruit et sans codage	210
3.4	TEB, débit, et nombre d'utilisateurs U_i pour chaque classe de service pour le cas avec limiteur et code LDPC(905,725)	212
E.1	Nombre de bits par trame FEC et débit en ligne réel pour différentes classes de codes RS	262
E.2	Gains de codage obtenus pour différents codes RS	267
E.3	Comparaison des débits utiles des différents codes respectant le gain de codage à 10^{-12}	267
E.4	Comparaison des débits utiles avec la durée du container adaptée à chacun des codes	268
E.5	Tailles de trame FEC pour une redondance de 4% ($ns=25$)	273

E.6	Couples (q, ns) correspondant à une taille de trame FEC recherchée	274
E.7	Rendements obtenus avec différentes valeurs de ns	274
E.8	Paramètres des codes LDPC proches des codes RS de classe $m = 7, 8, 9$ et 10	275
E.9	Débits utiles et durées optimales pour différentes valeurs de ns	276

Introduction Générale

L'évolution croissante des besoins en termes de débit de transmission et de qualité de service, motive la recherche de nouveaux moyens de communications, performants et peu coûteux. De plus, le développement des applications multimédia, nécessitant le transport simultané de plusieurs services différents, contribue à l'évolution de ces besoins.

Les infrastructures du réseau historique utilisant un support cuivré ne sont pas suffisantes, notamment pour les débits de communication visés. Des solutions de nouvelle génération, basées sur un transport tout optique c'est-à-dire utilisant la fibre de bout en bout, permettent de fournir de plus hauts débits. Ainsi, de nouveaux concepts sont envisagés, pour les réseaux d'accès, métropolitains, régionaux et longues distances.

Conjointement à l'investigation de nouveaux protocoles dans la couche réseau, des solutions appliquées à la couche physique sont recherchées. Dans ce cadre, les techniques déjà utilisées pour les communications hertziennes sont envisagées pour une application à l'optique.

Parmi ces techniques, nous nous intéressons au codage correcteur d'erreurs, afin d'améliorer les performances des transmissions optiques.

Dans le contexte des futurs réseaux optiques, les performances en termes de taux d'erreurs doivent être les plus faibles possibles (10^{-9} à 10^{-12}), tout en assurant des

débits importants. Cependant, dans ce contexte, les imperfections des composants optoélectroniques introduisent de nombreuses limitations sur les performances. C'est pourquoi, les techniques de codage correcteur d'erreurs, ou codage de canal, ayant déjà fait leurs preuves dans le domaine hertzien, sont envisagées pour l'optique.

D'une manière générale, la détection et la correction des erreurs ne sont possibles que par l'ajout de données redondantes en plus des données utiles à transmettre. Le débit utile des informations est de ce fait réduit par rapport à une transmission sans codage. Ainsi, dans le cadre de transmissions optiques à très hauts débits, des codes efficaces mais peu redondants sont recherchés. En plus de la contrainte sur le débit utile, il est important, afin d'assurer des temps de traitements raisonnables, de tenir compte de la complexité liée à l'ajout du système de codage et décodage dans la chaîne de transmission.

Pour cela, plusieurs études récentes ont envisagé de remplacer les codes de Reed-Solomon, déjà normalisés pour les communications transocéaniques, par de nouveaux codes, plus performants et moins redondants, comme les codes à faible densité (LDPC : Low Density Parity Check).

Un exemple de l'enjeu de l'utilisation des codes correcteurs d'erreurs pour les communications optiques est la mise en place de projets de recherche tels que l'ANR ECOFRAME (Eléments de COnvergence pour les Futurs Réseaux d'Accès et MEtropolitains haut débit) auquel notre équipe de recherche a participé durant ces trois dernières années. L'objectif de ce projet est d'établir à l'aide de concepts réseaux innovants, un réseau métro-accès permettant de fournir de hauts débits (10 à 40 Gb/s), de minimiser le taux d'erreurs sur les données transmises (inférieur à 10^{-12}), tout en élaborant une solution peu onéreuse. Notre contribution dans ce projet consiste à dimensionner des codes correcteurs d'erreurs par rapport au contexte, afin de compenser les dégradations dues aux composants optoélectroniques à bas coût.

Par ailleurs, afin d'optimiser le partage de la ressource en bande passante de l'optique, les techniques d'accès multiple, en particulier la technique d'accès multiple par répartition temporelle (ou TDMA : Time Division Multiple Access) et la technique de multiplexage en longueurs d'onde (ou WDMA : Wavelength Division Multiple Access) sont également adaptées de l'hertzien à l'optique et sont actuellement les plus répandues.

Avec la technique TDMA, chaque temps bit est subdivisé en un nombre d'intervalles de temps égal au nombre d'utilisateurs et les données sont alors multiplexées en temporel. Ainsi, les débits et les nombres d'utilisateurs proposés en TDMA sont rapidement limités par la bande passante électronique du système. La technique WDMA quant à elle, permet à chaque utilisateur d'occuper une longueur d'onde différente pour émettre ses données, qui sont ensuite multiplexées spectralement avec les données sur les autres longueurs d'onde. Dans ce cas, la complexité augmente avec le nombre de longueurs d'onde (plusieurs lasers ou laser multi-longueurs d'onde), ce qui limite l'utilisation pour un grand nombre d'utilisateurs.

D'autres méthodes d'accès multiple issues de l'hertzien ont été envisagées telle que la technique d'accès multiple par répartition de codes (ou CDMA : Code Division Multiple Access) qui affecte à chaque utilisateur un code particulier permettant de différencier ses données de celles des autres utilisateurs. Appliquée à l'optique, la technique OCDMA (Optical CDMA) permet d'atteindre de plus hauts débits avec un plus grand nombre d'utilisateurs, par rapport aux techniques précédemment exposées.

De plus, il est possible d'utiliser les codes OCDMA pour différencier non seulement les utilisateurs mais aussi les services proposés (taux d'erreurs et débits), réalisant ainsi un système de communication multimédia. Différents travaux ont été menés dans notre équipe sur la technique OCDMA et en particulier sur une des principales limitations de cette technique, l'interférence d'accès multiple (IAM). Ainsi, il a été montré qu'une

réception multi-utilisateurs pouvait réduire l'impact de l'IAM sur les performances mais au prix d'une complexité en réception. L'application des codes correcteurs à la chaîne de transmission optique utilisant l'OCDMA peut donc constituer une solution alternative permettant de réduire les limitations liées à cette technique d'accès. Cette application est une partie du travail développé dans cette thèse.

Le premier chapitre de ce document est consacré à la présentation des codes correcteurs d'erreurs. Les différentes familles de codes, ainsi que les principes généraux du codage et décodage des informations, sont tout d'abord décrits. Les caractéristiques des codes correcteurs d'erreurs envisagés par la suite pour une application à l'optique, c'est-à-dire les codes RS (Reed Solomon) et les codes LDPC (Low Density Parity Check), sont plus particulièrement développées.

Le second chapitre constitue une étude de l'application des codes correcteurs de types RS et LDPC à un système de transmission sur fibre optique. En considérant les différents bruits pouvant dégrader le signal sur la chaîne de transmission, deux modèles de canaux optiques sont envisagés : le modèle classique à bruit additif gaussien et un modèle plus réaliste, prenant en compte la conversion quadratique en réception, appelé modèle χ^2 . Après une présentation détaillée de ces deux modèles, les performances des codes sont évaluées sur chacun d'entre eux. Nous montrons ainsi qu'il est important de connaître fidèlement la statistique du canal afin d'exploiter au mieux les performances du codage correcteur d'erreurs. Une application du dimensionnement des codes correcteurs réalisé dans le contexte du réseau ECOFRAME permet d'illustrer cette problématique.

Dans le troisième chapitre, on s'intéresse aux potentialités liées à l'application du codage correcteur d'erreurs dans une transmission OCDMA. En effet, en se basant sur les résultats du chapitre 2, le codage correcteur d'erreurs de type LDPC apparaît comme

une solution potentielle pour améliorer les performances d'un tel système. Après une présentation de la chaîne OCDMA ainsi que de son application pour les communications multimédia, l'étude de la statistique du canal est réalisée dans le but d'adapter correctement le codage correcteur d'erreurs. En plus de l'IAM, une autre limitation appelée "beat noise", liée au battement entre les signaux des différents utilisateurs en réception, est également prise en compte. En considérant les différentes perturbations inhérentes à cette technique d'accès multiple, les performances théoriques du système OCDMA sont établies de manière analytique. Des simulations permettent d'obtenir les performances de la chaîne de transmission OCDMA avec codage LDPC. L'application des codes correcteurs à une transmission multimédia utilisant la technique OCDMA permet de conclure ce chapitre et illustre les potentialités d'un tel système.

Chapitre 1

Codes Correcteurs d'Erreurs

Sommaire

1.1	Introduction	2
1.2	Les différents schémas de codage correcteur d'erreurs	3
1.2.1	Définitions générales	3
1.2.2	Codage	7
1.2.3	Décodage	11
1.2.4	Les codes correcteurs d'erreurs appliqués aux transmissions optiques	15
1.3	Codes de Reed Solomon	18
1.3.1	Introduction	18
1.3.2	Notions sur les corps de Galois	19
1.3.3	Les codes BCH	21
1.3.4	Les codes de Reed Solomon	24
1.3.5	Décodage des codes RS et BCH	27
1.3.6	Conclusion - Application des codes RS à l'optique	32
1.4	Codes LDPC	34
1.4.1	Introduction	34
1.4.2	Méthodes de construction	37
1.4.3	Décodage	48
1.4.4	Conclusion - Application des codes LDPC aux transmissions optiques	54
1.5	Conclusion	56

1.1 Introduction

Le principe du codage correcteur d'erreurs (ou codage de canal) est d'ajouter aux données utiles, des données redondantes de manière à rendre plus fiable la transmission des données utiles. Ce principe peut être couplé avec du codage de source qui consiste à rendre l'information transmise la plus concise possible sans dégrader les performances du système. Généralement, le codage de source sert à compenser la redondance introduite par le codage de canal. L'étude présentée dans cette thèse se limite au codage de canal.

Un exemple simple de codage de canal consiste à doubler l'information, c'est-à-dire à envoyer deux fois les données à émettre, de sorte que si une partie des données est mal détectée dans la première séquence, elle puisse être récupérée dans la deuxième. Pour les codes utilisés en pratique, chaque donnée redondante est calculée en fonction de chacune des données de l'information utile. Cela permet, en plus d'apporter de la redondance aux informations, de les sécuriser puisqu'il est nécessaire en réception de connaître la règle pour décoder l'information reçue.

Les informations traitées au niveau du codage (ou de l'encodage) sont sous forme de symboles binaires, (0 ou 1) ou de blocs de symboles binaires formant ainsi des mots. Les paramètres caractérisant un code sont sa taille, ainsi que sa redondance. La taille correspond au nombre de bits (ou de mots) émis, y compris ceux de redondance. La redondance est exprimée en pourcentage et correspond au nombre de bits de redondance sur le nombre de bits utiles émis. Un code C de taille N ayant $N - K$ bits (ou mots) de redondance est noté $C(N, K)$. La performance des codes en terme de correction d'erreurs dépend de K , de la taille N du code et de la méthode de construction.

Pour une méthode de construction donnée et une longueur N fixée, plus il y a de redondance, plus le pouvoir de correction est élevé. Cependant une redondance élevée entraîne une diminution du débit utile. Ainsi, il y a nécessairement un compromis entre

pouvoir de correction et débit utile.

Les codes correcteurs sont classifiés en différentes familles ou classes de codes, en fonction de la manière dont la redondance est calculée et insérée dans les données utiles. On distingue deux catégories, les codes en blocs et les codes convolutifs dont les principales caractéristiques sont présentées dans le paragraphe suivant. L'application du codage de canal aux transmissions optiques est ensuite développée. Dans cette thèse, deux types de codes en blocs sont étudiés plus en détails, les codes de Reed Solomon (RS) [1] et les codes Low Density Parity Check (LDPC) [2].

1.2 Les différents schémas de codage correcteur d'erreurs

1.2.1 Définitions générales

1.2.1.1 Vocabulaire

Le rendement R d'un code $C(N, K)$ qui transforme un mot de K bits en un mot de N bits en ajoutant $N - K$ bits de redondance est défini par $R = K/N$.

La redondance du code, généralement notée r est donnée par :

$$r = \frac{N - K}{K} \approx \frac{N - K}{N} = 1 - R.$$

Tout mot de N bits obtenu en sortie du codeur à partir d'un mot de K bits s'appelle un mot de code. Si un mot reçu n'est pas un mot de code, c'est qu'il y a eu une erreur de transmission qui nécessite d'être corrigée. Cependant, si le mot reçu est un mot de code, il peut toutefois ne pas correspondre au mot émis, et donc comporter des erreurs, qui ne

sont pas détectées.

La distance de Hamming d_H entre deux mots de code est définie comme le nombre de symboles différents entre ces deux mots. Par exemple, les mots de symboles binaires suivants : 10001001 et 10110001 diffèrent de 3 bits. La distance de Hamming est donc $d_H = 3$. La distance de Hamming minimale d_{Hmin} est définie comme la plus petite des distances de Hamming entre deux mots de code distincts [3].

Le pouvoir de détection du code, correspondant au nombre maximal d'erreurs détectables dans un mot dépend directement de la distance de Hamming minimale. Il est déterminé par $t = d_{Hmin} - 1$.

Le pouvoir de correction, défini comme le nombre maximal d'erreurs corrigibles dans un mot est donné par [3] :

$$\left\lfloor \frac{t}{2} \right\rfloor = \left\lfloor \frac{d_{Hmin} - 1}{2} \right\rfloor \quad (1.1)$$

où $\lfloor x \rfloor$ représente la partie entière par valeur inférieure de la quantité x .

1.2.1.2 Représentations polynomiales et matricielles

Pour illustrer le principe de codage de l'information, deux représentations sont généralement utilisées : la représentation polynomiale et la représentation matricielle.

La représentation polynomiale consiste à écrire chaque mot non codé de taille K comme un polynôme $U(X)$ de degré $K - 1$ en X :

$$U(X) = \sum_{i=0}^{K-1} u_i X^i \quad (1.2)$$

où les u_i correspondent aux valeurs des différents symboles composant le mot $U = [u_0 u_1 \dots u_{K-1}]$.

Dans cette représentation, on définit de la même manière un polynôme générateur

1.2 Les différents schémas de codage correcteur d'erreurs

$G(X)$ de degré $N - K$ tel que chaque mot de code $C(X)$ puisse s'écrire comme le produit de $G(X)$ avec n'importe quel mot non codé $U(X)$. La multiplication de ce polynôme par $G(X)$ fournit le mot de code correspondant :

$$C(X) = U(X) \cdot G(X) = \left(\sum_{j=0}^{K-1} u_j X^j \right) \cdot \left(\sum_{i=0}^{N-K} g_i X^i \right) \quad (1.3)$$

Dans cette notation, les g_i sont les coefficients associés au polynôme générateur $G(X)$. Le mot en sortie du codeur C , de taille N est équivalent à un polynôme de degré $N - 1$ qui s'écrit d'une façon générale de la manière suivante :

$$C(X) = \sum_{i=0}^{N-1} c_i X^i \quad (1.4)$$

Les coefficients c_i de ce polynôme sont obtenus en développant le produit des sommes dans (1.3). Ils s'écrivent alors comme une combinaison linéaire des symboles d'information du mot en entrée du codeur. De plus, suivant la nature du code utilisé, les coefficients du polynôme générateur peuvent être eux-mêmes des combinaisons linéaires des mots d'information précédant le mot à coder. Ainsi, le mot codé s'écrit comme une combinaison linéaire des symboles d'informations du mot à coder et/ou des mots précédemment codés. Notons que dans la représentation polynomiale, toutes les opérations sont faites modulo $X^N - 1$ si les mots de codes sont sur N bits.

Dans certains cas, on préférera la notation matricielle où les mots à coder sont représentés sous forme de vecteurs de taille K et les mots codés sous forme de vecteurs de taille N . L'opération de codage s'effectue en multipliant le vecteur à coder par la matrice d'encodage G de taille $K \times N$. Si on note par exemple un mot à coder $U = [u_0 u_1 \dots u_{K-1}]_K$,

le mot codé est obtenu par la multiplication matricielle suivante :

$$C = [c_0c_1\dots c_{N-1}]_N = [u_0u_1\dots u_{K-1}]_K \cdot G_{K \times N} \quad (1.5)$$

Soient g_{ij} pour $i, j \in [0, K - 1], [0, N - 1]$ les éléments de la matrice G , chaque c_j s'écrit alors comme combinaison linéaire des u_i :

$$c_j = \sum_{i=0}^{K-1} u_i \cdot g_{ij} \quad (1.6)$$

Cette représentation est identique à la représentation polynomiale, mais est plus appropriée dans certains schémas de codage.

1.2.1.3 Codes convolutifs et codes en blocs

De part la manière dont les informations sont codées au niveau de l'émetteur, deux grandes familles de codes correcteur d'erreurs sont à distinguer : les codes convolutifs et les codes en blocs. Quelle que soit la nature du code, il est dit systématique si on retrouve dans le mot codé C les K bits d'information correspondant au mot non codé U . Cela se traduit dans la représentation matricielle par le fait qu'une partie de la matrice génératrice est composée de la matrice identité. Cette représentation est appelée la forme systématique de la matrice génératrice.

Les paragraphes suivants présentent les caractéristiques du codage et du décodage de ces deux principales familles de codes qui sont les codes convolutifs et les codes en blocs.

1.2.2 Codage

1.2.2.1 Codes convolutifs

Les codes convolutifs, aussi appelés codes convolutionnels ont été initialement inventés en 1954 par Elias [4]. Le principe des codes convolutifs est de considérer le message à émettre comme une séquence semi-infinie de symboles. Le codage convolutif consiste à faire passer le mot à coder dans une succession de registres à décalage, dont le nombre m est appelé mémoire du code. Pour construire un bloc de N symboles en sortie du codeur à partir de K symboles d'entrée, cela signifie qu'il faut m blocs de K symboles en plus du bloc de symboles utiles. On dit aussi que la longueur de contrainte du code est de $m + 1$. Un code convolutif est alors déterminé par trois paramètres : K le nombre de symboles d'information, N le nombre de symboles associés aux K symboles d'information, et m la mémoire du code.

Un exemple de codeur convolutif de rendement $R = 1/2$ transformant un mot $U = [u_0 u_1 \dots u_{K-1}]$ en un mot $C = [c_0 c_1 \dots c_{N-1}] = [c_0^1 c_0^2 c_1^1 c_1^2 \dots c_{K-1}^1 c_{K-1}^2]$ est représenté sur la figure 1.1. La valeur u_i correspond au bit qui parvient à l'entrée du codeur à l'instant i . Les bits de sortie sont alors déterminés par : $c_i^1 = u_i + u_{i-1} + u_{i-2}$ et $c_i^2 = u_i + u_{i-2}$ (additions modulo 2). Supposons que les deux registres soient initialement à 0 et que le

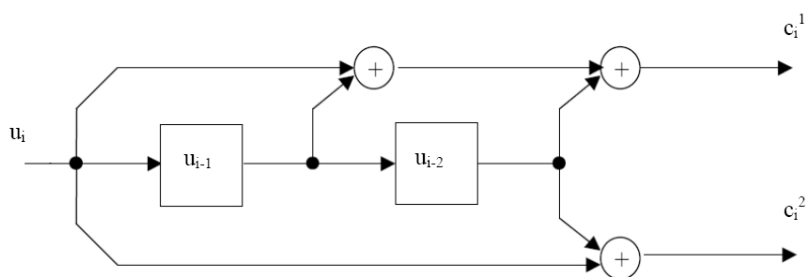


FIG. 1.1 – Exemple de codeur convolutif

codeur reçoit le message $U = [1011]$. La figure 1.2 illustre le fait que la séquence codée

devienne $C = [11100001]$, et que les registres à la fin soient à l'état $11 (= u_3u_2)$.

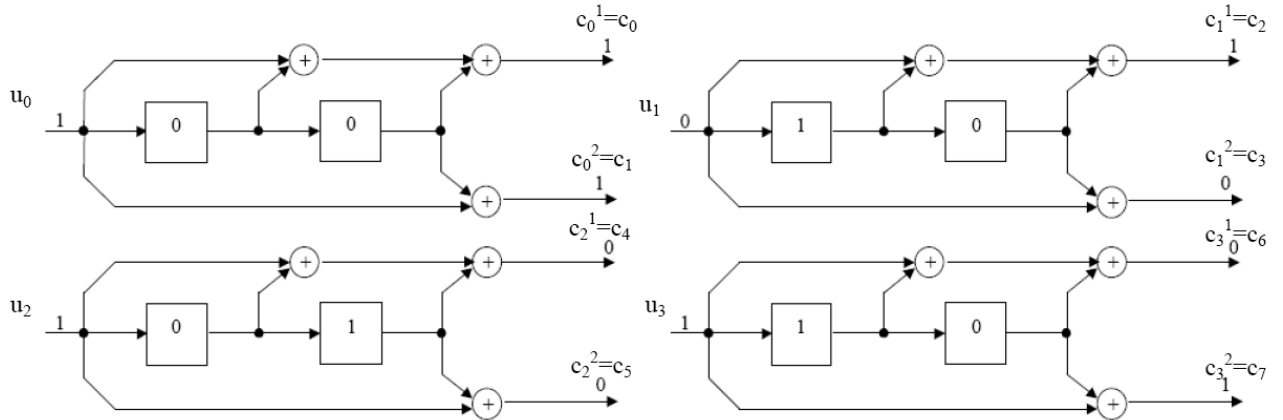


FIG. 1.2 – Exemple de codage d'information à travers un codeur convolutif

Parmi les codes convolutifs, on distingue deux grandes catégories : les codes non systématiques (ou NSC : Non Systematic Convolutional codes) et les codes systématiques récursifs (ou RSC : Recursive Systematic Convolutional codes). Un code convolutif est dit systématique si l'un des bits de sortie est identique au bit d'entrée. En effet dans ce cas, les K bits d'information en entrée du codeur se retrouvent parmi les N bits du mot codé. Un mot de code en sortie d'un codeur non-systématique possède donc plus d'informations qu'en sortie d'un codeur systématique et conduit à de meilleures performances pour les mêmes paramètres caractéristiques N , K et m . Un code convolutif est dit récursif si la séquence passant dans les registres à décalage est alimentée par le contenu de ces registres comme sur l'exemple de la figure 1.3.

L'intérêt à utiliser la récursivité est qu'en pratique, les performances obtenues à faible rapport signal à bruit sont meilleures que celles obtenues avec un code non récursif de taille identique [5]. C'est à partir de ces codes convolutifs récursifs qu'ont été inventés les turbo-codes basés sur les mêmes considérations [5]. Les codes convolutifs sont d'autant plus performants que la mémoire du code m est élevée. La contrepartie est un délai

1.2 Les différents schémas de codage correcteur d'erreurs

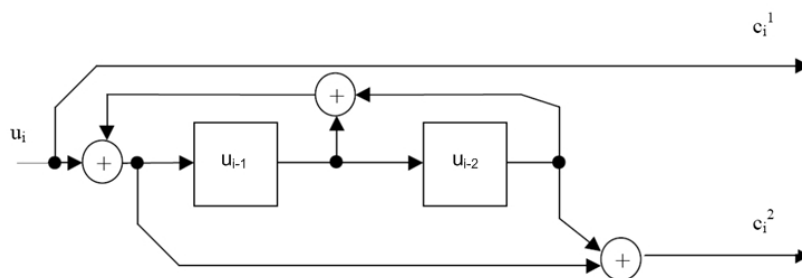


FIG. 1.3 – Exemple de codeur convolutif récursif

d'autant plus important que la mémoire du code est longue. L'utilisation de codes en blocs permet de contourner ce problème de délai.

1.2.2.2 Codes en blocs

Le codage en bloc consiste à découper les informations à transmettre en blocs de K bits. Si on considère des codes binaires, on a 2^K mots différents. A chacun de ces 2^K mots non codés est associé un mot de code unique de N bits. Ainsi, parmi les 2^N mots possibles de N bits, seuls 2^K sont des mots de codes tandis que le nombre de mots n'appartenant pas au code est plus important (égal à $2^N - 2^K$).

La plupart des codes correcteurs d'erreurs utilisés sont des codes linéaires. Ils sont caractérisés par le fait que toute combinaison linéaire de mots de code donne un mot de code.

Une caractéristique fondamentale des codes linéaires est que tout code linéaire admet une représentation systématique et est donc équivalent à un code systématique. Un code linéaire systématique est aussi appelé un code polynomial. En effet, dans ce cas, les coefficients du polynôme générateur sont simples (ils ne dépendent pas des symboles précédemment codés), et la représentation polynomiale est la plus appropriée.

Pour passer à la représentation matricielle, on utilise directement le polynôme générateur

qui permet de construire les lignes de la matrice génératrice.

Parmi les codes linéaires, un code est dit cyclique si toute permutation circulaire d'un mot de code est encore un mot de code.

Parmi les codes cycliques les plus connus, on peut citer les codes de Hamming [6], les codes CRC (Cyclic Redundancy Check) [7], les codes BCH (Bose-Chaudhuri-Hocquenghem) [8, 9] et les codes RS (Reed Solomon) [1].

Les codes LDPC (Low Density Parity Check) font partie des codes en blocs linéaires mais ne sont a priori pas cycliques.

1.2.2.3 Conclusions

Suivant la nature du code utilisé nous avons vu que les caractéristiques du codeur au niveau de l'émission sont différentes. Le codage réalisé sur les codes convolutifs met en oeuvre une succession de registres à décalage, qui impose un effet de mémoire et de latence lors du codage des informations. A l'opposé, le codage en bloc réalise l'opération de codage en une fois par multiplication avec le polynôme ou la matrice génératrice du code. L'émission est dans ce cas cadencée en fonction de la taille des blocs.

Au niveau du récepteur, un décodeur adapté au codeur d'émission est nécessaire pour prendre une décision correcte sur les données utiles reçues à partir de l'ensemble des données détectées. Les informations redondantes ajoutées à l'émission sont ainsi utilisées pour détecter et/ou corriger les erreurs. Ainsi, suivant la nature du code utilisé (convolutif ou en bloc), le décodage est basé sur des principes différents. De plus, les données reçues ayant transité le long de la chaîne de transmission ne sont plus des informations binaires (1, 0) mais des données à valeurs réelles plus ou moins proches de la valeur d'émission en fonction du niveau de bruit ou d'interférences. On distingue alors deux manières pour effectuer le décodage : soit de manière souple (soft decoding) en utilisant directement

1.2 Les différents schémas de codage correcteur d'erreurs

les valeurs incidentes, soit de manière dure (hard decoding) en utilisant les valeurs après une prise de décision à seuil. Les principes les plus couramment utilisés au niveau du décodage (effectué de manière souple ou dure en fonction du code) sont présentés dans les paragraphes suivants.

1.2.3 Décodage

1.2.3.1 Décodage des codes convolutifs

Le diagramme en treillis, qui peut aussi être utilisé pour représenter le codeur convolutif est la base du décodage [10]. Il permet de faire apparaître l'évolution des mots en sortie du codeur en fonction du temps. Un exemple de représentation en treillis est donné figure 1.4. La représentation en treillis est la plus utilisée car elle tient compte du temps et du fait que le fonctionnement du codeur est périodique, c'est-à-dire que quel que soit l'état initial du codeur, le motif se répète après $m + 1$ décalages. Dans cette

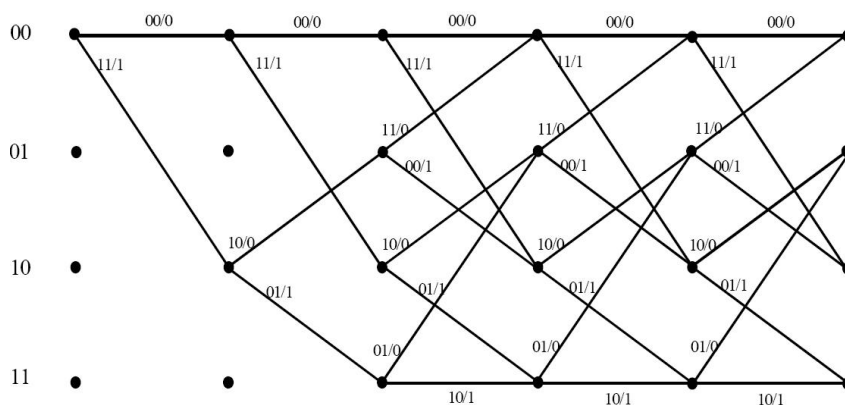


FIG. 1.4 – Exemple de diagramme en treillis, pour un code $R=1/2$

représentation, les lignes correspondent aux différents états dans lequel le codeur peut se trouver, c'est-à-dire aux valeurs présentes dans les registres à décalage. Les branches symbolisent la transition d'un état à un autre suivant qu'un '1' ou un '0' est présent en

entrée du codeur. Les valeurs reportées sur chaque branche correspondent aux valeurs des symboles c_i^1 et c_i^2 présents en sortie du codeur à l'instant i . La succession de branches crée un chemin dans le treillis, associé à une séquence en sortie du codeur. Toutes ces séquences correspondent aux différents mots de code. Si la séquence reçue par le décodeur correspond à un chemin du treillis, le mot reçu est un mot de code et aucune erreur n'est détectée. Il se peut malgré tout que la séquence reçue soit erronée mais soit quand même un mot de code si plusieurs erreurs sont présentes. Dans ce cas, les erreurs ne sont pas détectées. Si par contre, la séquence reçue ne correspond à aucun chemin du treillis, une erreur est détectée. La séquence décidée est alors la séquence la plus probable. Parmi les algorithmes basés sur le diagramme en treillis, l'algorithme de Viterbi est le plus utilisé [5]. Le principe est de déterminer à partir des distances de Hamming entre les différents mots de codes la probabilité de chaque séquence en sortie du diagramme en treillis. Ainsi, plus la distance de Hamming est élevée, plus la séquence la plus probable se distingue. La correction est d'autant plus efficace que la distance de Hamming minimale du code est élevée, ce qui est en accord avec la définition du pouvoir de correction du code (1.1). Cette technique de correction d'erreurs est basée sur le critère du maximum de vraisemblance (ML Maximum Likelihood [10]) et peut être améliorée en utilisant le Maximum *A Posteriori* (MAP) [3, 10]. Le principe général est le même mais une donnée supplémentaire portant sur la probabilité a posteriori de la séquence ou du bit décidé entre en jeu dans la détermination du maximum de vraisemblance.

La complexité du décodage est alors d'autant plus importante que le nombre d'états est élevé, mais dépend aussi du nombre de branches dans le diagramme en treillis. Elle augmente donc avec la longueur des blocs à coder ainsi qu'avec la mémoire du code m .

1.2.3.2 Décodage des codes en blocs

Pour les codes en blocs, le décodage se fait généralement à partir de la matrice de parité, appelée aussi matrice de contrôle. La matrice de parité H est définie de manière conjointe à la matrice génératrice du code G aussi appelée matrice d'encodage. Elle est définie de sorte à satisfaire la relation dite de parité :

$$G \cdot H^T = 0 \tag{1.7}$$

Dans cette relation, H^T correspond à la matrice transposée de la matrice de parité H . Tout mot de code s'écrivant $C = U \times G$, l'équation (1.7) permet de détecter si le mot reçu V est un mot de code en effectuant la multiplication de ce mot par la matrice H^T . En effet, si V est un mot de code, alors il s'écrit comme $V = U_1 \times G$ et la multiplication avec la matrice H^T donne une matrice vide (que des zéros) puisque :

$$V \cdot H^T = U_1 \cdot G \cdot H^T = 0 \tag{1.8}$$

Si une erreur est présente dans le mot reçu V alors, il y a de fortes chances qu'il ne s'écrive plus comme un mot de code, et dans ce cas le résultat de $V \cdot H^T$ n'est pas nul. Une erreur est alors détectée et doit être corrigée. Il peut arriver que malgré la présence d'erreurs dans le mot reçu, ce dernier soit quand même un mot de code. Dans ce cas, tout comme pour le cas des codes convolutionnels, l'erreur ne peut pas être détectée. De la même manière que pour les codes convolutifs, on remarque que plus la distance entre deux mots de code est élevée, moins ce genre de situations apparaît et moins d'erreurs sont effectuées.

La matrice G étant de taille $K \times N$, H doit comporter N colonnes. Soit M le nombre de lignes de H , l'équation (1.7) est alors équivalente à $K \times M$ équations de parité faisant

intervenir les N éléments du mot reçu. M est défini comme le nombre de bits (ou symboles) de parité, et est en général égal au nombre d'éléments (bits ou symboles) de redondance : $M = N - K$.

Le décodage des codes en blocs est basé soit sur le calcul du syndrome du mot reçu (décodage par syndrome [10–14]), soit sur le graphe de Tanner établi à partir de la matrice de parité [15]. Le décodage de la plupart des codes cycliques est un décodage par syndrome. Parmi les algorithmes utilisés pour le décodage par syndrome les plus connus sont l'algorithme d'Euclide dont l'application au codage de canal date de 1975 [14] ainsi que l'algorithme de Berlekamp Massey [12, 13], dont les étapes sont plus particulièrement détaillées dans la partie 1.3. Dans le cas de décodages basés sur le graphe de Tanner, des échanges itératifs ont lieu entre les différents éléments du graphe et on parle de décodage à propagation de croyance (ou BP : Belief Propagation [16, 17]). Ce dernier type de décodage est généralement appliqué au décodage des codes LDPC dont la structure se prête bien à ce type d'échanges. Parmi les algorithmes de décodage basés sur la propagation de croyance, on distingue les décodages souples ou durs (soft ou hard decoding). Ces deux types de décodages sont illustrés dans la partie 1.4.

1.2.3.3 Conclusions

Suivant la famille de code envisagée, différents schémas de codage et de décodage peuvent être envisagés. Au niveau du décodage des informations, les décodeurs basés sur un algorithme souple sont en règle générale plus efficaces que ceux basés sur un décodage dur, dans la mesure où les données analysées ne sont pas déjà soumises à une première prise de décision et contiennent donc plus d'informations. Cependant, les décodages durs effectués sur les valeurs seuillées sont plus simples à implémenter. Ainsi, quelque soit le domaine d'application, un compromis apparaît entre les deux types de décodage.

De plus, en fonction de l'application pour laquelle le codage correcteur est envisagé, les contraintes ne sont pas les mêmes. Le paragraphe suivant présente les caractéristiques propres au domaine optique en terme de codage de canal ainsi que l'état de l'art des travaux effectués dans ce contexte.

1.2.4 Les codes correcteurs d'erreurs appliqués aux transmissions optiques

Les premiers codes correcteurs développés dans le domaine des communications optiques étaient basés sur les codes en blocs, plus faciles à implémenter dans les systèmes optiques. En effet, les contraintes imposées en terme de délai par les codes convolutifs sont trop importantes pour leur mise en oeuvre en optique. Parmi les codes en blocs, les codes Bose-Chaudhuri-Hocquenghem (BCH) [8,9] et les codes de Reed Solomon (RS) [1] ont été plus particulièrement envisagés pour l'optique. Bien que la théorie ait été établie dans les années 1960, les premières solutions déployées dans le cadre des communications transocéaniques datent des années 1990 [18–22].

L'utilisation de codes correcteurs d'erreurs en communications optiques est soumise à différentes contraintes. La performance d'un code est évaluée à partir de son gain de codage (appelé "Net Coding Gain" ou NCG) défini comme la différence (en dB) entre le rapport signal à bruit (S/B) nécessaire pour obtenir une performance donnée dans les cas avec et sans codage correcteur respectivement. Pour les transmissions optiques, l'ordre de grandeur du gain de codage requis est d'environ 6dB à des taux d'erreurs très faibles (typiquement 10^{-12}). Les débits en lignes escomptés sont très élevés (10 à 40Gb/s). Ainsi, la redondance du code doit être suffisamment faible de manière à ne pas trop diminuer le débit utile. Typiquement, la redondance est toujours inférieure à 25%.

Au vu de ces contraintes, le code RS(255,239) a fait l'objet de recommandations

internationales dans une grande variété de systèmes longues distances normalisés (ITU-G975 et G709) [23,24]. En effet, le code RS(255,239), caractérisé par une faible redondance (6,7%) conduit à un gain de codage d'environ 5,8 dB pour une probabilité d'erreur de 10^{-13} . Une autre raison au choix de ce code réside dans la faible complexité de son algorithme de décodage.

Cependant, pour les communications optiques de débits supérieurs à 10 Gbit/s, les signaux sont plus sensibles aux non linéarités et des schémas de codage correcteur d'erreurs plus puissants sont nécessaires pour augmenter le gain de codage.

Pour obtenir des gains de codage plus élevés, une solution classique est d'ajouter plus de redondance, ce qui diminue le débit utile. Ceci peut être effectué en concaténant plusieurs codes correcteurs, ce qui augmente la complexité [22, 25–27]. Cette solution, basée sur l'utilisation de la concaténation fait partie de la seconde génération de système correcteur d'erreurs pour les communications optiques et a fait l'objet d'une nouvelle normalisation [28]. Deux codes sont généralement utilisés en cascade et séparés par un entrelaceur dont le rôle est de disperser les erreurs susceptibles d'être non corrigées par le premier décodeur. En utilisant deux codes RS (255,239) concaténés, le gain de codage à 10^{-12} augmente ainsi jusqu'à environ 7,4 dB. Les meilleures performances en matière de concaténation correspondent à un gain de codage d'environ 9,4 dB. La figure 1.5 issue de [29] répertorie ces différentes familles de codes en fonction de leur gain de codage à 10^{-12} et de l'année où ces solutions ont été exploitées. Pour améliorer encore les performances, il faut s'intéresser à des solutions plus puissantes dites de troisième génération.

L'avènement des turbo-codes a révolutionné le domaine du codage de canal pour les communications radio-mobiles. L'utilisation de turbo-codes en blocs a donc été proposée et appliquée aux communications optiques [29]. Cependant, le décodage turbo est difficilement compatible avec la technologie optique et possède un degré de complexité

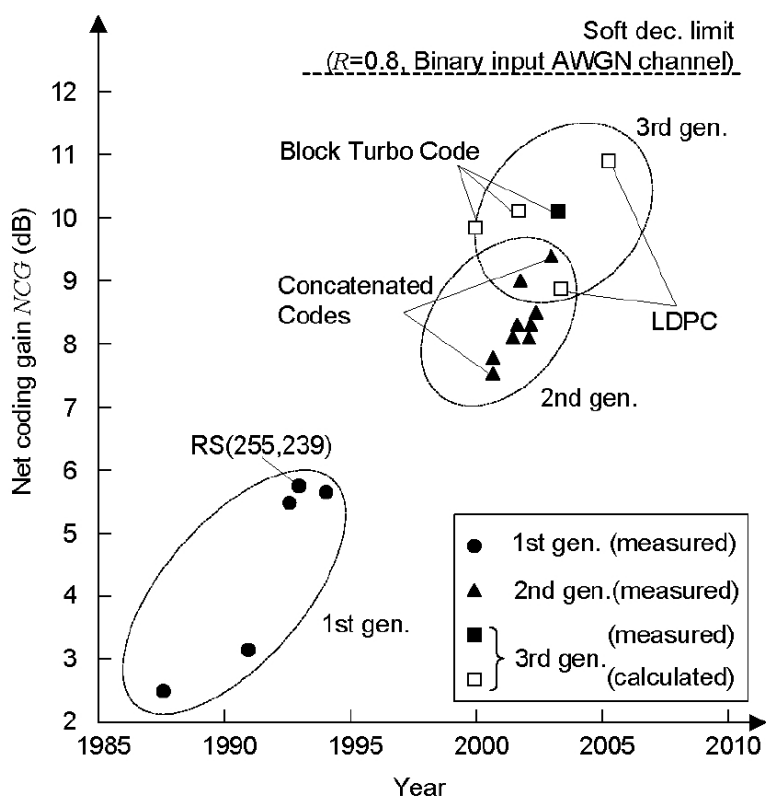


FIG. 1.5 – Evolution des gains de codage de différents codes

important pour ce type de transmission.

Bien que longtemps mis de côté, les codes LDPC (Low Density Parity Check) utilisant la théorie des graphes ont fait l'objet de récentes études pour les nouveaux schémas de communication. Celles-ci ont révélé des performances très efficaces, notamment sur canal à bruit additif gaussien. L'avantage de ces codes est d'avoir, de par leur structure très peu dense, une complexité de décodage nettement simplifiée. L'utilisation des LDPC dans le cas des communications optiques a donc été développée et a fait l'objet de nombreux travaux [30–41] menés notamment par Djordjevic et Vasic.

Ainsi, dans ce mémoire, l'étude se concentre sur ce type de codes de troisième génération, qui s'avère être pertinent pour les communications optiques. De manière

à évaluer l'intérêt des codes LDPC, plus complexes que les codes RS déjà normalisés, l'étude portera aussi sur les codes de première génération de type RS afin de comparer les performances. Les deux paragraphes suivants détaillent plus particulièrement les caractéristiques de ces deux grandes familles de codes.

1.3 Codes de Reed Solomon

1.3.1 Introduction

Les codes de Reed-Solomon (RS) [1] sont obtenus par extension des codes BCH. Ces deux familles de codes font partie des codes en blocs linéaires cycliques. Rappelons qu'un code cyclique est un code en bloc linéaire tel que chaque mot de code permuté à droite ou à gauche donne un autre mot de code. Cela signifie que si $C = [c_0c_1\dots c_{N-1}]$ appartient au code, $C' = [c_1c_2\dots c_{N-1}c_0]$ ou $C'' = [c_{N-1}c_0c_1c_2\dots c_{N-2}]$ appartiennent aussi au code. Avec la représentation polynomiale, cette propriété s'illustre par le fait que si C appartient au code alors toute permutation s'écrivant $C^{(p)}(X) = X^p \cdot C(X)$ modulo $(X^N - 1)$ avec $p \in \mathbb{N}$ appartient aussi au code.

Une propriété particulière des codes linéaires cycliques est que le polynôme générateur du code $G(X)$ peut s'écrire comme un diviseur de $X^N - 1$. En effet, le polynôme générateur du code est défini comme le polynôme de degré le plus petit qui génère l'ensemble des mots de code. Or, d'après la définition des codes linéaires, le mot '0' doit appartenir au code. Ceci est assuré s'il existe un polynôme $P(X)$ tel que :

$$P(X) \cdot G(X) = X^N - 1 = 0 \quad (X^N - 1) \tag{1.9}$$

avec la notation (a) désignant le fait que l'opération a lieu modulo a .

1.3 Codes de Reed Solomon

Ainsi, $G(X)$ est toujours un facteur de la décomposition de $X^N - 1$ et de degré $N - K$ de manière à transformer un mot de taille K (polynôme de degré $K - 1$) en un mot de code de taille N (polynôme de degré $N - 1$).

La différence entre les codes RS et BCH se situe dans le fait que les symboles binaires des codes BCH sont transformés en symboles m -aires (composés de m bits) pour obtenir les codes RS. Cependant, les principes d'encodage et décodage sont communs entre les deux classes de codes et les codes BCH et RS sont tous deux basés sur la théorie des corps de Galois qui définit des concepts communs. Le paragraphe suivant rappelle les notions utiles sur les corps de Galois pour la définition des paramètres des codes RS et BCH. Les codes BCH et RS sont définis plus en détail dans les autres paragraphes. Un paragraphe dédié au décodage, dont le principe est commun entre les deux classes de codes est ensuite présenté avant de conclure sur l'application de ces codes à l'optique.

1.3.2 Notions sur les corps de Galois

Un corps de Galois noté $\text{GF}(Q)$ (pour Galois Field) est un ensemble fini qui possède Q éléments. Un corps de Galois défini à partir d'un élément primitif est tel que le nombre d'éléments Q du corps de Galois s'écrit $Q = q^m$ où q est un nombre premier. La définition de $\text{GF}(q^m)$ repose sur un polynôme irréductible $p(x)$ de degré m , à coefficients dans $\text{GF}(q)$. Comme $p(x)$ est irréductible, il ne possède pas de racine dans $\text{GF}(q)$. Cependant, si on suppose que ce polynôme possède m racines dans un autre ensemble et qu'on note α l'une de ces racines, cette racine est alors définie comme l'élément primitif du corps de Galois $\text{GF}(q^m)$. Le polynôme $p(x)$ dont α est une racine est appelé le polynôme primitif de $\text{GF}(q^m)$ puisqu'il est essentiel à l'obtention des éléments du corps. En effet à partir de

ce polynôme et de α , tous les éléments du corps de Galois sont obtenus :

$$GF(q^m) = \{0, \alpha^0, \alpha, \dots, \alpha^{q^m-2}\}. \quad (1.10)$$

Les opérations de multiplication et d'addition dans $GF(q^m)$, sont réalisées comme suit :

- pour l'addition, on réalise la somme bit par bit modulo q
- pour la multiplication, on utilise la notation en puissances de α c'est-à-dire pour tout i, j entiers : $\alpha^i \cdot \alpha^j = \alpha^{(i+j)(q^m-1)}$.

La définition des codes correcteurs d'erreurs BCH ou RS repose sur des corps de Galois définis à partir d'un élément primitif avec $q = 2$ et tel que $m \geq 3$. Le corps de Galois s'écrit $GF(2^m)$ et contient 2^m éléments. Les polynômes primitifs des codes BCH les plus courants sont répertoriés dans des tables, comme dans le tableau 1.1.

m	$P(X)$	m	$P(X)$
3	$1 + X + X^3$	14	$1 + X + X^6 + X^{10} + X^{14}$
4	$1 + X + X^4$	15	$1 + X + X^{15}$
5	$1 + X^2 + X^5$	16	$1 + X + X^3 + X^{12} + X^{16}$
6	$1 + X + X^6$	17	$1 + X^3 + X^{17}$
7	$1 + X^3 + X^7$	18	$1 + X^7 + X^{18}$
8	$1 + X^2 + X^3 + X^4 + X^8$	19	$1 + X + X^2 + X^5 + X^{19}$
9	$1 + X^4 + X^9$	20	$1 + X^3 + X^{20}$
10	$1 + X^3 + X^{10}$	21	$1 + X^2 + X^{21}$
11	$1 + X^2 + X^{11}$	22	$1 + X + X^{22}$
12	$1 + X + X^4 + X^6 + X^{12}$	23	$1 + X^5 + X^{23}$
13	$1 + X + X^3 + X^4 + X^{13}$	24	$1 + X + X^2 + X^7 + X^{24}$

TAB. 1.1 – Polynômes primitifs les plus courants

Les additions dans le corps de Galois ont donc lieu bit à bit modulo (2), et ainsi l'opérateur $+$ est égal à l'opérateur $-$. De plus, les 2^m éléments du corps, appelés symboles sont composés des éléments de base '0' et '1'. Pour en avoir 2^m il suffit de construire des symboles de m bits. Ces symboles m -aires sont utiles pour la définition des polynômes

générateurs des codes BCH ou RS. En effet, ils sont construits comme étant des polynômes de degré $N - K$ et possédant comme racines les $2t$ puissances de l'élément primitif α : $\{\alpha^i\}$, $i \in [1, 2t]$ où t correspond au nombre maximal d'erreurs corrigibles par le code considéré. De part cette propriété, ces symboles m -aires sont manipulés lors du décodage des données pour localiser et évaluer les erreurs de transmission tant pour les codes BCH que pour les codes RS.

1.3.3 Les codes BCH

1.3.3.1 Paramètres caractéristiques

Les codes BCH sont des codes binaires dont les symboles sont par conséquent composés d'un seul bit '1' ou '0'. Les paramètres caractéristiques d'un code BCH défini à partir de $\text{GF}(2^m)$ où m est un entier quelconque sont :

- la taille du code $N = 2^m - 1$, correspondant au nombre de bits par mot codé
- le nombre de bits utiles par mot codé : K
- le nombre maximum d'erreurs corrigibles dans un mot t , tel que $\frac{N-K}{m} \leq t \leq \frac{N}{2}$

Ces caractéristiques correspondent à la définition de base des codes BCH les plus usuels. Une extension de cette définition permet d'obtenir des codes BCH ne possédant pas la contrainte $N = 2^m - 1$. Dans ce cas, pour une dimension de code N donnée, le code est obtenu à partir de $\text{GF}(2^{m'})$ où $m' \in \mathbb{N}$ définit $N' = 2^{m'} - 1$ tel que $X^N - 1$ divise $X^{N'} - 1$.

Les caractéristiques de ces codes BCH étendus ne sont pas détaillées dans cette thèse, qui se limite à l'étude des cas $N = 2^m - 1$.

1.3.3.2 Polynômes générateurs

Les polynômes générateurs de codes BCH de paramètres $(N, K; t)$ sont généralement répertoriés en octal dans des tables. Un exemple de table est représenté figure 1.6. Prenons par exemple le premier code BCH($N=7, K=4; t=1$). Le polynôme générateur est 13 en octal c'est-à-dire $(13)_8 = (1011)_2$ d'où l'écriture polynomiale du polynôme générateur du code : $G(X) = 1 + X + X^3$.

Rappelons que par construction, les $\alpha^i, i \in [1, 2t]$, puissances de l'élément primitif de $\text{GF}(2^m)$ sont des racines du polynôme générateur de degré $N - K$.

1.3.3.3 Codage

L'encodage d'un mot composé de K symboles binaires s'effectue en multipliant le mot à coder écrit sous forme polynomiale par le polynôme générateur. Le polynôme obtenu correspond alors au mot de code correspondant.

Reprenons l'exemple du code BCH($N=7, K=4; t=1$) dont le polynôme générateur est $G(X) = 1 + X + X^3$ et le mot à coder $U = [1010]$. Ce mot U s'écrit sous la forme $U(X) = 1 + X^2$ en représentation polynomiale. En multipliant le mot à coder par le polynôme générateur, on obtient :

$$C(X) = (1 + X^2) \cdot (1 + X + X^3) = 1 + X + X^3 + X^2 + X^3 + X^5 = 1 + X + X^2 + X^5 \quad (1.11)$$

Cette représentation polynomiale correspond à $C = [1110010]$ en puissances de X croissantes. Le tableau 1.2 représente l'ensemble des mots de codes obtenus à partir du code BCH($N=7, K=4; t=1$), la ligne grisée correspondant à l'exemple présenté.

1.3 Codes de Reed Solomon

TABLE 6.4 Generators of Primitive BCH Codes

n	k	t	$g(x)$	n	k	t	$g(x)$
7	4	1	13	255	171	11	15416214212342356077061630637
15	11	1	23	163	12	12	7500415510075602551574724514601
	7	2	721	155	13	13	375751300540766501572250646677633
	5	3	2467	147	14	14	1642130173537165525304165305441011711
31	26	1	45	139	15	15	461401732060175561570722730247453567445
	21	2	3551	131	18	18	2157133314715101512612502774421420241
	16	3	107657				65471
	11	5	5423325	123	19	19	12061450522420660037172103265161412262
	6	7	313365047				72506267
63	57	1	103	115	21	21	6052666557210024726363640460027635255
	51	2	12471				6313472737
	45	3	1701317	107	22	22	2220577232206625631241730023534742017
	39	4	166623567				6574750154441
	36	5	1033500423	99	23	23	1065666725347317422274141620157433225
	30	6	157464165547				2411076432303431
	24	7	17323260404441	91	25	25	6750265030327444172723631724732511075
	18	10	1363026512351725				550762720724344561
	16	11	6331141367235453	87	26	26	1101367634147432364352316343071720462
	10	13	472622305527250155				06722545273311721317
	7	15	5231045543503271737	79	27	27	6670003563765750002027034420736617462
							1015326711766541342355
127	120	1	211	71	29	29	2402471052064432151555417211233116320
	113	2	41567				5444250362557643221706035
	106	3	11554743	63	30	30	1075447505516354432531521735770700366
	99	4	3447023271				6111726455267613656702543301
	92	5	624730022327	55	31	31	7315425203501100133015275306032054325
	85	6	130704476322273				414326755010557044426035473617
	78	7	26230002166130115	47	42	42	2533542017062646563033041377406233175
	71	9	6255010713253127753				123334145446045005066024552543173
	64	10	1206534025570773100045	45	43	43	1520205605523416113110134637642370156
	57	11	335265252505705053517721				3670024470762373033202157025051541
	50	13	54446512523314012421501421	37	45	45	5136330255067007414177447245437530420
	43	14	17721772213651227521220574343				735706174323432347644354737403044003
	36	15	3146074666522075044764574721735	29	47	47	3025715536673071465527064012361377115
	29	21	403114461367670603667530141176155				34224232420117411406025475741040356
	22	23	123376070404722522435445626637647043				5037
	15	27	22057042445604554770523013762217604353	21	55	55	1256215257060332656001773153607612103
	8	31	7047264052751030651476224271567733130217				22734140565307454252115312161446651
255	247	1	435				3473725
	239	2	267543	13	59	59	464173200505256454426573714250066004
	231	3	156720665				33067744547656140317467721357026134
	223	4	75626641375				460500547
	215	5	23157564726421	9	63	63	1572602521747246320103104325535513461
	207	6	16176560567636227				41623672120440745451127661155477055
	199	7	7633031270420722341				61677516057
	191	8	2663470176115333714567				
	187	9	52755313540001322236351				
	179	10	22624710717340432416300455				

Source: Reprinted with permission from "Table of Generators for BCH Codes," *IEEE Trans. Inf. Theory*, vol. IT10, no. 4, Oct. 1964, p. 391. © 1964 IEEE.

FIG. 1.6 – Table des polynômes générateurs des codes BCH

Mot à coder U	$U(x)$	$C(x)$	Mot codé C
0000	0	0	0000000
1000	1	$1 + X + X^3$	1101000
0100	X	$X + X^2 + X^4$	0110100
1100	$1 + X$	$1 + X^2 + X^3 + X^4$	1011100
0010	X^2	$X^2 + X^3 + X^5$	0011010
1010	$1 + X^2$	$1 + X + X^2 + X^5$	1110010
0110	$X + X^2$	$X + X^3 + X^4 + X^5$	0101110
1110	$1 + X + X^2$	$1 + X^4 + X^5$	1000110
0001	X^3	$X^3 + X^4 + X^6$	0001101
1001	$1 + X^3$	$1 + X + X^4 + X^6$	1100101
0101	$X + X^3$	$X + X^2 + X^3 + X^6$	0111001
1101	$1 + X + X^3$	$1 + X^2 + X^6$	1010001
0011	$X^2 + X^3$	$X^2 + X^4 + X^5 + X^6$	0010111
1011	$1 + X^2 + X^3$	$1 + X + X^2 + X^3 + X^4 + X^5 + X^6$	1111111
0111	$X + X^2 + X^3$	$X + X^5 + X^6$	0100011
1111	$1 + X + X^2 + X^3$	$1 + X^3 + X^5 + X^6$	1001011

TAB. 1.2 – Exemple : mots de code du BCH(7,4;1)

1.3.4 Les codes de Reed Solomon

1.3.4.1 Paramètres caractéristiques

Les codes de Reed Solomon sont obtenus par extension des codes BCH aux codes non binaires. Le principe est identique à la différence que les symboles composant le mot à coder ne sont plus binaires ('0' ou '1') mais des symboles composés de m bits, où m définit le corps de Galois $\text{GF}(2^m)$ et fixe la taille du code N de la même manière que pour les codes BCH.

Les paramètres caractéristiques d'un code $\text{RS}(N, K)$ sont :

- la taille du code $N = 2^m - 1$, correspondant au nombre de symboles par mot codé (soit $N \times m$ bits dans le mot codé)
- le nombre de symboles utiles par mot codé : K

Le nombre maximum d'erreurs corrigibles t (en symboles) dans un mot reçu est fixé par

1.3 Codes de Reed Solomon

ces paramètres et donné par la relation $2t = N - K$.

m est aussi appelé la classe du code.

1.3.4.2 Polynômes générateurs

Chaque code $RS(N, K)$ est défini par un polynôme générateur. Tout comme les codes BCH, le polynôme générateur de degré $N - K$ possède $2t$ racines qui sont des puissances de l'élément primitif α du corps de Galois $GF(2^m)$. Cependant dans le cas des codes RS, $N - K = 2t$, donc le polynôme générateur est entièrement défini à partir des puissances de α et n'est généralement pas repertorié dans des tables. Il est obtenu comme :

$$G(X) = (X - \alpha)(X - \alpha^2)\dots(X - \alpha^{2t}) \quad (1.12)$$

La représentation en puissance de α pour les 2^m éléments du corps de Galois $GF(2^m)$ est équivalente à la représentation de ces éléments comme symboles de m bits. Tandis que cette représentation est utilisée pour les codes BCH uniquement pour le décodage, elle sert pour les codes RS dès la définition de $G(X)$ donc dès l'encodage. Le détail de cette représentation ainsi que de l'obtention de $G(X)$, est donné dans le cas particulier $RS(255,239)$ en annexe A.

1.3.4.3 Codage

Pour effectuer l'encodage des messages avec les codes RS, on peut procéder de la même manière que pour les codes BCH c'est-à-dire en développant les termes dans la multiplication polynomiale $U(X) \cdot G(X)$. Cependant, comme les coefficients du polynôme générateur ainsi que ceux du mot à coder ne sont plus des bits mais des symboles de m bits (ou des puissances de α), la complexité de l'encodage augmente. Une manière plus

simple, utilisant les propriétés des codes cycliques est alors mise en place pour l'encodage.

Pour obtenir le mot codé, il suffit en effet de :

- Multiplier le mot à coder $U(X)$ par X^{N-K} pour obtenir $A(X)$
- Diviser $A(X)$ par le polynôme $G(X)$ pour en déterminer le reste $R(X)$
- Additionner $A(X)$ et $R(X)$ pour obtenir le mot codé $C(X)$

Le mot ainsi obtenu $C(X) = A(X) + R(X)$ est bien égal à $C(X) = U(X) \cdot G(X)$ de part la propriété des codes linéaires. En effet, tout code linéaire étant équivalent à un code systématique, tout mot codé $C(X) = U(X) \cdot G(X)$ peut aussi s'écrire :

$$C(X) = U(X) \cdot X^{N-K} + b(X) \tag{1.13}$$

Ainsi, on a l'égalité :

$$U(X) \cdot G(X) = U(X) \cdot X^{N-K} + b(X) \tag{1.14}$$

En divisant (1.14) par $G(X)$ et sachant que $+ = -$ dans le corps $\text{GF}(2^m)$, on obtient :

$$\frac{U(X) \cdot X^{N-K}}{G(X)} = U(X) + \frac{b(X)}{G(X)} \tag{1.15}$$

Ainsi, $b(X)$ s'écrit bien comme le reste de la division de $U(X) \cdot X^{N-K}$ par $G(X)$, et à partir de (1.14), le mot codé $C(X) = U(X) \cdot G(X)$ comme la somme de $U(X) \cdot X^{N-K}$ et du reste $b(X)$. Le schéma de codage est alors simplifié puisqu'il suffit de calculer ce reste. Ce calcul peut être réalisé à l'aide de bascules et de registres à décalage. Le schéma de codage est alors équivalent à la figure 1.7 avec en entrée "in" le polynôme $U(X) \cdot X^{N-K}$.

Sur ce schéma, les b_j sont les valeurs prises par les différentes bascules du registre à décalage. Elles évoluent en fonction des valeurs d'entrées "in" et de la position ouverte

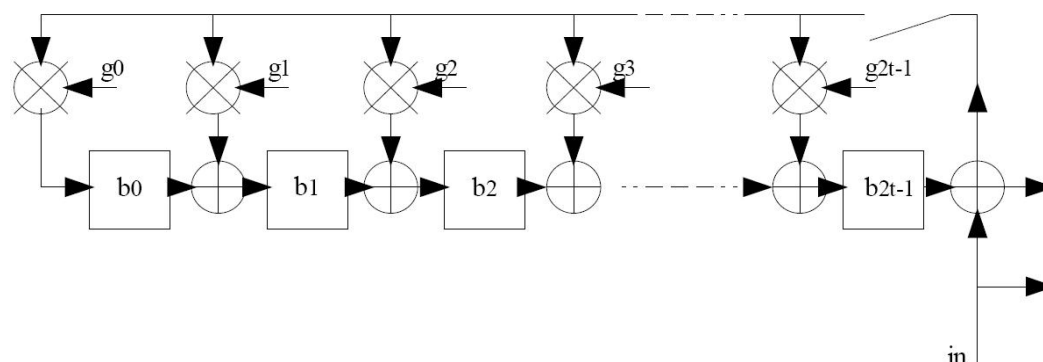


FIG. 1.7 – Représentation équivalente du codeur cyclique utilisée pour le codeur RS

ou fermée de l'interrupteur. Les g_j sont les coefficients du polynôme $G(X)$, quotient de la division. A la fin, les valeurs prises par les b_j correspondent aux coefficients du reste de la division de l'entrée par $G(X)$. Elles sont récupérées en sortie par un simple décalage avec l'interrupteur en position fermée.

1.3.5 Décodage des codes RS et BCH

1.3.5.1 Principe général

Le principe général du décodage est le même pour les codes RS et les codes BCH. Il s'agit tout d'abord de détecter la ou les erreurs de transmission à l'aide de l'évaluation des syndromes. Ensuite, les erreurs sont localisées, évaluées puis corrigées. Dans le cas des codes binaires tels que les BCH, l'évaluation et la correction des erreurs est immédiate (un '0' doit être modifié en un '1' et inversement). Cependant, pour le décodage des codes RS, une évaluation de l'erreur doit avoir lieu.

En utilisant la représentation polynomiale, chacune de ces étapes est détaillée ci-après.

1.3.5.2 Détection d'erreurs

Soit V le mot reçu, celui-ci peut s'écrire à partir du mot codé $C(X)$:

$$V(X) = C(X) + E(X) \tag{1.16}$$

en notant $E(X)$ le polynôme d'erreurs introduites par le canal et s'écrivant d'une manière générale comme :

$$E(X) = e_0 + e_1X + \dots e_iX^i + \dots + e_{N-1}X^{N-1} \tag{1.17}$$

avec $e_i \in GF(2^m)$ pour tout $i \in [0, N - 1]$.

Lorsque $e_i \neq 0$, cela signifie qu'une erreur de transmission se trouve sur le symbole en position i .

Si on estime le polynôme du mot reçu $V(X)$ aux valeurs α^i , $i \in [1, 2t]$, on a :

$$V(\alpha^i) = C(\alpha^i) + E(\alpha^i) = E(\alpha^i) \tag{1.18}$$

En effet, rappelons que pour tout code BCH ou RS pouvant corriger jusqu'à t erreurs, les α^i , $i \in [1, 2t]$ sont des racines du polynôme générateur $G(X)$ c'est-à-dire $G(\alpha^i) = 0$. Ainsi, $C(\alpha^i) = U(\alpha^i) \cdot G(\alpha^i) = 0$. Le décodage est alors basé sur le syndrome calculé à partir du mot reçu et défini par un vecteur à $2t$ composantes :

$$S = [S_1 S_2 \dots S_{2t}] \tag{1.19}$$

avec $S_i = V(\alpha^i) = E(\alpha^i)$, $i \in [1, 2t]$. Lorsque S est égal au vecteur nul, il n'y a pas d'erreur. Lorsque certaines composantes de S sont non nulles, il y a des erreurs de

transmission qui peuvent être éventuellement corrigées en fonction de leur nombre et du pouvoir de correction du code.

1.3.5.3 Localisation et évaluation d'erreurs

En présence de t erreurs de transmission, le polynôme d'erreurs $E(X)$ est de la forme :

$$E(X) = e_{n_1}X^{n_1} + \dots e_{n_j}X^{n_j} + \dots e_{n_t}X^{n_t} \quad (1.20)$$

où les e_{n_j} , $j \in [1, t]$ sont des coefficients non nuls prenant leurs valeurs dans le corps $\text{GF}(2^m)$. Les composantes S_i du syndrome peuvent donc s'écrire :

$$S_i = E(\alpha^i) = e_{n_1} (\alpha^i)^{n_1} + \dots e_{n_j} (\alpha^i)^{n_j} + \dots e_{n_t} (\alpha^i)^{n_t} \quad (1.21)$$

En posant $Z_j = \alpha^{n_j}$, l'écriture de (1.21) se simplifie en :

$$S_i = e_1 Z_1^i + \dots e_j Z_j^i + \dots e_t Z_t^i \quad (1.22)$$

Pour déterminer la position des erreurs de transmission il reste à connaître les valeurs des quantités Z_j . Pour les corriger, il faut évaluer les coefficients e_j . On utilise principalement pour cela des méthodes itératives et un décodage dur. Les deux algorithmes de décodage durs les plus couramment employés sont l'algorithme de Berlekamp-Massey [12, 13] et l'algorithme d'Euclide [14].

Dans chacun de ces deux algorithmes de décodage, deux polynômes importants sont recherchés : le polynôme localisateur d'erreurs, noté $\sigma(X)$ et le polynôme évaluateur d'erreurs $\omega(X)$. Le polynôme localisateur d'erreurs est utilisé pour déterminer la position des erreurs. Les racines de ce polynôme sont par définition les quantités $\frac{1}{Z_j}$, il peut donc

s'écrire :

$$\sigma(X) = \prod_{j=1}^t (1 + X) \quad (1.23)$$

Le polynôme évaluateur d'erreurs permet de déterminer la valeur de l'erreur e_j et est défini par :

$$\omega(X) = \sum_{j=1}^t e_j Z_j \frac{\sigma(X)}{1 + Z_j X} \quad (1.24)$$

De plus, les polynômes localisateur et évaluateur d'erreurs vérifient la relation suivante (1.25) :

$$\sigma(X) \cdot S(X) = \omega(X) (X^{2t}) \quad (1.25)$$

avec $S(X)$ le polynôme associé au syndrome défini par (1.19). Cette relation (1.25) s'appelle l'équation clé du décodage d'un code cyclique.

Les polynômes $\sigma(X)$ et $\omega(X)$ sont obtenus itérativement à partir de la relation (1.25), de manières différentes par chacun des algorithmes. Les étapes des algorithmes d'Euclide et de Berlekamp-Massey sont détaillées en annexe B. Une fois les polynômes déterminés, la correction des erreurs peut avoir lieu.

1.3.5.4 Correction des erreurs

Une erreur e_j en position n_j est déterminée à l'aide des polynômes $\sigma(X)$ et $\omega(X)$. En effet, par définition du polynôme localisateur d'erreur, on a $\sigma(Z_j^{-1}) = 0$. Ainsi, si on évalue $\omega(X)$ en Z_j^{-1} , on obtient :

$$\omega(Z_j^{-1}) = e_j Z_j \left. \frac{\sigma(X)}{1 + Z_j X} \right|_{X=Z_j^{-1}} \quad (1.26)$$

1.3 Codes de Reed Solomon

L'expression (1.26) est indéterminée puisque $\sigma(Z_j^{-1}) = 0$ et le dénominateur $(1 + Z_j X)$ vaut aussi 0 en $X = Z_j^{-1}$. Pour obtenir l'expression de (1.26), il faut passer par les dérivées des numérateur et dénominateur. Par définition de la fonction dérivée, il vient que : $\frac{f'(u)}{g'(u)} = \frac{f(u)}{g(u)}$. En appliquant cette règle à (1.26) avec $f(X) = \sigma(X)$ et $g(X) = 1 + Z_j X$ on a $f'(X) = \sigma'(X)$ et $g'(X) = Z_j$, d'où :

$$\omega(Z_j^{-1}) = e_j \sigma'(Z_j^{-1}) \quad (1.27)$$

A partir de (1.27), il vient alors que :

$$e_j = \frac{\omega(Z_j^{-1})}{\sigma'(Z_j^{-1})} \quad (1.28)$$

Une fois chaque e_j déterminé, il suffit d'additionner le mot reçu $V(X)$ qui s'écrit comme $C(X) + E(X)$ avec le mot d'erreur $E(X)$ puisque dans ce cas on aura :

$$V(X) + E(X) = C(X) + E(X) + E(X) = C(X) \quad (1.29)$$

Remarque : pour le cas des codes BCH, les coefficients e_i prennent comme valeur '0' ou '1'. Ainsi, dès que l'erreur est localisée par $\sigma(X)$, elle peut être corrigée en interchangeant simplement '0' et '1'. Il n'est alors pas nécessaire de calculer le polynôme évaluateur d'erreurs.

1.3.5.5 Conclusions

Les résultats obtenus à l'aide des deux algorithmes sont identiques en terme de détection, évaluation et correction d'erreurs, puisqu'à l'issue des deux algorithmes, les polynômes obtenus sont identiques, ou équivalents [42]. La différence réside dans le

fait qu'au cours de l'algorithme de Berlekamp-Massey, on n'explicite pas les polynômes R_i obtenus dans l'algorithme d'Euclide par divisions polynomiales. Ainsi, l'obtention simultanée des polynômes σ_i et ω_i nécessite environ deux fois moins de calculs qu'avec la méthode d'Euclide [42].

Une autre méthode non présentée ici est une méthode directe (Peterson [11]) qui consiste à résoudre un système non linéaire à $2t$ équations. Cependant la complexité de cette méthode augmente avec le carré de t tandis qu'avec les méthodes itératives, elle n'augmente que linéairement avec t .

Enfin, d'autres méthodes de décodage basées sur un décodage souple ont été envisagées. L'algorithme de Chase [43] est un exemple de ce type de décodage souple, sous-optimal. D'autres types de décodages basés sur les treillis [44] ont aussi été envisagés mais deviennent vite lourds pour de longues tailles de codes en blocs et n'ont pas été utilisés dans cette étude.

1.3.6 Conclusion - Application des codes RS à l'optique

La théorie des codes RS et BCH repose sur les corps de Galois et ces deux grandes familles de codes possèdent alors des caractéristiques communes. Leurs codage et décodage s'effectuent de manière simple à l'aide d'opérations polynomiales. Ils font partie des codes correcteurs d'erreurs dits de première génération. En effet, les premiers déploiements de codage correcteur d'erreurs pour les transmissions optiques sous-marines ont utilisé ce type de codes [18–22].

La première réalisation datant de 1991, a consisté en un système de transmission fonctionnant à 565 Mb/s avec un code BCH étendu : BCH(167,151) de redondance égale à 10% [18] et pouvant corriger jusqu'à $t = 2$ erreurs par bloc. L'année suivante, une autre démonstration utilisant les codes RS et fonctionnant à 2.5 Gb/s a été réalisée [19]. Une

1.3 Codes de Reed Solomon

transmission directe (sans répéteur) de 357 km a ainsi été mise en place.

Enfin, l'introduction d'amplificateurs optiques de type EDFA (Erbium Doped Fiber Amplifier) a permis la réalisation en 1994 de systèmes de transmissions à plus longues distances (6400 km [21]) et fonctionnant à des débits plus rapides (5 Gbs/s) [20, 21].

Les normalisations ITU-G975 et G709 [23, 24] ont défini un premier standard de communication utilisant le code RS(255,239), $t = 8$ et appartenant à la classe $m = 8$ c'est-à-dire construit sur $GF(2^8)$. A partir de ce code normalisé, plusieurs réalisations de systèmes utilisant la concaténation du code RS(255,239) avec d'autres codes ont été mises en place en vue d'améliorer les performances [25–27]. C'est ainsi qu'une version modifiée ITU-G975.1 [28] de la normalisation du code RS (255,239) est apparue en 2004 définissant des standards basés sur différentes concaténations de codes.

Finalement, le code RS(255,239) reste le code de référence le plus simple pour évaluer par comparaison les performances de n'importe quel système correcteur d'erreurs. A ce titre, l'annexe A fournit en détail les caractéristiques de construction de ce code ainsi que des éléments du corps de galois $GF(2^8)$ duquel il est issu. Les gains de codage obtenus avec les codes LDPC présentés dans le paragraphe suivant seront ainsi comparés à ceux obtenus avec les codes RS. Etant donné que la complexité du système de codage / décodage est augmentée avec les codes LDPC, il est important d'évaluer l'amélioration qu'ils apportent afin de juger si elle est suffisamment significative pour privilégier ce type de code au détriment des codes RS.

1.4 Codes LDPC

1.4.1 Introduction

Les codes LDPC sont des codes correcteurs en blocs linéaires. Proposés en 1962 par Gallager [2], ces codes n'ont pas immédiatement suscité un grand intérêt auprès de la communauté de la théorie de l'information. Ce n'est que durant les années 1990 que les codes de Gallager furent redécouverts, notamment par MacKay [45] qui a remis en lumière l'intérêt des codes LDPC à l'aide des représentations et de l'approche récursive de Tanner [15].

Le principe des codes LDPC est basé sur le contrôle de parité, un des systèmes de contrôle les plus simples appelé bit de parité (ou VRC : Vertical Redundancy Check). Dans ce schéma simple de codage, un bit supplémentaire (le bit de parité) est ajouté aux K bits de données pour former le mot de code de taille $N = K + 1$. Généralement $K = 7$ bits, ce qui forme un octet avec le bit de parité. Ce bit de parité prend une valeur ('0' ou '1') telle que le nombre total de bits à '1' dans le mot de code soit pair. Ainsi, un '1' sera ajouté si le nombre de bits du mot à coder est impair, un '0' dans le cas contraire. Ce principe de base est étendu dans le cas des codes LDPC où plusieurs équations de parité permettent de définir la séquence redondante à ajouter. Ces équations sont regroupées pour former la matrice de parité.

Chaque code LDPC est défini par une matrice de parité unique H , de dimension $m \times N$. Il y a alors m équations de parité. Le nombre de '1' par ligne, appelé poids du code, correspond au nombre de bits mis en jeu dans l'équation de parité.

La particularité des codes LDPC est que la matrice de parité est creuse, c'est-à-dire qu'elle ne contient qu'un très faible nombre d'éléments non nuls, tandis qu'elle comporte beaucoup de '0'. Ceci permet d'avoir un schéma de décodage simplifié et donc un décodage

rapide comparé aux autres types de codes en blocs usuels.

Une représentation classique de la matrice H se fait par un graphe de Tanner, aussi appelé graphe bipartite, comme celui présenté figure 1.8.

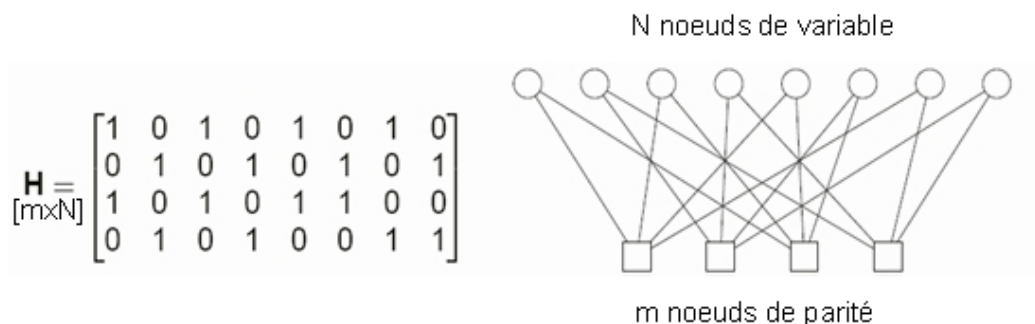


FIG. 1.8 – Représentation équivalente de la matrice de parité par le graphe de Tanner

Dans cette représentation, les carrés symbolisent les nœuds de parité, et leur nombre correspond au nombre de bits participant à l'équation de parité. Il y en a donc autant que de lignes dans la matrice H , c'est-à-dire m . Les ronds symbolisent les nœuds de variable, et leur nombre correspond au nombre de bits incidents au décodeur qui sont testés par les équations de parité, il y en a donc N (c'est-à-dire le nombre de colonnes de H).

Chaque bit à '1' dans la matrice de parité est représenté sur le graphe de Tanner par une liaison entre le nœud de variable et le nœud de parité, correspondant à sa position dans la matrice. Ainsi, si w_c est le poids des colonnes de la matrice, chaque nœud de variable est lié à w_c nœuds de parité (dans l'exemple w_c vaut 2). De même, si on note w_l le poids des lignes, chaque nœud de parité (ou nœud de contrôle) est lié à w_l nœuds de variable.

Lors du décodage, les opérations de contrôle de parité se représentent par des échanges via ces liaisons entre les nœuds de variables et ceux de parité.

Le schéma de codage et décodage utilisant un algorithme itératif peut être représenté d'une manière générale comme sur la figure 1.9.

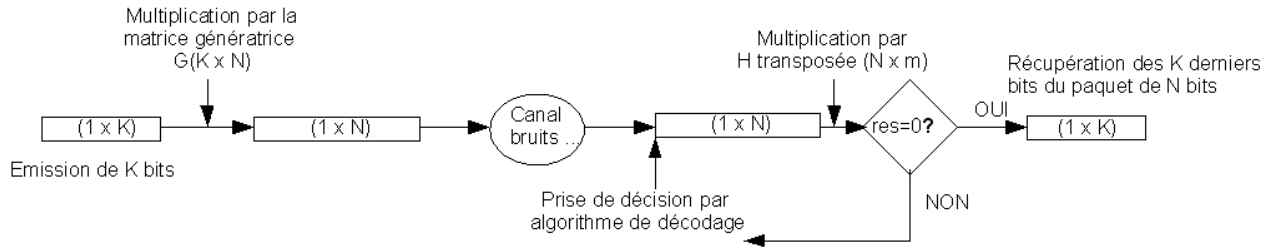


FIG. 1.9 – Chaîne de transmission avec codage/décodage

L'émetteur comporte un bloc de codage qui réalise la multiplication de la donnée utile avec la matrice génératrice. Cette opération d'encodage ne pose a priori pas de contraintes particulières si ce n'est le fait de travailler sur des paquets de données de taille fixe imposée par K (ou N).

Au niveau de la réception, un algorithme itératif dont le rôle est de déterminer la séquence la plus probable, est mis en place. Une décision est prise sur les bits composant le mot reçu et un test est effectué à chaque itération pour vérifier si le mot décidé est un mot de code ou non. Cette opération est réalisée par la multiplication du mot décidé avec la matrice H transposée de sorte à vérifier l'équation de parité. Tant que le mot décidé n'est pas un mot de code et tant que le nombre maximal d'itérations n'est pas atteint, les itérations se poursuivent. La pertinence de l'utilisation de codes LDPC repose sur les caractéristiques des éléments du décodage qui sont principalement la structure de la matrice de parité ainsi que l'algorithme de décodage utilisé.

Les paragraphes suivants présentent les méthodes de construction les plus courantes ainsi que les algorithmes pour le décodage de codes LDPC, en se limitant aux codes LDPC binaires. Enfin, l'application des codes LDPC aux transmissions optiques permettra de conclure sur ce type de codage correcteur d'erreurs.

1.4.2 Méthodes de construction

1.4.2.1 Principe général

Pour la génération de codes LDPC, le principe est de construire à partir de la matrice de parité H la matrice génératrice du code G . De part la propriété des codes linéaires, la matrice génératrice du code admet une représentation systématique. On procède de la manière suivante pour la déterminer :

- Transformation de la matrice H sous la forme systématique $H' = [I \cdot A]$, où I représente la matrice identité
- Calcul de $G = [A^T \cdot I]$

Cette transformation sous la forme systématique se fait à l'aide d'éliminations suivant le pivot de Gauss. A la fin de la transformation, la matrice H d'origine doit être soumise au réarrangement des colonnes suivant les opérations effectuées durant l'élimination de Gauss.

Deux cas possibles peuvent se présenter à la fin de la transformation :

- H' ne comporte pas de lignes dépendantes : dans ce cas la transformation aboutit à une matrice identité de taille $m \times m$. Par conséquent A est de taille $(m \times N - m)$ et G de taille $(N - m \times N)$,
- H' comporte au moins 2 lignes dépendantes : dans ce cas, la transformation fait apparaître des lignes tout à 0. Ces lignes sont supprimées et la matrice transformée comporte moins de lignes. Soit L le nombre de lignes dépendantes de H' , la transformation aboutit à I de taille $(m-L \times m-L)$. A est alors de taille $(m - L \times N - (m - L))$, et G de taille $(N - (m - L) \times N)$.

Dans tous les cas, considérons que G est de dimension $K \times N$, en posant $K = N - (m - L)$. Cela signifie que pour un mot à coder U de K bits, on obtient un mot codé

$C_{[1 \times N]} = U_{[1 \times K]} \cdot G_{[K \times N]}$ de taille N . La taille des paquets de bits à émettre va donc dépendre de la matrice H utilisée et de la dépendance de ses lignes.

Une fois que $H(m \times N)$ et $G(K \times N)$ sont construites, on peut vérifier que, par construction :

$$G \cdot H^T = 0 \tag{1.30}$$

L'équation (1.30) est l'équation de parité sous forme matricielle. Elle est équivalente à l'ensemble des K équations de parité qui caractérisent le code LDPC(N, K). Dans ce cas, si un mot codé $C = U \cdot G$ est reçu sans erreur, on a $C \cdot H^T = 0$, puisque $C \cdot H^T = (U \cdot G) \cdot H^T = U \cdot (G \cdot H^T)$ et $G \cdot H^T = 0$.

Ainsi, la matrice H est utilisée pour vérifier la parité $v \cdot H^T = 0$ où v est le mot reçu. Si le résultat est non nul, une erreur est détectée et doit être corrigée. Comme le décodage est basé sur la structure de la matrice de parité, la construction de cette dernière est une étape très importante dans la génération d'un code.

Un critère ayant une influence importante sur la convergence du décodage est la présence de cycles courts dans la matrice de parité. Un cycle d'ordre S est caractérisé par le fait que S éléments à '1' dans la matrice se trouvent sur les mêmes $S/2$ lignes et colonnes de telle sorte que les échanges entre les noeuds de contrôle et de variable correspondant n'évoluent pas et ne convergent pas. En effet, si des erreurs ont lieu sur les bits participant au cycle, elles peuvent se "compenser" et ne pas être corrigées. Ce phénomène est illustré dans la partie de ce chapitre consacrée au décodage. La taille du cycle le plus court dans la matrice de parité (aussi appelé "girth") est ainsi une caractéristique importante puisque les performances seront d'autant meilleures que ce cycle minimal est grand.

D'une manière générale, la matrice H peut être contruite selon une procédure aléatoire ou bien déterministe. De plus, on distingue aussi les matrices dites régulières

ou irrégulières. Ces deux notions sont définies dans les paragraphes suivants avant d'être détaillées par la suite à l'aide d'exemples de méthodes de construction parmi les plus courantes.

Régulières - Irrégulières

Une matrice de parité est dite régulière lorsque le nombre de '1' par ligne et par colonne est constant. Si au moins l'une de ces conditions n'est pas respectée, le code (ou la matrice) n'est plus régulier mais irrégulier. On distingue alors les codes irréguliers par ligne des codes irréguliers par colonne, mais un code peut être à la fois irrégulier par ligne et par colonne. Si le poids des lignes et celui des colonnes ne sont pas constants, la matrice est dite irrégulière. Si seul le poids des colonnes est constant, tandis que celui des lignes ne l'est pas, la matrice est régulière par colonne et de même s'il s'agit des lignes.

Lorsqu'il s'agit de matrices irrégulières par ligne ou par colonne, on définit le poids comme la moyenne du nombre de '1' par ligne (ou par colonne). Pour de tels codes irréguliers, le nombre de '1' par ligne (ou colonnes) est généralement choisi le plus uniforme possible et donc proche du poids moyen.

Aléatoires - Déterministes

Les méthodes aléatoires consistent à placer les '1' dans la matrice de parité sur des positions au hasard. A l'opposé les méthodes déterministes consistent à construire la matrice selon une procédure particulière. Dans chacun des deux cas, le nombre de '1' par ligne et par colonne peut être constant ou non. Il y a donc des codes aléatoires réguliers ou irréguliers ainsi que des codes déterministes réguliers ou non.

Quelle que soit la structure de la matrice de parité, une opération de masquage de matrice permet d'obtenir à partir d'un code LDPC un nouveau code aux propriétés différentes. Cette opération revient à multiplier la matrice de parité du code d'origine par une matrice de masquage composée par définition de sous-matrices vides [46]. La

conséquence est que certains '1' dans la matrice de parité sont supprimés ce qui implique que le graphe de Tanner comporte moins de liaisons entre les noeuds. Ainsi, la longueur du cycle le plus court a de fortes chances d'être augmentée et le code d'être plus performant.

1.4.2.2 Méthodes aléatoires

Le principe pour obtenir une matrice de parité avec une méthode de construction aléatoire, est de créer une matrice vide de dimension souhaitée $(N - K) \times N$ puis d'y placer des '1' aléatoirement sur les lignes et colonnes.

Cette méthode de construction présente l'avantage de ne pas imposer a priori de contraintes sur les dimensions du code. Cependant, de manière à avoir un poids de lignes w_l et un poids de colonnes w_c constants (codes réguliers) ou seulement uniformes (codes non réguliers) il apparaît quelques contraintes, présentées dans les paragraphes suivants pour des codes aléatoires, réguliers ou non réguliers.

a- Construction de codes réguliers aléatoires

Pour les codes dits réguliers, toutes les colonnes doivent avoir le même poids w_c et toutes les lignes doivent avoir le même poids w_l . Ainsi, le nombre de '1' dans la matrice est donné par n_1 :

$$n_1 = n_c \times w_c = n_l \times w_l \tag{1.31}$$

où n_c et n_l sont respectivement le nombre de colonnes et le nombre de lignes et sont liés aux dimensions du code LDPC(N, K) par : $n_c = N - K$ et $n_l = N$.

On a alors la relation :

$$(N - K) \times w_l = N \times w_c \tag{1.32}$$

Le poids des colonnes w_c est généralement choisi assez faible (souvent 3) de manière à avoir une matrice dite "creuse" c'est-à-dire qui ne possède pas beaucoup de '1'. Ainsi, la

relation (1.32) n'est satisfaite que pour certaines valeurs des paramètres, N , K . En effet, on a :

$$\frac{w_l}{w_c} = \frac{N}{N - K} \quad (1.33)$$

Ainsi, si N et K sont fixés, on ne peut pas nécessairement satisfaire (1.33) avec une contrainte supplémentaire sur w_c (égal à 3 ou 4 par exemple) ou w_l . Inversement, si w_l et w_c sont fixés, (1.33) ne peut pas être satisfaite pour n'importe quelles valeurs de N et K . Par la suite, on notera (N_s, K_s) les dimensions souhaitées du code et (N, K) les dimensions réelles du code obtenu.

Pour construire un code LDPC régulier de poids de colonne w_c et de ligne w_l constants, on peut citer 2 méthodologies : celle énoncée par Gallager [2] utilisant des sous-matrices aléatoirement permutées ou la méthode la plus simple proposée par MacKay [45] où la position de tous les éléments est indépendante.

Dans la première méthode, les poids w_l et w_c imposent de fortes contraintes sur les dimensions (N, K) du code. Cette méthode est détaillée en annexe C.1.

Dans la deuxième méthode, présentée en annexe C.2, N est fixé, ainsi que le poids des colonnes w_c . La valeur de K est soumise à quelques contraintes, mais est choisie de manière à s'approcher au mieux de la dimension souhaitée K_s . Finalement, le poids des lignes w_l est entièrement fixé par les trois autres paramètres.

b- Construction de codes non réguliers aléatoires

A partir de la méthode de MacKay (détaillée en annexe C.2), il est facile de construire des codes non réguliers par ligne, mais réguliers par colonne. En effet, la procédure est identique à part la dernière étape uniformisant le poids des lignes qui n'est pas réalisée.

Une autre méthode pour construire des codes aléatoires, irréguliers ou réguliers par colonne ou par ligne consiste à suivre la méthode des deux vecteurs comme exposée par Luby [47] et détaillée en annexe C.3.

c- Post-traitement des matrices aléatoires obtenues

A l'issue de chacune des méthodes citées, la matrice finale obtenue doit être soumise à deux étapes de post-traitement. Premièrement, certaines lignes de la matrice de parité peuvent être linéairement dépendantes. S'il en existe, elles doivent être supprimées et le code aura des dimensions légèrement différentes de celles déterminées lors de la construction.

La deuxième étape consiste à détecter la présence de cycles courts et à les supprimer de manière à éviter une divergence de l'algorithme de décodage. En effet, le caractère aléatoire fait apparaître des cycles dans la matrice de parité. La figure 1.10 représente un exemple de cycle d'ordre 4 dans la matrice de parité ainsi que sa représentation équivalente par le graphe de Tanner.

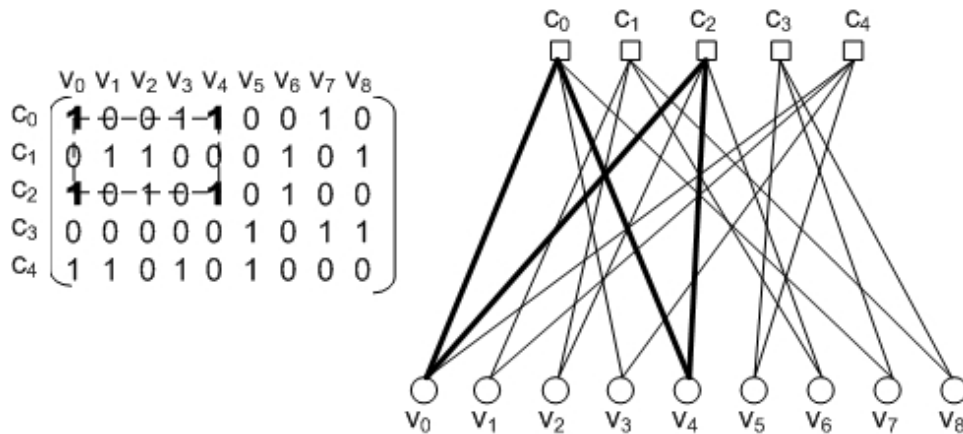


FIG. 1.10 – Exemple de cycle dans la matrice de parité et dans le graphe de Tanner équivalent

Plus les cycles sont longs, plus ils sont difficiles à détecter, mais moins ils ont d'influence sur le décodage. Les plus simples à détecter sont les cycles d'ordre 4 correspondant à deux colonnes ayant deux fois la même entrée. Les cycles d'ordres plus élevés correspondent à des figures particulières, plus difficiles à repérer mais pouvant l'être en suivant une méthodologie particulière [48]. Lorsqu'un cycle est détecté, un des '1' formant le cycle est

supprimé.

Chacune de ces deux étapes conduit, suite à la suppression d'éléments non nuls dans la matrice, à une matrice irrégulière même si elle a été construite de manière régulière. Cependant, lorsque les dimensions de la matrice deviennent élevées ces étapes ne sont pas nécessaires car la présence de cycles et de lignes dépendantes devient peu probable.

1.4.2.3 Méthodes déterministes

Contrairement aux méthodes aléatoires, les méthodes déterministes suivent une règle de construction bien précise. Parmi ces méthodes déterministes, les plus connues sont les constructions basées sur la géométrie finie (Euclidienne ou projective) [32, 36, 39, 49] et celles basées sur les BIBD (Balanced Incomplete Block Design) [38, 40, 41, 50]. D'autres méthodes de constructions basées sur la structure des codes RS [51], des modèles graphiques [52], des concepts algébriques [53] ou combinatoires [33, 37] ont été envisagées. L'intérêt d'adopter une méthodologie de construction particulière est d'éviter l'apparition de cycles courts dus à la position aléatoire des '1' dans la matrice de parité. De plus, tandis qu'on n'est pas sûr, à l'aide des méthodes de constructions aléatoires, d'obtenir la matrice de taille et poids désirés, les matrices créées par des méthodes déterministes correspondent parfaitement aux dimensions prévues. Cependant, la construction impose certaines contraintes sur la dimension des codes notamment. Dans les paragraphes suivants, deux méthodes basées sur les BIBD issues de [50] sont présentées. En effet, ces méthodes sont simples à implémenter et des études ont montré qu'elles conduisent à de bonnes performances en terme de correction d'erreurs [38, 40, 41, 50].

a- Méthode BIBD simple

Les constructions BIBD sont basées sur les corps de Galois $GF(q)$ où q est un nombre premier. La valeur de q impose une condition sur la taille de la matrice de parité du code

LDPC. En effet, à partir de q , on définit le corps de Galois $\text{GF}(q)$, ainsi qu'un élément primitif de ce corps noté x . Pour un même corps de Galois $\text{GF}(q)$, différents codes BIBD peuvent être construits. Leur définition repose alors sur le poids imposé pour les colonnes w_c ainsi que sur le nombre de séquences BIBD différentes choisies pour former la matrice de parité.

Le poids des colonnes w_c définit le nombre de '1' dans chaque séquence notée B_i de longueur q . Ces séquences, composées de '0' et de '1' sont telles que l'intercorrélacion entre séquences soit au plus égale à 1. Pour une intercorrélacion maximale de 1, le nombre maximal de séquences différentes t est donné par : $t = \left\lfloor \frac{q-1}{w_c \cdot (w_c-1)} \right\rfloor$.

Une fois w_c et t déterminés, on construit à partir du nombre premier q , les séquences B_i (i de 0 à $t-1$) selon la méthode expliquée ci-après.

- Si w_c est pair : $B_i = \{0\} \cup \{x^{i \frac{w_c}{2} + j t w_c} (q)\}_{j \in [0, 1, \dots, w_c-2]}$, on rappelle que la notation (q) signifie que l'opération a lieu modulo q .
- Si w_c est impair : $B_i = \{x^{i \frac{w_c-1}{2} + j t (w_c-1)} (q)\}_{j \in [0, 1, \dots, w_c-1]}$

Avec cette notation, les séquences B_i sont définies par la position des '1' dans la séquence.

Parmi les t différentes séquences B_i , α séquences sont choisies pour construire la matrice de parité. Pour chacune de ces séquences de longueur q , il existe q permutations circulaires, et chaque vecteur permuté constitue une colonne de la matrice de parité H . Ainsi, la matrice de parité comporte q lignes et $N = \alpha q$. La matrice de codage possédera alors un nombre de lignes K et de colonnes N tous deux multiples de q . En effet, la taille $(K \times N)$ est donnée par $((\alpha - 1) \cdot q \times \alpha \cdot q)$.

Le rendement du code est alors $R = K/N = (\alpha - 1)/\alpha$.

Comme l'intercorrélacion entre deux séquences est au maximum égale à 1, les cycles d'ordre 4 ne peuvent pas apparaître, et ainsi les cycles les plus courts sont d'ordre 6. Un exemple est donné figure 1.11 avec $q=13$, $w=3$. Ceci impose $t=2$, et on a choisi $\alpha = t = 2$.

$$\begin{array}{c}
 \text{Permutations de } B_0 \qquad \qquad \text{Permutations de } B_1 \\
 \left. \begin{array}{l}
 00001000001010010011000000 \\
 1000010000010000010011000000 \\
 0100001000001000001001100000 \\
 1010000100000000000100110000 \\
 0101000010000000000001001100 \\
 001010000100000000000100110 \\
 00010100001000000000010011 \\
 000010100001010000000001001 \\
 000001010000111000000000100 \\
 1000001010000001100000000010 \\
 0100000101000000110000000001 \\
 0010000010100100100110000000 \\
 0001000001010010010011000000 \\
 0001000001010010010011000000
 \end{array} \right\} H=
 \end{array}$$

FIG. 1.11 – Exemple de matrice de parité construite avec la méthode BIBD

Les deux séquences sont $B_0 = \{1, 3, 9\}$ et $B_1 = \{7, 8, 11\}$, les chiffres correspondant à la position des 1 dans le vecteur B_i de longueur q . Les 13 premières colonnes de H sont construites par permutations de B_0 . Les 13 suivantes sont construites par permutation de B_1 .

b- Méthode BIBD par décomposition

Cette méthode de construction est basée sur le même principe que la méthode précédente auquel est ajoutée une "décomposition" de la matrice finale. Cela a pour but de rendre la matrice plus aérée et donc de diminuer le nombre de cycles d'ordre 6. Le principe de construction pour q nombre premier, et w_c poids des lignes fixés, est le suivant :

- Générer les t séquences $B_i, i \in [0, t - 1]$ comme dans le cas de la méthode BIBD simple.
- Pour chaque i de 0 à $t - 1$, construire les matrices carrées Q_i par permutation de la séquence B_i , c'est-à-dire : $Q = \left(B_i B_i^{(1)} \dots B_i^{(q-1)} \right)$ avec $B_i^{(j)}$ le vecteur de base B_i

permuté j fois. Chaque matrice Q_i est donc de taille $q \times q$ et possède w_c éléments à '1' par ligne et par colonne.

- Pour chaque matrice Q_i , décomposer chacune de ses lignes en w_c lignes. La première ligne sera composée de '0' et d'un seul '1' à la position correspondant au premier '1' de la ligne de Q_i en question, à droite de, ou sur la diagonale de Q_i . La deuxième ligne possède un seul '1' à la position du deuxième '1' après la diagonale dans la ligne de Q_i et ainsi de suite jusqu'au $w_c^{ième}$. Si on indexe par k chaque ligne de Q_i , on notera $g_{i,k}^j$ la $j^{ième}$ ligne de la décomposition de la ligne k de la matrice Q_i .
- Construire les matrices $Q_{i,j}$, pour $j \in [0, w_c - 1]$ à partir des $g_{i,k}^j$ de la manière suivante :

$$Q_{i,j} = \begin{pmatrix} g_{i,0}^j \\ g_{i,1}^j \\ \dots \\ g_{i,q-1}^j \end{pmatrix}$$

Les $t \times w_c$ matrices $Q_{i,j}$, $i \in [0, t - 1]$ et $j \in [0, w_c - 1]$ sont donc de dimension $q \times q$

- Construire pour chaque j les matrices D_j à partir des $Q_{i,j}$: $D_j = (Q_{0,j} Q_{1,j} \dots Q_{n-1,j})$. Ces matrices sont alors de taille $q \times (nq)$.
- En dernier lieu, construire la matrice de parité H de dimension $(w_c q) \times (nq)$ à partir des D_j :

$$H = \begin{pmatrix} D_0 \\ D_1 \\ \dots \\ D_{w_c-1} \end{pmatrix}$$

Cette dernière peut aussi s'exprimer à partir des $Q_{i,j}$ ou des $g_{i,k}^j$:

$$H = \begin{pmatrix} Q_{0,0} & Q_{1,0} & \cdots & Q_{n-1,0} \\ Q_{0,1} & Q_{1,1} & \cdots & Q_{n-1,1} \\ \cdots & \cdots & \cdots & \cdots \\ Q_{0,w_c-1} & Q_{1,w_c-1} & \cdots & Q_{n-1,w_c-1} \end{pmatrix} = \begin{pmatrix} g_{0,0}^0 & g_{1,0}^0 & \cdots & g_{n-1,0}^0 \\ g_{0,1}^0 & g_{1,1}^0 & \cdots & g_{n-1,1}^0 \\ \cdots & \cdots & \cdots & \cdots \\ g_{0,q-1}^0 & g_{1,q-1}^0 & \cdots & g_{n-1,q-1}^0 \\ g_{0,0}^1 & g_{1,0}^1 & \cdots & g_{n-1,0}^1 \\ g_{0,1}^1 & g_{1,1}^1 & \cdots & g_{n-1,1}^1 \\ \cdots & \cdots & \cdots & \cdots \\ g_{0,q-1}^1 & g_{1,q-1}^1 & \cdots & g_{n-1,q-1}^1 \\ \cdots & \cdots & \cdots & \cdots \\ g_{0,0}^{w_c-1} & g_{1,0}^{w_c-1} & \cdots & g_{n-1,0}^{w_c-1} \\ g_{0,1}^{w_c-1} & g_{1,1}^{w_c-1} & \cdots & g_{n-1,1}^{w_c-1} \\ \cdots & \cdots & \cdots & \cdots \\ g_{0,q-1}^{w_c-1} & g_{1,q-1}^{w_c-1} & \cdots & g_{n-1,q-1}^{w_c-1} \end{pmatrix}$$

1.4.2.4 Conclusions

Les codes LDPC sont classifiés en fonction des caractéristiques de la matrice de parité. Plusieurs études ont été réalisées pour déterminer des structures permettant d'obtenir de bonnes performances en terme de correction d'erreurs. La généralisation des codes LDPC binaires aux codes M-aires c'est-à-dire dont les symboles sont composés de M bits a ainsi été envisagée [35, 54].

Cependant, en vue de l'insertion des codes LDPC dans de nouveaux standards de transmission, la question de l'implémentation en pratique de tels codes a remis en jeu le compromis efficacité/complexité d'implémentation. En effet, la complexité non seulement

du codeur est un paramètre important mais aussi celle du décodeur. Certaines études ont montré qu'une structure cyclique par exemple, peut largement simplifier le décodeur [55, 56], ce qui dans le contexte optique devient un critère important. Dans ce contexte, les codes LDPC binaires déterministes apparaissent finalement comme un bon compromis. Par la suite, nous utiliserons donc des codes LDPC déterministes basés sur la méthode de construction BIBD.

1.4.3 Décodage

1.4.3.1 Principe général

L'algorithme optimal pour les codes LDPC est celui du maximum de vraisemblance [2, 10]. Or, de part sa complexité, son implémentation n'est pas envisageable et ce type de décodage optimal n'est jamais utilisé. D'autres méthodes de décodage basées sur un décodage sous-optimal ont été mises en place. Ainsi, les algorithmes de décodage itératifs basés sur la propagation de croyance, dits à transferts de messages ("Message Passing Algorithm"), sont le plus couramment utilisés [16, 17, 57]. Comme le nom l'indique, ce type d'algorithme est basé sur l'échange de messages entre les noeuds de variables et les noeuds de parité. Les messages transmis donnent une indication sur la fidélité des bits reçus c'est pourquoi on parle de propagation de croyance. Le message émis d'un noeud de variable vers un noeud de parité tient compte de la valeur de ce noeud de variable, ainsi que des messages reçus à l'étape précédente des noeuds de parité à l'exception de celui concerné par le message. Inversement, le raisonnement est identique en ce qui concerne les messages émis par les noeuds de parité vers les noeuds de variable. A chaque itération, une décision est prise au niveau des noeuds de variable pour former le message décidé \hat{v} . Un nombre maximal d'itérations est imposé de sorte que tant que ce nombre n'est pas

atteint, si $\hat{v} \cdot H^T \neq 0$ une nouvelle itération commence. Lorsque $\hat{v} \cdot H^T = 0$, cela signifie que le mot décidé \hat{v} est un mot de code, et que la ou les erreurs sont censées être corrigées, donc le décodage s'arrête.

Si le nombre maximal d'itérations est atteint et que $\hat{v} \cdot H^T \neq 0$, le décodage échoue et il y a au moins une erreur dans le paquet de N bits.

Le plus simple des décodages est celui réalisé en dur, traitant des valeurs binaires obtenues après une détection à seuil. Ce type de décodage, moins utilisé que le décodage souple, car conduisant à de moins bonnes performances, permet néanmoins de bien comprendre le fonctionnement du transfert de messages et est présenté dans le paragraphe suivant.

1.4.3.2 Décodage dur ou hard

Pour illustrer le principe de décodage, nous utilisons la représentation de Tanner de la matrice de parité. Le principe, basé sur le contrôle du bit de parité est le suivant :

1. Initialisation des N noeuds de variable avec les valeurs du mot reçu. Chaque noeud de variable émet sa valeur vers les w_c noeuds de parité auquel il est lié.
2. A partir de toutes les informations reçues par les différents noeuds de variable, chaque noeud de contrôle effectue la vérification (ou le contrôle) de parité (somme des éléments reçus, hors celui du noeud de contrôle en question, modulo 2 = 0), et renvoie à chaque noeud de variable la valeur nécessaire pour satisfaire cette équation de parité par rapport aux autres noeuds de variable.
3. Chaque noeud de variable actualise la valeur à émettre aux noeuds de contrôle en fonction des w_l valeurs qu'il reçoit et de la valeur qu'il possédait avant, en choisissant la valeur (0 ou 1) qui a la majorité.
4. Ces dernières valeurs constituent les valeurs décidées sur les noeuds de variable. Si

elles sont identiques à celles établies au tour précédent, l'algorithme s'arrête. Sinon, il recommence à l'étape 2.

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

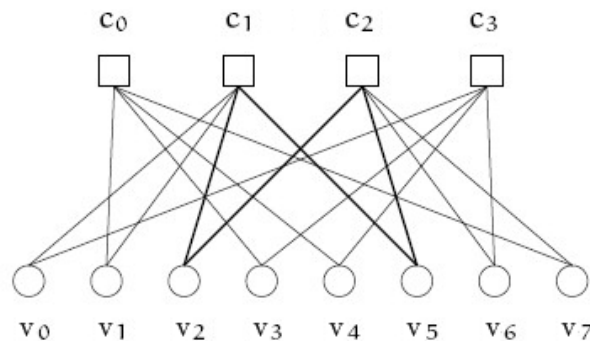


FIG. 1.12 – Exemple de décodage à partir du graphe de Tanner

Illustrons cet algorithme avec l'exemple représenté figure 1.12. Sur cette figure, on note $v_i, i \in \{0, 1, \dots, 7\}$ les noeuds de variable et $c_i, i \in \{0, \dots, 3\}$ les noeuds de contrôle (ou de parité).

On suppose que le mot codé émis est $u = [10010101]$, et que l'on reçoit le mot suivant, avec le deuxième bit erroné $v = [11010101]$. Pour chacune des étapes de l'algorithme, nous allons évaluer les valeurs présentes aux différents noeuds.

1. Initialisation :

$$v_0 = 1, v_1 = 1, v_2 = 0, v_3 = 1, v_4 = 0, v_5 = 1, v_6 = 0, v_7 = 1.$$

Les noeuds de contrôle reçoivent (\Leftarrow) les informations via les liaisons :

$$c_0 \Leftarrow 1 \text{ (de } v_1), 1 \text{ (de } v_3), 0 \text{ (de } v_4), 1 \text{ (de } v_7)$$

$c_1 \Leftarrow 1$ (de v_0), 1 (de v_1), 0 (de v_2), 1 (de v_5)

$c_2 \Leftarrow 0$ (de v_2), 1 (de v_5), 0 (de v_6), 1 (de v_7)

$c_3 \Leftarrow 1$ (de v_0), 1 (de v_3), 0 (de v_4), 0 (de v_6)

2. Chaque noeud de contrôle renvoie (\Rightarrow) à chaque noeud de variable la valeur nécessaire pour satisfaire l'équation de parité.

Pour le noeud de contrôle c_0 :

$\Rightarrow 0$ à v_1 car $\sum_{i \neq 1} v_i = 1 + 0 + 1(2) = 0$

$\Rightarrow 0$ à v_3 car $\sum_{i \neq 3} v_i = 1 + 0 + 1(2) = 0$

$\Rightarrow 1$ à v_4 car $\sum_{i \neq 4} v_i = 1 + 1 + 1(2) = 1$

$\Rightarrow 0$ à v_7 car $\sum_{i \neq 7} v_i = 1 + 1 + 0(2) = 0$

De même pour les autres noeuds :

$c_1 \Rightarrow 0$ (à v_0), 0 (à v_1), 1 (à v_2), 0 (à v_5)

$c_2 \Rightarrow 0$ (à v_2), 1 (à v_5), 0 (à v_6), 1 (à v_7)

$c_3 \Rightarrow 1$ (à v_0), 1 (à v_3), 0 (à v_4), 0 (à v_6)

3. Chaque noeud de variable décide de la valeur à émettre au tour suivant vers les noeuds de contrôle (0 ou 1) selon la majorité :

v_0 , initialement à 1 a reçu 0 de c_1 et 1 de c_3 donc $v_0=1$

De même pour les autres noeuds :

$v_1 = 0, v_2 = 0, v_3 = 1, v_4 = 0, v_5 = 1, v_6 = 0, v_7 = 1$

4. Dans ce cas, toutes les valeurs décidées sur les noeuds de variable ne sont pas identiques à l'état précédent, un nouveau tour a alors lieu. Remarquons qu'à l'issue de ce premier tour, l'erreur sur le bit c_1 a été corrigée. Ainsi, le tour suivant, les valeurs des c_i seront toutes identiques à celles obtenues à l'issue de ce tour, et l'algorithme s'arrêtera, en ayant corrigé l'erreur.

La matrice de l'exemple précédent présente un cycle d'ordre 4 entre les éléments c_0, v_3, v_4, c_3 . Dans l'exemple précédent, le mot reçu erroné a été corrigé, le cycle n'a donc pas eu d'influence sur le décodage. Nous allons voir à travers l'exemple suivant les conséquences que ce cycle peut avoir sur le décodage.

On suppose que le mot reçu possède 2 bits erronés aux positions du cycle, c'est-à-dire en v_3 et v_4 : $v = [1000\mathbf{1}101]$. Dans ce cas, les différentes étapes conduisent à :

1. Initialisation :

$$v_0 = 1, v_1 = 0, v_2 = 0, v_3 = 0, v_4 = 1, v_5 = 1, v_6 = 0, v_7 = 1.$$

Les noeuds de contrôle reçoivent (\Leftarrow) les informations via les liaisons :

$$c_0 \Leftarrow 1, 0, 1 \text{ et } 1$$

$$c_1 \Leftarrow 1, 1, 0 \text{ et } 1$$

$$c_2 \Leftarrow 0, 1, 0 \text{ et } 1$$

$$c_3 \Leftarrow 1, 0, 1 \text{ et } 0$$

2. Chaque noeud de contrôle renvoie (\Rightarrow) à chaque noeud de variable la valeur nécessaire pour satisfaire l'équation de parité.

$$c_0 \Rightarrow 0 \text{ (à } v_1), 1 \text{ (à } v_3), 0 \text{ (à } v_4), 0 \text{ (à } v_7)$$

$$c_1 \Rightarrow 0 \text{ (à } v_0), 0 \text{ (à } v_1), 1 \text{ (à } v_2), 0 \text{ (à } v_5)$$

$$c_2 \Rightarrow 0 \text{ (à } v_2), 1 \text{ (à } v_5), 0 \text{ (à } v_6), 1 \text{ (à } v_7)$$

$$c_3 \Rightarrow 1 \text{ (à } v_0), 0 \text{ (à } v_3), 1 \text{ (à } v_4), 0 \text{ (à } v_6)$$

3. Chaque noeud de variable décide la valeur 0 ou 1 selon la majorité :

$$v_0 = 1, v_1 = 0, v_2 = 0, v_3 = 0, v_4 = 1, v_5 = 1, v_6 = 0, v_7 = 1$$

4. Dans ce cas, toutes les valeurs décidées sur les noeuds de variables sont identiques au tour précédent, l'algorithme s'arrête donc. Or le mot n'a pas du tout été corrigé. Ceci est dû au fait que les erreurs sont paires et qu'elles sont situées sur les noeuds de variables correspondant aux positions du cycle, donc elles se compensent.

1.4.3.3 Décodage souple ou soft

Le décodage souple implanté en pratique est le décodeur somme-produit, utilisant le principe de la propagation de croyance, ou "Belief Propagation" [16, 17, 57]. Dans cet algorithme, les messages transportés entre les noeuds de variable et de contrôle sont des probabilités estimées sur la fiabilité de chacun des bits reçus. Plus particulièrement, chaque noeud de variable v_j envoie à chaque noeud de contrôle c_i , la probabilité que la valeur correcte du bit soit un 0 ou soit un 1, en fonction de la valeur de v_j ainsi que des valeurs communiquées au tour précédent par les noeuds de contrôles $c_{i'}(i' \neq i)$. De même les noeuds de parité envoient aux noeuds de variable une probabilité calculée à partir des valeurs communiquées au tour précédent par les noeuds de variable $v_{j'}(j' \neq j)$ ainsi que de la valeur antérieure de ce noeud de parité.

À chaque itération, une valeur est décidée au niveau des noeuds de variable pour chaque variable à partir des messages reçus, et l'opération $\hat{v} \cdot H^T$ est effectuée pour vérifier si le résultat donne 0. Les itérations continuent si ce n'est pas le cas (jusqu'au nombre maximal d'itérations).

Les itérations successives conduisent à des multiplications entre les probabilités d'où des résultats numériques très faibles qui peuvent être erronés par les imprécisions de calcul. C'est pourquoi les messages sont généralement échangés sous forme de rapport de vraisemblance dans le domaine logarithmique (LLR : Logarithmic Likelihood Ratio). Dans ce cas, les opérations effectuées sont des additions, ce qui enlève le problème d'imprécision des calculs.

Notons que, de la même manière que pour le décodage hard, la présence de cycles engendre des divergences de calcul c'est pourquoi il faudra les éviter.

Les deux versions de l'algorithme BP (échanges de probabilité ou de LLR) sont détaillées en annexe D.

1.4.3.4 Conclusions

La réduction de la complexité du décodage est non seulement recherchée au niveau des méthodes de constructions de code, mais aussi au niveau des algorithmes de décodages choisis sous-optimaux. Dans ce cadre, des études portant conjointement sur la structure, les performances, et l'implémentation ont été réalisées [55, 56].

Dans cette thèse, nous nous focaliserons par la suite pour l'évaluation des performances sur l'algorithme de décodage souple à propagation de croyance (BP).

Le paragraphe suivant donne une conclusion sur l'application des codes LDPC aux transmissions optiques.

1.4.4 Conclusion - Application des codes LDPC aux transmissions optiques

Les codes LDPC ont fait l'objet de nombreuses études appliquées aux canaux de transmissions optiques [30–41]. Dans [30], le modèle utilisé pour caractériser le canal optique est le plus simple c'est-à-dire le modèle AWGN. Les autres travaux ont pris en compte à l'aide d'un simulateur système différentes perturbations propres au canal optique telles que le bruit d'émission spontanée, les non-linéarités de la fibre, la dispersion chromatique ainsi que l'interférence entre symboles [31].

D'autre part, différentes familles de codes LDPC ont été étudiées. Des codes obtenus à partir de considérations combinatoires ont été envisagés dans [32, 33, 37]. Ces codes possèdent un cycle le plus court au moins égal à 6 et présentent aussi l'avantage d'avoir une construction simple, qui nécessite une faible capacité de mémoire au niveau de

l'implémentation. Ce type de code conduit à des performances intéressantes, qui sont comparées dans [32] à d'autres types de codes. Plus particulièrement, [32] compare les performances de codes structurés, obtenus à partir des combinatoires [33,37] mais aussi de la géométrie finie [36,39], aux performances de turbo-codes et de codes RS concaténés. Les résultats montrent que la dégradation de performances due au calcul quantifié des LLR n'est pas significative et que les codes LDPC sont tout à fait compétitifs par rapport aux autres. En effet, il est établi dans [36] que les résultats en terme de gain de codage sont meilleurs de 2 dB environ avec les codes LDPC basés sur la géométrie finie par rapport aux codes RS concaténés. Par comparaison au code de référence RS(255,239), l'amélioration avec des codes LDPC de redondance environ égale à 20 %, est de 3-4 dB.

Les codes étudiés dans [38, 40, 41] sont eux aussi des codes structurés, mais basés sur la méthode de construction BIBD. Les codes BIBD ont par construction un cycle minimal de taille 6 mais un algorithme [38] peut être utilisé pour supprimer ces cycles, et obtenir par conséquent des codes de cycle minimal de taille 8. Ces codes possèdent des caractéristiques intéressantes en vue d'une implémentation tout en assurant de bonnes performances [38, 40, 41].

D'autres types de codes LDPC possédant diverses propriétés ont aussi été étudiés pour les communications optiques. Les performances des codes LDPC étendus aux codes non binaires sont présentées dans le cadre d'une transmission optique dans [35]. Les codes LDPC "généralisés" obtenus à partir de codes linéaires simples (Hamming, BCH, RS, ...) sont étudiés dans [34]. Initialement inventés par Tanner [15], ils n'ont pas fait l'objet de nombreuses études avant 1999 [58]. L'application des codes LDPC généralisés à l'optique, envisagée pour des codes construits de manière aléatoire ou déterministe a montré des résultats satisfaisants en terme de gain de codage [34].

Enfin, des résultats récents [31] combinant les codes LDPC avec une modulation DPSK

M-aire ainsi qu'un entrelacement ont confirmé l'intérêt de l'application des codes LDPC à l'optique puisqu'une liaison fonctionnant à 100 Gb/s a été envisagée.

Les codes LDPC représentent donc en réalité une très large famille de codes, issus de différentes méthodes de construction et dont les performances en terme de gain de codage se sont révélées très intéressantes pour les transmissions optiques. L'algorithme de décodage, utilisant une méthode itérative et des échanges de valeurs issues de calculs simples (sommés et produits) est particulièrement bien adapté à l'application sur des circuits électroniques ayant un temps de calcul très rapide, et donc aux transmissions optiques haut débits.

Au niveau des différentes méthodes de construction, les méthodes déterministes ont été retenues pour leur application à l'optique pour des raisons d'architecture du décodeur. Les résultats en terme de gain de codage étant tout à fait comparables entre ces différentes méthodes déterministes, nous nous sommes focalisés par la suite sur l'une d'entre elle : la méthode issue de la technique BIBD. En effet, d'une part, la structure quasi cyclique est intéressante pour l'implémentation de codes de tailles importantes, et d'autre part, la possibilité de supprimer les cycles d'ordre 6 apparaît comme un atout pour une éventuelle amélioration du système.

1.5 Conclusion

Au cours de ce chapitre, les notions de base sur les codes correcteurs d'erreurs ont été présentées. Les codes en blocs sont mieux adaptés aux communications optiques que les codes convolutifs qui imposent des contraintes de délai. Parmi les codes en blocs, les deux familles de codes RS et LDPC ont été plus particulièrement présentées. Afin d'évaluer les performances, ces deux derniers types de codes sont appliqués par la suite aux

1.5 Conclusion

communications optiques dans deux cas d'études différents.

Le premier cas, étudié dans le chapitre suivant, repose sur l'insertion de codes correcteurs dans une communication sur fibre optique dans le cadre d'un réseau à topologie anneau proposé par l'ANR ECOFRAME (Eléments de CONvergence pour les Futurs Réseaux d'Accès et METropolitains haut débit) [59]. Ce réseau utilise une technique d'accès WDM, et on suppose que les interférences entre longueurs d'ondes sont inexistantes. Cependant, dans ce cas, ce sont les éléments optiques et opto-électroniques présents le long de la chaîne de transmission qui apportent des imperfections sur le canal et donc qui sont sources d'erreurs de transmission. L'objectif de cette étude est de fournir une solution électrique à base de codes correcteurs d'erreurs permettant de relâcher les contraintes sur les éléments optiques du réseau et de diminuer ainsi le coût global du réseau.

Une autre application du codage correcteur d'erreurs aux communications optiques est présentée au cours du dernier chapitre, dans le cadre de l'accès multiple par répartition de codes. Dans ce cas, et contrairement au cas précédent, des erreurs de transmission sont inévitablement dues à l'accès multiple. Ainsi, le canal est fortement perturbé par ces interférences et possède une statistique différente du canal considéré dans le réseau ECOFRAME. L'étude d'une solution utilisant le codage correcteur d'erreurs à l'aide des codes RS et LDPC est réalisée dans ce contexte. Grâce à la réduction significative du Taux d'Erreur Binaire (TEB), cette solution rend envisageable l'utilisation de l'accès multiple par répartition de code. De plus, une application utilisant le codage correcteur est proposée dans un système optique à répartition de code pour différencier à la fois les utilisateurs mais aussi les services, c'est-à-dire proposer des débits et des TEB différents à différents utilisateurs.

Chapitre 2

Codage correcteur d'erreurs pour les transmissions sur fibre optique -

Application au réseau métro/accès

ECOFRAME

Sommaire

2.1	Introduction	61
2.2	Chaîne de transmission optique	62
2.2.1	Emetteur	62
2.2.2	Fibre optique	65
2.2.3	Amplificateurs	66
2.2.4	Circuit de réception	66
2.2.5	Conclusions	68
2.3	Modèles de canal	69
2.3.1	Modèle AWGN	69
2.3.2	Modèle χ^2	72
2.4	Performances avec un modèle AWGN	86
2.4.1	Introduction	86

2.4.2	Performances dans le cas non codé	86
2.4.3	Performances avec les codes RS	90
2.4.4	Performances avec les codes LDPC	95
2.4.5	Conclusions	103
2.5	Performances avec le modèle χ^2	104
2.5.1	Cas non codé	104
2.5.2	Performances avec les codes RS	111
2.5.3	Performances avec les codes LDPC-BIBD	115
2.5.4	Conclusions	121
2.6	Conclusion	123

2.1 Introduction

Les réseaux de communication utilisant la fibre optique, maintenant largement déployés, font encore l'objet de nombreuses recherches. L'objectif commun de ces études est d'améliorer le système de transmission pour le rendre non seulement plus performant en termes de débit et de Taux d'Erreur Binaire (TEB) mais aussi moins coûteux.

En effet, d'une manière générale, lors de son parcours le long d'une fibre optique, le signal lumineux est soumis principalement aux phénomènes d'atténuation, de dispersion (chromatique et de polarisation), et aux effets non linéaires (Kerr, Raman-Brillouin) [60]. Tous ces phénomènes déforment les impulsions lumineuses transmises, qui à l'arrivée, ont perdu de la puissance et sont étalées dans le temps. En fait, pour un réseau d'accès, limité à de courtes distances ces phénomènes n'ont pas d'impact significatif sur les erreurs en détection [61]. Cependant, dès qu'on considère des distances un peu plus importantes, comme dans le cadre d'un réseau métro-accès par exemple, ils peuvent avoir un impact sur le signal transmis et limiter les performances. Sur de très longues distances (pouvant aller jusqu'à 1000 km), ces perturbations ont un impact important puisque les impulsions successives dans le temps peuvent interférer entre elles.

De plus, chacun des éléments optiques et opto-électroniques composant la chaîne de transmission constitue une source de bruit par rapport au signal utile transmis. Lors de la détection, des erreurs apparaissent et sont d'autant plus nombreuses que le signal utile est perturbé c'est-à-dire que le canal de transmission est bruité.

Les effets de ces différentes perturbations peuvent être limités en utilisant des composants précis et peu générateurs de bruits. Cependant, cette solution est coûteuse et d'autres voies sont recherchées. Une solution pour compenser la dégradation de performances due au canal de transmission consiste à insérer un système de codage correcteur d'erreurs. Toutes les opérations de codage et décodage s'effectuent plus

facilement dans le domaine électrique et peuvent ainsi être réalisées de manière peu coûteuse.

De plus, de récentes études ont permis de montrer la capacité des circuits électroniques de type DSP (Digital Signal Processing) ou FPGA (Field Programmable Gate Array) à assurer ce genre d'opérations [55]. Le bloc de codage se situe alors à l'émission avant la conversion électro-optique tandis que le bloc de décodage se situe après la conversion en réception, mais avant la prise de décision.

Les caractéristiques de chacun des éléments de la chaîne de transmission que nous considérons comme ayant un impact sur le canal sont décrits dans le paragraphe suivant. Pour rendre compte des imperfections dues aux éléments de la chaîne de transmission, deux modèles sont envisagés par la suite pour modéliser le canal optique : le modèle classique additif gaussien AWGN (Additive White Gaussian Noise) et un modèle à statistique χ^2 (aussi noté *chi* - 2). L'étude des performances est réalisée en présence de codage correcteur d'erreurs de type RS et LDPC sur chacun des deux canaux. Une application du codage correcteur d'erreurs est illustrée dans le contexte du projet ANR ECOFRAME (Eléments de CONvergence pour les Futurs Réseaux d'Accès et METropolitains haut débit) [59] visant à assurer un débit de transmission élevé tout en apportant une correction d'un TEB de 10^{-5} à 10^{-12} .

2.2 Chaîne de transmission optique

2.2.1 Emetteur

On considère que le canal de propagation est un canal unipolaire, basé sur la puissance optique, et que les données émises ('0' ou '1') sont équiprobables. De plus, la modulation la plus simple est envisagée, c'est-à-dire une modulation On Off Keying (OOK).

2.2 Chaîne de transmission optique

Au niveau de l'émission, la conversion électro-optique des données à émettre sur la fibre peut être réalisée par un laser à semi-conducteur (appelé aussi diode laser) ou une diode LED (Light Emitting Diode). Les lasers possèdent une meilleure efficacité et fiabilité, ainsi qu'une faible surface d'émission de la lumière. Cette dernière caractéristique est adaptée aux communications par fibre optique dont le diamètre est très étroit (de l'ordre de $60\mu m$). De plus, les lasers sont capables d'avoir un spectre d'émission plus étroit que les LED. Or, dans le cadre de transmissions WDM, les spectres d'émissions doivent être étroits pour qu'il n'y ait pas d'interférence entre longueurs d'ondes. L'étude se ramène alors au cas d'une transmission mono-utilisateur mettant en oeuvre une seule longueur d'onde.

Les sources lasers envisagées pour l'optique sont des sources dites cohérentes, dont le temps de cohérence est inversement proportionnel à la largeur du spectre d'émission de la source [62]. Pour une source purement monochromatique, le temps de cohérence est théoriquement infini puisqu'il n'y a qu'une composante spectrale. Une source dite cohérente possède donc un spectre d'émission étroit et un grand temps de cohérence. Cependant, cette caractéristique peut varier d'une source à l'autre et, selon le contexte envisagé, une source cohérente peut finalement avoir un temps de cohérence peu important devant le temps bit. La cohérence de la source a un impact sur la manière dont se combinent plusieurs champs optiques différents au niveau du récepteur. Cependant, dans le cadre d'une transmission mono-utilisateur, la cohérence n'aura pas d'impact puisqu'un seul champ optique est incident au niveau du détecteur.

Chaque laser est caractérisé par un courant de seuil I_{seuil} (figure 2.1). Au dessus de cette valeur, l'émission est dite stimulée et la puissance de la lumière émise est proportionnelle au courant d'entrée. De part la structure physique du laser, en dessous du courant de seuil, une faible émission de lumière a lieu. Ce phénomène inévitable dû à la génération aléatoire de photons s'appelle l'émission spontanée (Spontaneous Emission).

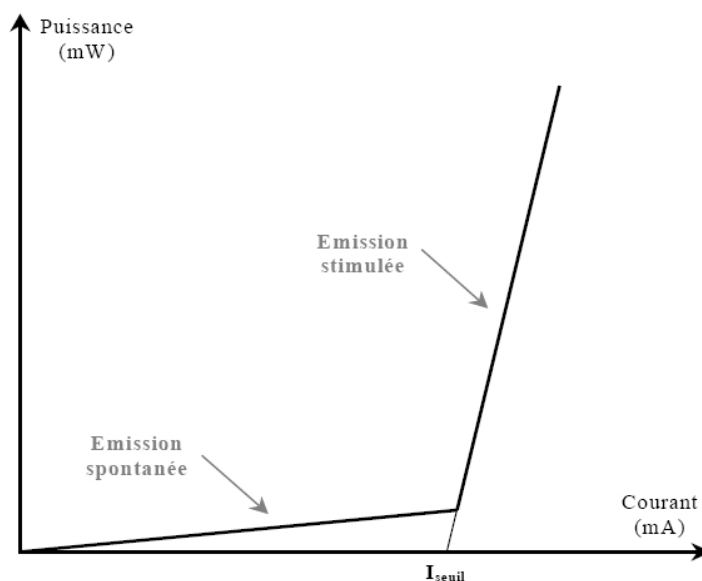


FIG. 2.1 – Caractéristique Puissance - Courant du laser

L'émission spontanée génère un bruit d'amplitude (ou RIN : Relative Intensity Noise) ainsi qu'un bruit de phase. Le bruit d'amplitude, (défini en fait comme un bruit de puissance) se traduit par des fluctuations de la puissance optique autour de sa valeur moyenne. Le bruit de phase est lié à la largeur de la raie, traduisant le fait que le laser n'est pas parfaitement monochromatique. D'autres phénomènes indirectement liés à l'émission spontanée tels que des fluctuations de phase ou de fréquence instantanée (effet de "chirp") peuvent aussi perturber le signal transmis. L'étude réalisée dans cette thèse se limite au bruit d'émission spontanée sans considérer les bruits qui en découlent.

La modulation directe des données, c'est-à-dire lorsque les données à émettre constituent le courant d'entrée, correspond au cas où la zone d'émission stimulée est utilisée pour faire varier la puissance lumineuse émise en fonction des données. Pour de hauts débits (supérieurs à 2.5 Gb/s), une autre solution est la modulation externe où dans ce cas, le laser émet en permanence un signal optique, qui est modulé en sortie par un modulateur externe reposant sur le principe d'absorption de la lumière.

2.2 Chaîne de transmission optique

Dans les deux cas, la puissance émise pour une donnée '0' n'est jamais parfaitement nulle. En effet, l'émission spontanée existe lors de la modulation directe, et en modulation externe, l'absorption de la lumière pour un '0' émis n'est jamais totale. L'émission est alors caractérisée par un rapport d'extinction en entrée ER_{in} , correspondant au rapport des puissances optiques d'émission pour un '1' et pour un '0' :

$$ER_{in} = \frac{A_1^2}{A_0^2} \quad (2.1)$$

Ce rapport, généralement exprimé en dB ($ER_{in}(dB) = 10 \cdot \log(ER_{in})$) est une donnée caractéristique du système. Dans le cas idéal où il n'y a pas d'émission spontanée ou bien si le modulateur externe fonctionne de manière parfaite, $A_0 = 0$ et le rapport ER_{in} est infini.

Par la suite, sans préciser le type de modulation (directe ou externe), nous considérerons que l'émission est caractérisée par un rapport ER_{in} .

2.2.2 Fibre optique

Lors de sa propagation le long de la fibre, le signal subit des dégradations dues à l'atténuation, la dispersion, et aux effets non linéaires. L'atténuation correspond à la réduction de la puissance moyenne du signal. Dans un réseau où l'atténuation peut avoir une influence importante, elle est souvent compensée par l'insertion d'amplificateurs, qui comme nous le verrons dans le paragraphe suivant perturbent eux aussi le signal transmis. Les effets de dispersion (chromatique et de polarisation) conduisent à un étalement temporel des impulsions transmises le long de la fibre. Pour compenser ces effets de dispersion, on utilise généralement des fibres compensatrices DCF (Dispersion Compensating Fiber) qui possèdent une dispersion négative. Les effets non-linéaires

principaux sont l'effet Kerr et l'effet Raman-Brillouin qui se traduisent par une dépendance entre la phase du signal et sa puissance et conduisent ainsi à une auto-modulation de phase. Ces effets sont peu importants lorsque la puissance injectée est faible.

Dans l'étude réalisée, nous considérerons la fibre optique comme idéale ce qui signifie que l'impact de ces effets (atténuation, dispersion, non-linéarités) sur la nature de la statistique du signal transmis est négligé. Une étude plus précise pourrait néanmoins mettre en relation ces effets avec les paramètres de la statistique envisagée.

2.2.3 Amplificateurs

Dans les canaux optiques, le bruit d'émission spontanée dû au laser, bien que faible à l'émission, peut se retrouver fortement amplifié de part l'utilisation d'amplificateurs optiques sur la liaison (SOA : Semiconductor Optical Amplifier, EDFA : Erbium-Doped Fiber Amplifiers, ou Raman). De plus, chaque amplificateur optique amplifie non seulement le bruit présent en amont, mais génère aussi une émission spontanée de lumière [60]. Ce bruit global nommé bruit d'émission spontanée amplifié (ASE : Amplified Spontaneous Emission) est indépendant du signal utile. Comme sa densité spectrale de puissance est pratiquement constante [60], il est assimilé par la suite à un bruit blanc additif gaussien (AWGN).

2.2.4 Circuit de réception

En ce qui concerne la réception, on considère une détection directe réalisée à l'aide d'un photodétecteur. Le circuit de réception peut être décomposé en trois parties : la détection comportant le photodétecteur suivi d'un préamplificateur, l'amplification, et la prise de décision.

2.2 Chaîne de transmission optique

Au niveau de la détection, les photodétecteurs les plus couramment employés dans les systèmes de communication par fibre optique sont les photodiodes PIN (PIN signifiant qu'elle est composée d'une jonction P-N où est insérée une couche peu ou non dopée constituant ainsi le matériau intrinsèque noté I) et les photodiodes APD (photodiodes à avalanche ou *Avalanche PhotoDiode*) [60]. Chaque photodiode est caractérisée par sa sensibilité S , exprimée en Ampères par Watt, définissant le rapport du photocourant (courant généré par la photodiode) I_{ph} sur la puissance optique P_{opt} incidente. Le courant I_{ph} est donc proportionnel à la puissance lumineuse, ce qui signifie que la puissance du signal électrique est proportionnelle au carré de la puissance optique. Il s'agit donc d'une détection quadratique dont la loi de détection s'écrit de la manière suivante :

$$I_{ph} = S \cdot P_{opt} = S \cdot |E(t)|^2 \quad (2.2)$$

avec S la sensibilité de la photodiode, et $E(t)$ le champ optique incident.

Le signal est ensuite amplifié de manière supposée linéaire sur la plage d'utilisation de l'amplificateur avant d'être intégré sur le temps bit.

En notant le temps bit $T = 1/B_e$ avec B_e la bande passante électrique de la photodiode, la variable décisionnelle est donc obtenue comme suit :

$$Y = \frac{1}{T} \int_T I_{ph} dt = \frac{1}{T} \int_T S |E(t)|^2 dt \quad (2.3)$$

La variable décisionnelle est finalement comparée à un seuil de décision pour déterminer si le signal reçu correspond à un '1' ou un '0' initialement émis.

Les éléments électroniques présents dans le circuit d'amplification et dans celui de décision génèrent deux sortes de bruits : le bruit thermique et le bruit de grenaille appelé plus couramment "shot noise", qui ont pour conséquence une fluctuation du photocourant

autour de sa valeur moyenne [60]. Le bruit thermique généré par l'agitation aléatoire des électrons sous l'action de la chaleur est principalement provoqué par les résistances du circuit de réception et peut être assimilé à un processus gaussien. Le "shot noise" est lié à la nature discrète du champ optique qui génère ainsi une succession discrète et aléatoire d'électrons dans le photocourant. Ce bruit a une statistique poissonnienne, qui peut être également approchée par une loi gaussienne [60].

Dans les cas étudiés, on suppose que ces deux bruits sont négligeables par rapport aux bruits présents avant détection. En effet, ils peuvent être minimisés en faisant un choix judicieux de la photodiode et des composants électroniques du circuit de décision.

2.2.5 Conclusions

Etant donné l'ensemble des perturbations générées par les différents éléments opto-électroniques, le canal optique est fortement bruité, et le signal utile détecté est toujours dégradé. Le modèle le plus simple pour caractériser le signal reçu est un modèle considérant l'ensemble des perturbations avant détection comme un bruit blanc additif gaussien. Ce modèle est réaliste si on considère l'ASE prédominant. Or, la conversion opto-électronique suivant une loi quadratique engendre des battements ("beatings") entre les différentes composantes de bruit et du signal mais aussi entre celles du bruit lui-même. Ainsi, la distribution de la variable de décision suit une loi particulière non gaussienne. Puisqu'on considère que le bruit global avant détection est de nature additive gaussienne, les battements causés par la détection quadratique font apparaître des variables gaussiennes élevées au carré, dont la moyenne dépend du signal émis. Par définition, la variable résultante possède une distribution de type χ^2 , centrée ou non-centrée en fonction des données émises.

Pour caractériser l'ensemble du canal optique, on considèrera donc dans un premier

temps un modèle simple de type AWGN (Additive White Gaussian Noise) puis un modèle à statistique χ^2 , tous deux présentés dans les paragraphes suivants.

2.3 Modèles de canal

2.3.1 Modèle AWGN

2.3.1.1 Définition et modélisation du canal

On considère dans un premier temps que le canal est AWGN, c'est-à-dire que le signal reçu au niveau de la prise de décision est constitué du signal émis auquel est ajouté un bruit blanc gaussien. Ce bruit modélise de manière la plus simple possible l'ensemble des bruits perturbant le signal, qu'ils proviennent de l'émission, de la transmission ou de la détection. Le bruit global est entièrement caractérisé par sa variance obtenue comme la somme des variances des différents bruits supposés tous gaussiens et indépendants (bruit d'émission spontanée, bruit thermique, bruit de battement entre le signal et le bruit, bruit de battement entre le bruit et lui-même,...) [60].

Au niveau de la prise de décision, la variable décisionnelle Y est directement égale à $Y = r = e + n$, où e correspond au signal utile et n est une variable de bruit suivant une distribution gaussienne, de moyenne nulle et de variance σ^2 . Ainsi, la variable décisionnelle Y suit la loi normale, de variance σ^2 et de moyenne $m_Y = e$ dépendant du bit émis $i = 0$ ou 1. La figure 2.2 représente la modélisation du canal.

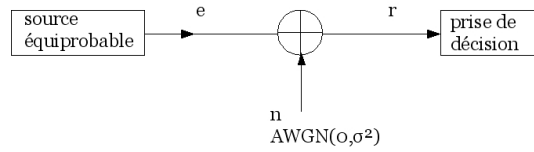


FIG. 2.2 – Modélisation du canal AWGN

2.3.1.2 Caractéristiques mathématiques

La densité de probabilité d'une variable X à statistique gaussienne, de moyenne nulle et de variance σ^2 s'exprime de la manière suivante [63] :

$$p_X(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (2.4)$$

Si la moyenne de la variable gaussienne X de variance σ^2 est non nulle ($m = \langle X \rangle \neq 0$), la distribution est normale non-centrée. La densité de probabilité de X est donnée par :

$$p_X(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m)^2}{2\sigma^2}} \quad (2.5)$$

La figure 2.3 représente un exemple de densité de probabilité d'une variable suivant la loi gaussienne non-centrée ($m = 0.5$) et de variance $\sigma^2 = 0.1$. Une telle densité de probabilité est symétrique, centrée en $m = 0.5$. La largeur de la gaussienne est d'autant plus grande que σ augmente.

2.3.1.3 Paramètres de simulation

Les performances sur canal AWGN sont généralement évaluées en fonction du rapport signal à bruit S/B exprimé en dB. Le S/B dépend directement de la variance du bruit global σ^2 et de la puissance utile moyenne par bit transmis.

Par la suite, les simulations numériques ne tiennent pas compte de la notion de temps

2.3 Modèles de canal

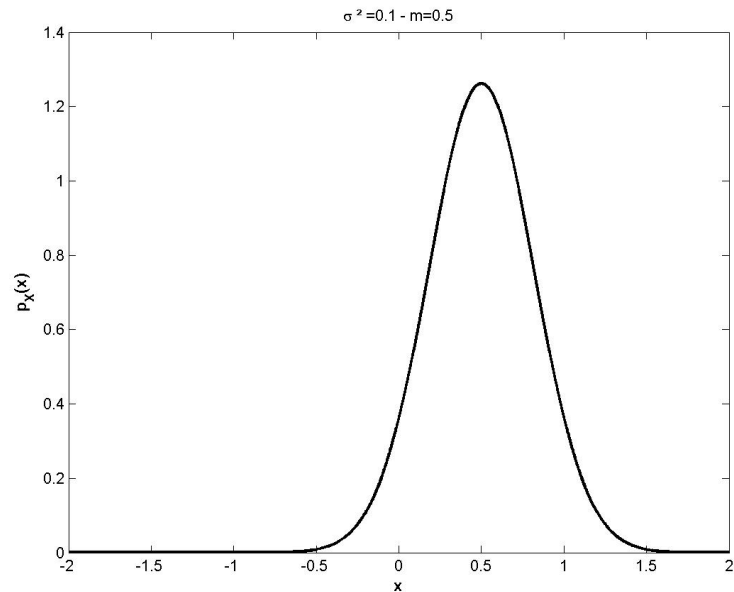


FIG. 2.3 – Densité de probabilité $p_X(x)$ d'une loi gaussienne avec $\sigma^2 = 0.1$ et $m = 0.5$

puisque le signal émis est représenté comme une succession de bits. Cela revient à effectuer une normalisation, de telle sorte que la puissance contenue dans le bit soit égale à l'énergie de celui-ci.

Par définition, si E_s est l'énergie moyenne du signal par bit et E_n l'énergie moyenne du bruit par bit, on a :

$$S/B(dB) = 10 \cdot \log \left(\frac{E_s}{E_n} \right) \quad (2.6)$$

Si on note A l'amplitude du signal utile correspondant à un '1' émis et si on considère que pour un '0' émis l'amplitude vaut 0, l'énergie utile moyenne par bit transmis est :

$$E_s = p(i = 0) \cdot 0^2 + p(i = 1) \cdot A^2 \quad (2.7)$$

En considérant une émission équiprobable de '0' et de '1', (c'est-à-dire $p(i = 0) = p(i = 1) = 1/2$), l'énergie utile moyenne par bit transmis dans le cas non codé est alors :

$E_s = A^2/2$. De plus, comme on considère une transmission en bande de base, la densité spectrale de puissance du bruit est $\sigma^2 = N_0/2$ d'où $E_n = N_0 = 2 \cdot \sigma^2$.

Le S/B s'exprime alors en fonction de la variance par :

$$S/B(dB) = 10 \cdot \log \left(\frac{A^2}{4\sigma^2} \right) \quad (2.8)$$

Pour le cas codé, il faut prendre en compte l'ajout des $N - K$ bits de redondance dans le calcul de l'énergie utile moyenne par bit transmis. En effet, l'énergie moyenne émise pour un paquet de taille N est de $A^2N/2$. Sachant que parmi les N bits, seuls K font partie des données utiles, l'énergie moyenne par bit utile est alors égale à : $A^2N/(2K)$. Le rapport S/B dans le cas codé devient :

$$S/B(dB) = 10 \cdot \log \left(\frac{A^2N}{4K\sigma^2} \right) \quad (2.9)$$

Ainsi, en simulation, il faudra choisir le rapport signal à bruit du système (codé ou non codé). La variance du bruit AWGN sur la liaison est ensuite obtenue à partir de l'équation (2.8) pour le cas non-codé ou (2.9) pour le cas codé. Avant d'étudier les performances sur canal AWGN, un canal plus réaliste dont la statistique est basée sur un modèle de type χ^2 est présenté.

2.3.2 Modèle χ^2

2.3.2.1 Justification du modèle

La détection quadratique, de part l'élévation au carré des différentes composantes du signal reçu, modifie la statistique du signal. Le modèle à statistique χ^2 traduit par définition la somme de carrés de variables à statistique gaussienne.

2.3 Modèles de canal

Dans le modèle χ^2 développé, on considère que le bruit global avant la photodétection est AWGN. Le signal se propageant sur la liaison $r(t)$ est donc tel que :

$$r(t) = e(t) + n(t) \quad (2.10)$$

où $e(t)$ correspond au signal émis et $n(t)$ au bruit AWGN.

En utilisant la représentation complexe des signaux, et en notant ω_c la pulsation correspondant au signal utile, le signal $e(t)$ peut s'écrire, pour t défini sur un intervalle de durée égale au temps bit T comme :

$$e(t) = A \cdot e^{j\omega_c t} \quad (2.11)$$

A représente l'amplitude du champ optique correspondant au signal utile. Cette valeur est réelle et supposée constante pour un '1' ou un '0' émis durant le temps bit T .

Le bruit AWGN $n(t)$ (principalement dû à l'ASE comme nous l'avons vu précédemment) peut être représenté par sa notation complexe et donc décomposé en série de Fourier pour t défini sur un intervalle de durée T :

$$n(t) = \sum_{k=0}^{\infty} n_k \cdot e^{j\omega_k t} \quad (2.12)$$

avec $\omega_k = \frac{2\pi \cdot k}{T}$ les pulsations contenues dans la décomposition. Dans cette notation, les n_k sont des variables gaussiennes centrées complexes, toutes indépendantes entre elles. Les termes $e^{j\omega_k t}$ de la série de Fourier forment une base orthonormale par rapport au temps, et sont donc tels que :

$$\int_T e^{j\omega_k t} \cdot e^{j\omega_l t} dt = \delta_{kl} \quad (2.13)$$

avec δ_{kl} la fonction de Kronecker qui vaut 1 si $k = l$ et 0 sinon.

Avant d' tre re u par le photod tecteur, on consid re que le signal passe par diff rents amplificateurs optiques et  ventuellement par un filtre optique en r ception. Les amplificateurs optiques poss dent une bande passante limit e et ne r g n rent le signal que dans cette bande. Soit B_o la bande passante la plus  troite entre celle impos e par les amplificateurs ou le filtre, on note f_1 et f_2 , les fr quences de coupures basse et haute de cette bande passante. De plus, on suppose que B_o est un multiple de la bande passante  lectrique $B_e = 1/T$. Avec M entier naturel, on a alors :

$$B_o = f_2 - f_1 = MB_e = \frac{M}{T} \quad (2.14)$$

Le signal en sortie du filtre optique (ou de la cha ne optique amplifi e) s' crit finalement de la mani re suivante sur un intervalle de dur e T :

$$r_f(t) = e(t) + \sum_{k=k_1}^{k_1+M} n_k \cdot e^{j\omega_k t} \quad (2.15)$$

avec k_1 indice qui correspond   la fr quence de coupure $f_1 = k_1/T$.

L' quation (2.15) correspond   l'expression du signal juste avant le photod tecteur. Apr s la loi quadratique de la photod tection, en posant la sensibilit  de la photodiode S  gale   1, le photocourant r sultant $I_{ph}(t)$ est obtenu comme le module au carr  du signal re u $r_f(t)$ selon (2.2) :

$$I_{ph}(t) = |r_f(t)|^2 = \left| A \cdot e^{j\omega_c t} + \sum_{k=k_1}^{k_1+M} n_k \cdot e^{j\omega_k t} \right|^2 \quad (2.16)$$

Le photocourant est ensuite int gr  sur le temps bit T de mani re   constituer la variable

2.3 Modèles de canal

décisionnelle Y selon (2.3) :

$$\begin{aligned}
 Y &= \frac{1}{T} \int_0^T I_{ph}(t) dt \\
 &= \frac{1}{T} \int_0^T \left| A \cdot e^{j\omega_c t} + \sum_{k=k_1}^{k_1+M} n_k \cdot e^{j\omega_k t} \right|^2 dt
 \end{aligned} \tag{2.17}$$

En développant les termes, et de part la linéarité de l'intégrale, il vient :

$$\begin{aligned}
 Y &= \frac{1}{T} \int_0^T A^2 dt + \frac{1}{T} \int_0^T \sum_{k=k_1}^{k_1+M} |n_k|^2 dt \\
 &+ \frac{1}{T} \int_0^T A \cdot e^{j\omega_c t} \sum_{k=k_1}^{k_1+M} n_k^* \cdot e^{-j\omega_k t} dt + \frac{1}{T} \int_0^T A \cdot e^{-j\omega_c t} \sum_{k=k_1}^{k_1+M} n_k \cdot e^{j\omega_k t} dt
 \end{aligned} \tag{2.18}$$

Etant donné que les $e^{j\omega_k t}$ forment une base orthonormale, seuls les termes pour $k = c$ subsistent dans les deux dernières sommes après l'intégration, et l'expression de Y se résume à :

$$Y = A^2 + \sum_{k=k_1}^{k_1+M} |n_k|^2 + A(n_c^* + n_c) \tag{2.19}$$

En notant $n_k^{(r)}$ et $n_k^{(i)}$ les parties réelles et imaginaires des composantes de bruit n_k , il vient :

$$Y = A^2 + \sum_{k=k_1}^{k_1+M} \left(n_k^{(r)2} + n_k^{(i)2} \right) + 2A \cdot n_c^{(r)} \tag{2.20}$$

Dans l'équation (2.20), trois termes apparaissent : le premier correspond au signal utile, le deuxième au battement bruit/bruit provoqué par la détection quadratique, et le dernier au battement signal/bruit. Ces deux derniers termes peuvent être considérés comme un bruit de battement ou "beat noise". Cependant, l'appellation beat noise est généralement préférée pour désigner des battements entre la donnée désirée et les données de signaux interférents.

S'il s'agit d'un '0'  mis avec une amplitude A du champ correspondant exactement nulle, seul le terme ind pendant du signal reste, c'est- -dire $\sum_{k=k_1}^{k_1+M} (n_k^{(r)^2} + n_k^{(i)^2})$ repr sentant le battement bruit-bruit. Sachant que chaque composante de bruit est une variable gaussienne centr e, cette somme correspond   une somme de $2M$ variables gaussiennes de moyennes nulles,  lev es au carr  donc par d finition   une variable χ^2 centr e d'ordre $2M$.

Dans le cas g n ral, s'il s'agit d'un '1'  mis ou si on consid re que la puissance correspondant   un '0' n'est pas exactement nulle ($A \neq 0$), on peut  crire l'expression (2.20) de la mani re suivante :

$$Y = \sum_{k=k_1}^{k_1+M} \left(n_k^{(r)} + \frac{A}{\sqrt{2M}} \right)^2 + \left(n_k^{(i)} + \frac{A}{\sqrt{2M}} \right)^2 \quad (2.21)$$

Dans cette expression, on voit que la variable d cisionnelle s' crit comme une somme de $2M$ variables gaussiennes, de variances  gales   celles des composantes de bruit $n_k^{(r)}$ et $n_k^{(i)}$ et de moyennes  gales   $m_k = \frac{A}{\sqrt{2M}}$, donc toutes identiques mais d pendantes de A donc du signal  mis.

Par d finition, la variable d cisionnelle a alors une statistique de type χ^2 non centr e d'ordre $2M$ (nombre de termes de la somme). Le param tre de non-centralit  s^2 est d fini d'une mani re g n rale comme la somme des moyennes au carr  des diff rentes variables :

$$s^2 = \sum_{k=1}^{2M} m_k^2 \quad (2.22)$$

Dans le cas consid r , le param tre de non-centralit  est directement  gal   : $s^2 = A^2$.

Finalement, au niveau du circuit de d cision, la variable re ue poss de ainsi dans tous les cas une statistique χ^2 , centr e ou non suivant les caract ristiques du signal  mis. L'ordre de la distribution est donn  par $n = 2M = 2B_o/B_e$ avec B_o et B_e les bandes

2.3 Modèles de canal

passantes optique et électrique du circuit de décision.

S'il s'agit d'un '1' ou d'un '0' émis, les composantes gaussiennes constituant la variable décisionnelle ont une variance identique mais des moyennes différentes, ce qui établit que le canal est asymétrique.

Ces résultats sont en accord avec ceux exposés par [64] qui énonce que dans le contexte optique, le rapport entre la bande passante optique et la bande passante électrique (ici noté M) permet de définir la dimension n de l'espace complexe contenant le signal transmis : $n = 2M$.

De plus, nous verrons par la suite que le modèle χ^2 se justifie lorsque des simulations physiques sont effectuées pour obtenir la statistique du canal ECOFRAME.

2.3.2.2 Caractéristiques mathématiques

Densité de probabilité

Soit Y une variable suivant une distribution de type χ^2 non centrée d'ordre n :

$$Y = \sum_{k=1}^n X_k^2. \quad (2.23)$$

Les $X_k, k = 1, 2, \dots, n$ sont des variables aléatoires gaussiennes de variance σ^2 et de moyennes respectives $\langle X_k \rangle = m_k$.

Si les moyennes ne sont pas nulles, le paramètre de non centralité de la distribution est défini par :

$$s^2 = \sum_{k=1}^n m_k^2. \quad (2.24)$$

La densité de probabilité de la variable Y à statistique χ^2 non centrée est nulle pour $y < 0$,

et s'exprime, pour $y \geq 0$ [63] :

$$p_Y(y) = \frac{1}{2\sigma^2} \left(\frac{y}{s^2}\right)^{(n-2)/4} e^{-(s^2+y)/(2\sigma^2)} I_{n/2-1} \left(\sqrt{y} \frac{s}{\sigma^2}\right) \quad (2.25)$$

O  $I_\alpha(x)$ repr sente la fonction de Bessel de premi re esp ce d'ordre α exprim e par :

$$I_\alpha(x) = \sum_{k=0}^{\infty} \frac{(x/2)^{\alpha+2k}}{k! \Gamma(\alpha + k + 1)}, x \geq 0 \quad (2.26)$$

et o  $\Gamma(p)$ est la fonction Gamma, d finie pour p entier diff rent de 0, par :

$$\Gamma(p) = (p - 1)! \quad (2.27)$$

Si les moyennes sont toutes nulles, alors $s^2 = 0$ et la variable Y a une statistique χ^2 centr e. Pour obtenir la densit  de probabilit  de Y , on ne peut pas remplacer directement $s = 0$ dans l'expression pr c dente puisque s est au d nominateur. En rempla ant la fonction de Bessel par son expression (2.26), il vient :

$$p_Y(y) = \frac{1}{2\sigma^2} \frac{y^{(n-2)/4}}{s^{(n-2)/2}} e^{-(s^2+y)/(2\sigma^2)} \sum_{k=0}^{\infty} \frac{\left(\sqrt{y} \frac{s}{2\sigma^2}\right)^{n/2-1+2k}}{k! \Gamma(n/2 + k)} \quad (2.28)$$

D'o  en rassemblant les termes relatifs   s :

$$p_Y(y) = \frac{1}{2\sigma^2} y^{(n-2)/4} e^{-(s^2+y)/(2\sigma^2)} \sum_{k=0}^{\infty} \left(\frac{\sqrt{y}}{2\sigma^2}\right)^{n/2-1+2k} s^{2k} \frac{1}{k! \Gamma(n/2 + k)}. \quad (2.29)$$

En  valuant cette expression quand $s = 0$, tous les termes de la somme sont nuls, sauf pour $k = 0$, et il vient :

$$p_Y(y) = \frac{1}{2\sigma^2} y^{(n-2)/4} e^{-y/(2\sigma^2)} \left(\frac{\sqrt{y}}{2\sigma^2}\right)^{n/2-1} \frac{1}{\Gamma(n/2)}. \quad (2.30)$$

2.3 Modèles de canal

La densité de probabilité pour une variable suivant une statistique χ^2 centrée s'exprime finalement de la manière suivante, pour $y \geq 0$:

$$p_Y(y) = \frac{1}{2^{n/2}\sigma^n\Gamma(\frac{1}{2}n)}y^{n/2-1}e^{-y/(2\sigma^2)}. \quad (2.31)$$

Un exemple de densité de probabilité pour une variable suivant une statistique χ^2 non-centrée ayant pour paramètres $\sigma^2 = 0.05$ et $s^2 = 0.5$ est représenté sur la figure 2.4 pour $n=2, 8$ et 20 .

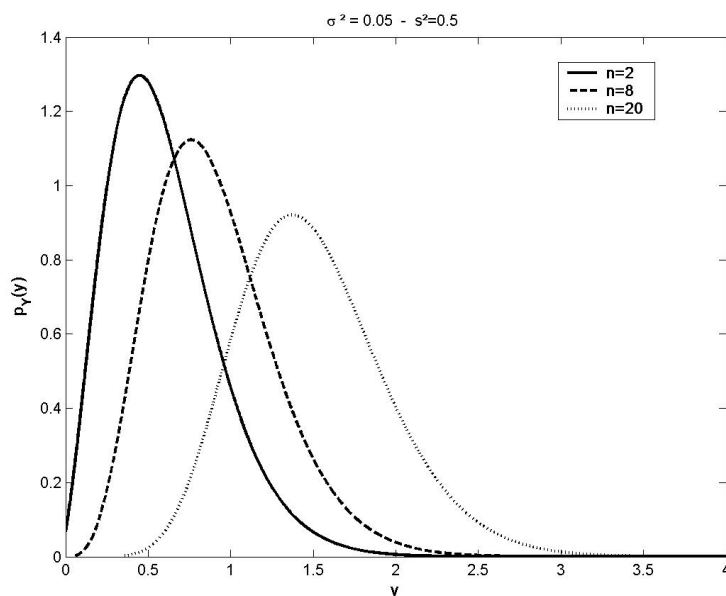


FIG. 2.4 – Densité de probabilité $p_Y(y)$ d'une variable à statistique χ^2 avec $\sigma^2 = 0.05$ et $s^2 = 0.5$

Pour les mêmes valeurs de n , un autre exemple représentant les densités de probabilités obtenues pour une variance plus élevée ($\sigma^2 = 0.1$ et $s^2 = 0.4$) sont représentées figure 2.5.

Ces différentes courbes montrent tout d'abord que pour une variance du bruit σ^2 fixée, plus l'ordre du χ^2 est élevé plus la courbe est décentrée vers la droite. En effet, comme

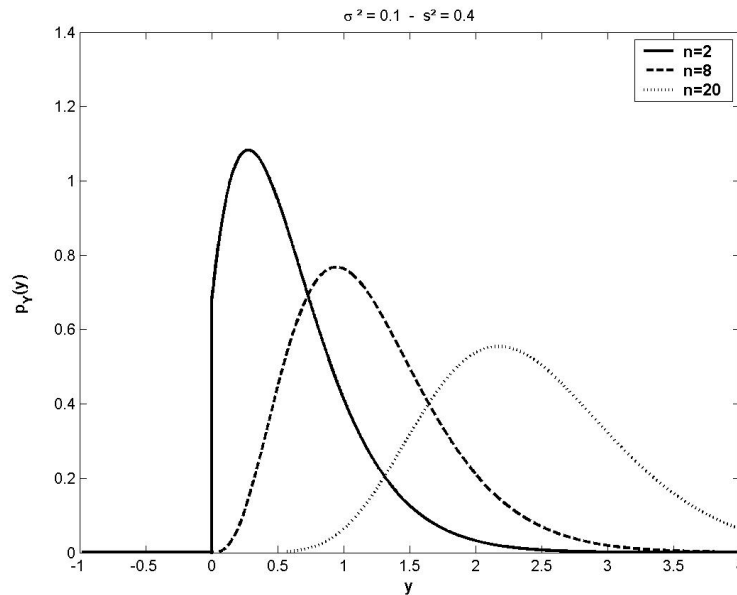


FIG. 2.5 – Densité de probabilité $p_Y(y)$ d'une variable à statistique χ^2 avec $\sigma^2 = 0.1$ et $s^2 = 0.4$

expliqué dans le paragraphe suivant, la moyenne de la variable χ^2 résultante augmente avec la variance du bruit σ^2 [63]. On remarque aussi que pour des valeurs élevées de n , la densité de probabilité se rapproche d'une loi gaussienne, même si on note dans tous les cas une dissymétrie plus ou moins importante (la partie droite de la "cloche" est plus haute que la gauche). Pour de faibles valeurs de n ($n = 2$ par exemple), la densité de probabilité est très éloignée d'une gaussienne puisque la dissymétrie augmente. Dans le cas où $\sigma^2 = 0.1$, la présence d'une importante discontinuité dans la fonction $p_Y(y)$ montre qu'il est impossible d'approximer cette densité de probabilité par une gaussienne. Ce phénomène, moins important dans le cas $\sigma^2 = 0.05$, doit toutefois être pris en compte lorsqu'on souhaite caractériser précisément le signal reçu.

Moyenne et variance de la variable χ^2

La moyenne et la variance de la variable résultante Y s'expriment à partir des moyennes

2.3 Modèles de canal

et variances des variables gaussiennes X_k composant Y par [63] :

$$m_Y = n\sigma^2 + s^2 \quad (2.32)$$

$$\sigma_Y^2 = 2n\sigma^4 + 4\sigma^2 s^2 \quad (2.33)$$

sachant que le paramètre de non-centralité s^2 , donné par l'équation (2.24) dépend des moyennes des X_k .

À partir de ces expressions, on retrouve le fait que pour une valeur fixée de σ^2 , la moyenne et la variance sont d'autant plus élevées que n est grand. Ainsi, pour comparer les statistiques des variables χ^2 dont l'ordre n diffère, il peut être intéressant de les tracer pour σ_Y^2 constant au lieu de σ^2 . En effet, dans ce cas, les statistiques peuvent toutes caractériser un phénomène de même variance σ_Y^2 .

Pour cela, à partir de (2.33), il faut déterminer l'expression de σ^2 en fonction de σ_Y^2 . Cela se traduit par la résolution d'une équation du second degré s'écrivant :

$$\sigma^4 + \frac{2s^2}{n}\sigma^2 - \frac{\sigma_Y^2}{2n} = 0 \quad (2.34)$$

La seule solution positive est donnée par :

$$\sigma^2 = \frac{\sqrt{\sigma_Y^2 + 2s^4/n}}{\sqrt{2n}} - \frac{s^2}{n} \quad (2.35)$$

Un exemple est présenté sur la figure 2.6 avec $\sigma_Y^2 = 0.5$ et $s^2 = 0.5$. Les variances des composantes gaussiennes X_k sont obtenues à partir de (2.35) et valent : $\sigma^2(n = 2) = 0.125$, $\sigma^2(n = 8) = 0.117$ et $\sigma^2(n = 20) = 0.0875$.

On remarque alors que les courbes sont toutes de largeurs comparables entre elles (la

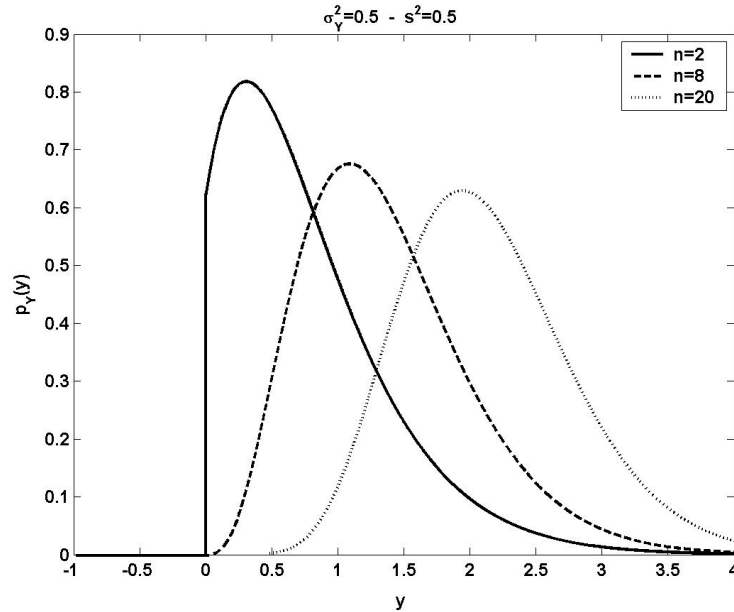


FIG. 2.6 – Densit  de probabilit  $p_Y(y)$ d’une variable   statistique χ^2 avec $\sigma_Y^2 = 0.5$ et $s^2 = 0.5$

variance r sultante σ_Y^2 est constante) et que seule la moyenne de Y d pend de n . De plus, pour de faibles valeurs de n , par exemple $n = 2$ on retrouve le ph nom ne de discontinuit  qui n’appara t pas pour des valeurs plus  lev es.

2.3.2.3 Mod lisation du canal

Pour mod liser le canal χ^2 , on va donc associer   chaque donn e  mise $i = 0$ ou 1 une variable al atoire suivant la loi du χ^2 d’ordre $n = 2\frac{B_o}{B_e}$ avec pour param tre de non centralit  $s^2 = A_i^2$ o  A_i est l’amplitude du champ  mis pour la donn e i . Pour cela, il faut g n rer $n = 2M = 2\frac{B_o}{B_e}$ variables gaussiennes X_k de moyenne commune $m_k^{(i)} = m^{(i)} = \frac{A_i}{\sqrt{2M}}$ et de variance σ^2 pour former la variable $Y = \sum_{k=1}^n X_k^2$.

Pour l’ mission de la donn e ‘0’, on distingue deux cas possibles : le cas id al o  lorsqu’un ‘0’ est  mis, l’amplitude du champ A_0 vaut 0, et le cas r el o  A_0 vaut une

2.3 Modèles de canal

certaine valeur non nulle.

Cas idéal : $A_0 = 0$

Lorsqu'il s'agit d'un '0' émis, pour chaque valeur de k , les composantes du signal sont nulles : $x_k = 0$ et la moyenne de $X_k = x_k + \alpha_k$ est alors nulle ($m^{(0)} = 0$). Dans ce cas, $s_0^2 = nm^{(0)^2} = A_0^2 = 0$ et il suffit de générer $n = 2M$ variables gaussiennes X_k de moyenne nulle ($m^{(0)} = 0$) et de variance σ^2 , afin de calculer $Y = \sum_{k=1}^n X_k^2$.

Cas réel : $A_0 \neq 0$

On suppose ici que l'émission d'un symbole '0' correspond en réalité à un champ d'amplitude A_0 ($A_0 \ll A_1$). Dans ce cas, $x_k = A_0/\sqrt{n}$ et il faut générer $n = 2M$ variables gaussiennes X_k de moyenne $m^{(0)} = x_k = A_0/\sqrt{n}$ et de variance σ^2 . La quantité obtenue en formant $Y = \sum_{k=1}^n X_k^2$ représente la variable χ^2 associée à la donnée émise.

Dans tous les cas, pour l'émission d'un '1', la génération de la variable χ^2 se fait de la même manière que pour un '0' tel que $A_0 \neq 0$, en remplaçant la moyenne des variables gaussiennes par $m^{(1)} = A_1/\sqrt{n}$.

Finalement, une modélisation du canal χ^2 peut se résumer par une fonction f (voir figure 2.7) qui à chaque valeur i valant '0' ou '1' affecte une variable $f(i)$ selon l'expression suivante :

$$f : i \mapsto f(i) = Y = \sum_{k=1}^n X_k^2 \quad (2.36)$$

X_k variable gaussienne($m^{(i)}, \sigma^2$)

2.3.2.4 Paramètres de simulation

Les paramètres nécessaires à la simulation d'une transmission unipolaire sur un canal de type χ^2 se résument finalement à :

- $n = 2B_o/B_e$: ordre du χ^2

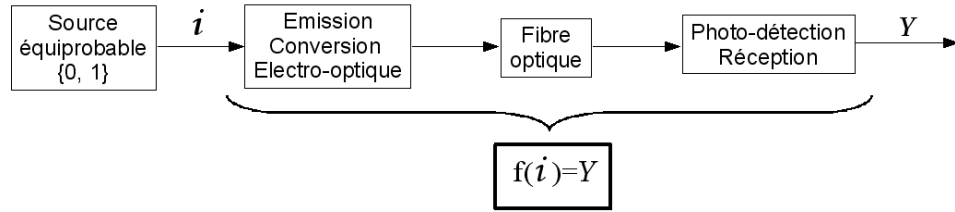


FIG. 2.7 – Modélisation du canal χ^2

- σ^2 : variance des variables gaussiennes
- A_1^2 : puissance d'un '1' émis (ou $m^{(1)} = \frac{A_1}{\sqrt{n}}$)
- A_0^2 : puissance d'un '0' émis (ou $m^{(0)} = \frac{A_0}{\sqrt{n}}$)

On définit généralement les paramètres de non-centralité normalisés δ_1 et δ_0 par :

$$\delta_i = \frac{A_i^2}{\sigma^2}.$$

Dans les simulations réalisées, les valeurs sont normalisées par rapport à la puissance des '1' émis, c'est-à-dire on pose $A_1^2 = 1$. Dans ce cas, les paramètres d'entrée du programme sont :

- n
- δ_1
- δ_0

Les paramètres permettant de générer les variables à statistique χ^2 sont ensuite déterminés par :

- $A_1^2 = 1$ (ou $m^{(1)} = \frac{1}{\sqrt{n}}$)
- $\sigma^2 = \frac{1}{\delta_1}$
- $A_0^2 = \delta_0 \cdot \sigma^2$ (ou $m^{(0)} = \sigma \sqrt{\frac{\delta_0}{n}}$)

Les performances sont ensuite représentées pour une valeur donnée de n en fonction des deux paramètres caractéristiques qui sont :

- le taux d'extinction ER_{in} (par définition donné par (2.1) : $ER_{in} = \frac{A_1^2}{A_0^2}$),
- le facteur de qualité Q , assimilable à un rapport signal à bruit électrique et défini

2.3 Modèles de canal

par :

$$Q = \frac{I_1 - I_0}{\sigma_1 + \sigma_0}. \quad (2.37)$$

Les valeurs I_1 et σ_1^2 (respectivement I_0 et σ_0^2) sont les moyennes $m_Y^{(i)}$ et variances $(\sigma_Y^{(i)})^2$ de la variable reçue ayant une statistique χ^2 lorsqu'il s'agit d'un '1' (respectivement d'un '0') émis, et pouvant s'exprimer avec (2.32) et (2.33).

Ces paramètres sont souvent exprimés en dB en prenant respectivement : $10\log(ER_{in})$ et $20\log(Q)$.

Chacun de ces indicateurs est lié aux paramètres d'entrée du programme avec, pour le taux d'extinction : $ER_{in} = \frac{\delta_1}{\delta_0}$.

Et, pour le paramètre Q :

$$Q = \frac{\delta_1 - \delta_0}{\sqrt{2n} \left(\sqrt{1 + 2\delta_1/n} + \sqrt{1 + 2\delta_0/n} \right)} \quad (2.38)$$

Dans le cas idéal où $s_0^2 = A_0^2 = 0$, le taux d'extinction tend vers l'infini $ER_{in} = \infty$, et Q s'exprime par : $Q = \frac{\delta_1}{\sqrt{2n}(1+\sqrt{1+2\delta_1/n})}$.

On peut remarquer que dans le cas symétrique (canal AWGN), on a $I_1 = A$, $I_0 = 0$ et $\sigma_1 = \sigma_0 = \sigma$ d'où :

$$Q_{AWGN} = 20\log\left(\frac{A}{2\sigma}\right) = 10\log\left(\frac{A^2}{4\sigma^2}\right) = S/B \quad (2.39)$$

Ainsi, dans ce cas, le paramètre Q correspond au S/B défini par (2.8) dans le cas du canal AWGN.

2.4 Performances avec un modèle AWGN

2.4.1 Introduction

Dans cette partie, les performances obtenues sur un canal AWGN comme décrit précédemment sont étudiées. Pour cela, une chaîne de simulation en C a été mise en place afin d'évaluer les TEB. Ces simulations sont effectuées selon la méthode de Monte-Carlo, c'est-à-dire en itérant un très grand nombre de fois la même expérience (ici : émission - réception d'un bit) pour avoir un nombre significatif d'événements recherchés (ici : des erreurs de transmissions).

Dans une première partie, les performances obtenues lors d'une transmission simple sans codage correcteur d'erreurs sont présentées et comparées aux performances théoriques afin de valider la chaîne de simulation. Ensuite, deux cas de transmission sur canal AWGN avec un codage correcteur d'erreurs sont étudiés, l'un utilisant les codes RS, l'autre les codes LDPC.

2.4.2 Performances dans le cas non codé

2.4.2.1 Expression de la probabilité d'erreur

On considère une émission équiprobable de '1' et de '0' ainsi qu'une modulation OOK. La variable reçue $Y = r = e + n$ est la somme du signal émis e et d'une contribution de bruit n , suivant la loi de probabilité décrite par (2.4) et caractérisée par une variance σ^2 . Le signal émis e s'écrit : $e = A \cdot i$ avec $i = 0$ ou 1 la donnée émise et A l'amplitude du signal. La variable décisionnelle Y est directement égale à r .

La densité de probabilité de la variable reçue $Y = r$ prend alors l'allure représentée figure 2.8 (avec Y normalisé par rapport à A , et $\sigma^2 = 0.1$).

2.4 Performances avec un modèle AWGN

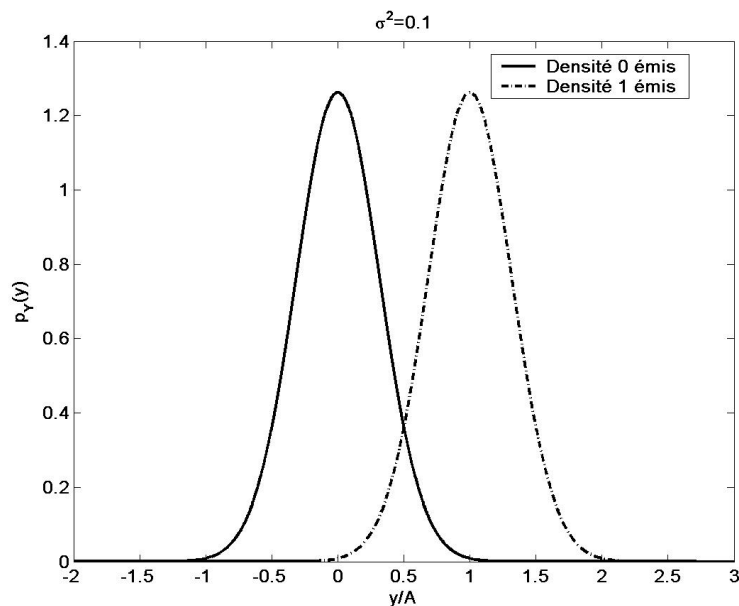


FIG. 2.8 – Exemple de densité de probabilité pour la variable reçue, avec $\sigma^2 = 0.1$ pour un canal AWGN

Soit S le seuil de décision, la probabilité d'erreur s'écrit :

$$P_e = \frac{1}{2}p(Y > S | i = 0) \cdot p(i = 0) + p(Y < S | i = 1) \cdot p(i = 1) \quad (2.40)$$

$$P_e = \frac{1}{2}p(A \cdot i + n > S | i = 0) + \frac{1}{2}p(A \cdot i + n < S | i = 1)$$

Cette relation peut s'exprimer à l'aide de la variable de bruit n dont on connaît la densité de probabilité :

$$P_e = \frac{1}{2}p(n > S) + \frac{1}{2}p(n < S - A) \quad (2.41)$$

Pour une variable aléatoire gaussienne de moyenne nulle (2.4), la probabilité que cette variable dépasse la valeur S est exprimée de manière simple à l'aide de la fonction d'erreur complémentaire :

$$p(n > S) = \frac{1}{2}erfc\left(\frac{S}{\sqrt{2}\sigma}\right) \quad (2.42)$$

En appliquant (2.42) à (2.41), il vient :

$$P_e = \frac{1}{2} \left(\operatorname{erfc} \left(\frac{S}{\sqrt{2}\sigma} \right) + \operatorname{erfc} \left(\frac{A-S}{\sqrt{2}\sigma} \right) \right) \quad (2.43)$$

Dans ce cas, il est possible de démontrer que le seuil optimal de détection qui minimise la probabilité d'erreur est égal à la valeur médiane entre l'amplitude des symboles émis, c'est-à-dire ici $A/2$. Ainsi, la probabilité d'erreur par bit pour le seuil optimal $S = A/2$ est égale à :

$$P_{e(OOK,AWGN)} = \frac{1}{2} \operatorname{erfc} \left(\frac{A}{2\sigma\sqrt{2}} \right) \quad (2.44)$$

Sachant que le rapport signal à bruit S/B est donné par (2.6) avec $E_s = \frac{A^2}{2}$ et que $\sigma^2 = \frac{N_0}{2}$, la probabilité d'erreur (2.44) peut aussi s'écrire en fonction du S/B :

$$P_{e(OOK,AWGN)} = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{S/B}{2}} \right) \quad (2.45)$$

Un tracé de la probabilité d'erreur théorique est représenté sur la figure 2.9 en fonction du S/B . Notons que, par rapport au cas d'une émission bipolaire, les résultats sont identiques, mais décalés de 3dB par rapport au S/B (conversion unipolaire / bipolaire).

2.4.2.2 Résultats de simulations

Le principe de la simulation peut être schématisé comme suit :

- Emission aléatoire et équiprobable d'une donnée binaire ('0' ou '1').
- Ajout d'une variable aléatoire dont la statistique suit la loi normale de moyenne nulle et de variance σ^2 , déterminée à partir du rapport S/B fixé par l'utilisateur.
- Prise de décision par rapport à un seuil (seuil optimal = 0.5).
- Comparaison du bit décidé avec le bit émis pour comptabiliser les erreurs.

2.4 Performances avec un modèle AWGN

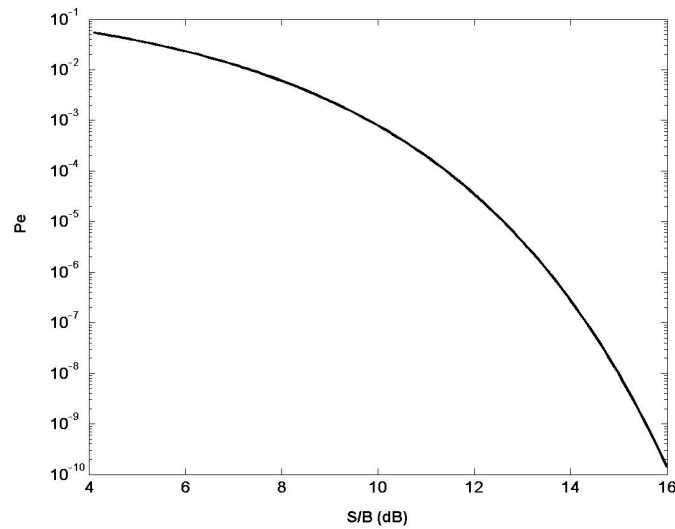


FIG. 2.9 – Probabilité d’erreur théorique sur canal AWGN

Ce principe est itéré un grand nombre de fois de manière à obtenir un nombre d’erreurs suffisamment significatif et à évaluer le Taux d’Erreur Binaire avec :

$$TEB = \frac{\text{Nombre de bits erronés}}{\text{Nombre de bits émis}} \quad (2.46)$$

Les TEB obtenus par simulation avec le seuil optimal sur canal AWGN sont représentés figure 2.10 avec, pour comparaison, la courbe théorique de la probabilité d’erreur.

Les résultats théoriques et simulés sont bien concordants, ce qui permet de valider la chaîne de transmission de base utilisée par la suite pour évaluer les performances des différents codes correcteurs d’erreurs sur ce type de canal. En effet, comme nous le verrons dans les paragraphes suivants, il n’est pas toujours évident de déterminer la probabilité d’erreur théorique lorsqu’on utilise des codes correcteurs et les TEB seront alors évalués uniquement par simulation.

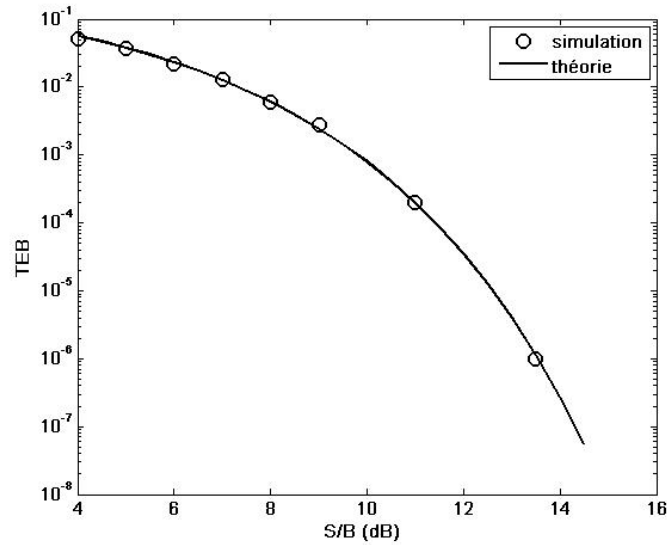


FIG. 2.10 – TEB obtenus par simulation sur canal AWGN

2.4.3 Performances avec les codes RS

2.4.3.1 Expression de la probabilité d'erreur

Le taux d'erreur binaire (TEB) d'une transmission utilisant les codes RS peut être obtenu de manière approchée comme décrit dans les normes ITU-T [23, 24]. Le raisonnement est décrit ci-après. Soit P_{UE} la probabilité d'erreur non corrigible, le TEB après codage, ou TEB_{OUT} peut s'exprimer avec :

$$TEB_{OUT} = 1 - (1 - P_{UE})^{\frac{1}{m}} \quad (2.47)$$

où on rappelle que m correspond au nombre de bits par symboles définissant la classe du code RS ($N = 2^m - 1$).

Les erreurs deviennent non corrigibles lorsqu'elles sont trop nombreuses par rapport au pouvoir de correction du code. Ainsi, la probabilité d'erreur non corrigible P_{UE} peut s'exprimer à partir de la probabilité d'erreur dans un mot P_{SE} et de t , nombre maximum

d'erreurs corrigibles :

$$P_{UE} = \sum_{i=t+1}^N \frac{i}{N} \binom{N}{i} (P_{SE})^i (1 - P_{SE})^{N-i} \quad (2.48)$$

La probabilité d'erreur dans un mot P_{SE} peut elle aussi être mise en relation avec le TEB avant codage TEB_{IN} de la manière suivante :

$$TEB_{IN} = 1 - (1 - P_{SE})^{\frac{1}{m}} \quad (2.49)$$

A partir de l'expression (2.49), on peut extraire P_{SE} :

$$P_{SE} = 1 - (1 - TEB_{IN})^m \quad (2.50)$$

Avec ces expressions, le TEB avec codage (TEB_{OUT}) est obtenu à partir du TEB avant codage (TEB_{IN}) en remplaçant (2.50) dans (2.48) puis dans (2.47).

On peut aussi exprimer le TEB codé en fonction du S/B sachant que sur canal AWGN, le TEB non codé est lié au S/B non codé noté ici S/B_u par (2.45). De plus, comme nous l'avons vu , le S/B dans le cas codé intègre un facteur N/K par rapport au S/B non codé, c'est-à-dire :

$$S/B = S/B_u \cdot \frac{N}{K} \quad (2.51)$$

En remplaçant l'expression de TEB_{IN} (2.45) dans (2.50), il vient :

$$P_{SE} = 1 - \left(1 - \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{S/B \cdot K}{2 \cdot N}} \right) \right)^m \quad (2.52)$$

Ainsi, le TEB codé est déterminé en calculant P_{SE} à l'aide de (2.52), puis P_{UE} avec (2.48)

et TEB_{OUT} avec (2.47) comme pr cedemment.

Des approximations peuvent  tre faites sous certaines conditions :

- Si $P_{UE} \ll 1$, (2.47) devient :

$$TEB_{OUT} = \frac{1}{m} P_{UE} \quad (2.53)$$

Cette simplification permet d' viter d'obtenir des valeurs erron es nulles pour TEB_{OUT} lorsque P_{UE} est trop faible. En effet, dans ce cas, $(1 - P_{UE})^{\frac{1}{m}}$ se trouve arrondi   1 et l'expression (2.47) donne un r sultat nul.

- Si $P_{SE} \ll 1$, les termes $(P_{SE})^i$ dans (2.48) deviennent n gligeables pour $i > t + 1$ donc seul un terme reste et (2.48) peut se simplifier en :

$$P_{UE} = \frac{t+1}{N} \cdot \binom{N}{t+1} (P_{SE})^{t+1} (1 - P_{SE})^{N-(t+1)} \quad (2.54)$$

- Si $TEB_{IN} \ll 1$, alors (2.50) devient :

$$P_{SE} = m \cdot TEB_{IN} \quad (2.55)$$

Suivant la valeur du S/B et les param tres du code RS, chacune de ces conditions peut  tre valable ou non, et l'obtention du TEB cod  peut  tre obtenu   l'aide des approximations.

Les r sultats th oriques obtenus avec $m = 8$ pour les codes $RS(255, K)$, pour diff rentes valeurs de K , sont repr sent s figure 2.11 en fonction du rapport signal   bruit cod  S/B. Dans chacun des cas, le pouvoir de correction t , donn  par $2t = N - K = 255 - K$ varie aussi.

On peut remarquer de mani re pr visible que plus K diminue, plus les performances

2.4 Performances avec un modèle AWGN

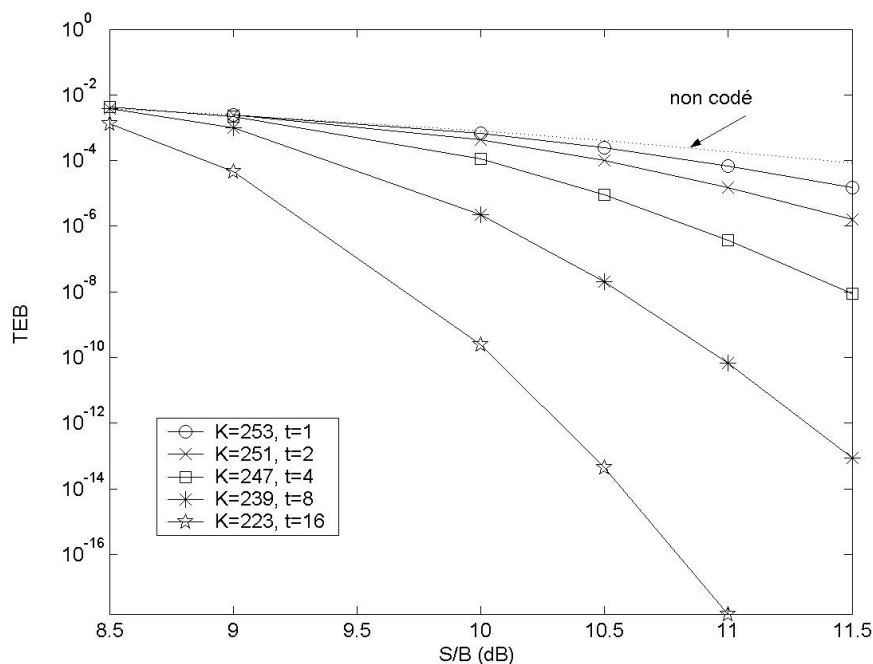


FIG. 2.11 – Performances théoriques sur canal AWGN des codes RS(255,K)

du code correcteur sont bonnes. En effet, dans ce cas, le pouvoir de correction t augmente, et le code est capable de corriger plus d'erreurs dans le mot reçu. On note ainsi un décalage important en terme de S/B requis pour atteindre un TEB visé. En effet, pour atteindre 10^{-8} par exemple, le S/B nécessaire avec un code RS(255,223) est de 9,7 dB tandis qu'il n'est atteint avec le RS(255,247) que pour un SNR de 11,5 dB. Cette différence est d'autant plus importante que le TEB est faible.

Cependant, même si un code plus redondant conduit à de meilleures performances, il ne faut pas oublier que c'est au détriment du débit utile qui se trouve alors diminué. Ce compromis sera plus particulièrement étudié dans la dernière partie de ce chapitre illustrant le dimensionnement des codes correcteurs pour le réseau optique ECOFRAME.

2.4.3.2 Résultats de simulations

Les simulations sont réalisées à partir de la chaîne précédemment validée sur canal AWGN dans le cas sans codage. Le principe est le même à part qu'il ne s'agit plus d'un bit émis puis reçu, mais d'un bloc de bits. Il peut donc se résumer par :

- Emission aléatoire et équiprobable de K données binaires constituant un bloc de taille K .
- Multiplication du bloc par le polynôme générateur du code pour former un bloc de N bits composé des K bits d'information et de $N - K$ bits de redondance.
- Ajout sur chaque bit d'une variable aléatoire dont la statistique suit la loi normale de moyenne nulle et de variance σ^2 , déterminée à partir du rapport S/B (codé) fixé par l'utilisateur.
- Prise de décision par rapport à un seuil (seuil optimal = 0.5) pour obtenir les valeurs "hard" à l'entrée du décodeur.
- Décodage utilisant l'algorithme de Berlekamp-Massey (voir Annexe B) [12, 13], à l'issue duquel les K bits d'information sont extraits et corrigés, en fonction du pouvoir de correction du code et du nombre d'erreurs dans le bloc de bits reçu.
- Comparaison de chaque bit décidé avec le bit émis correspondant et comptabilisation des erreurs.

La figure 2.12 représente les résultats obtenus par simulation, comparés à ceux obtenus à l'aide de la formule théorique (2.47) pour les codes $RS(255, K)$ avec $K = 247$ et $K = 239$. On peut remarquer que les résultats sont concordants, mais que ceux obtenus par simulation sont légèrement supérieurs à ceux obtenus par la formule théorique. La formule théorique, basée sur des approximations permet en effet de donner une borne inférieure puisque certains termes sont, comme nous l'avons vu, négligés.

2.4 Performances avec un modèle AWGN

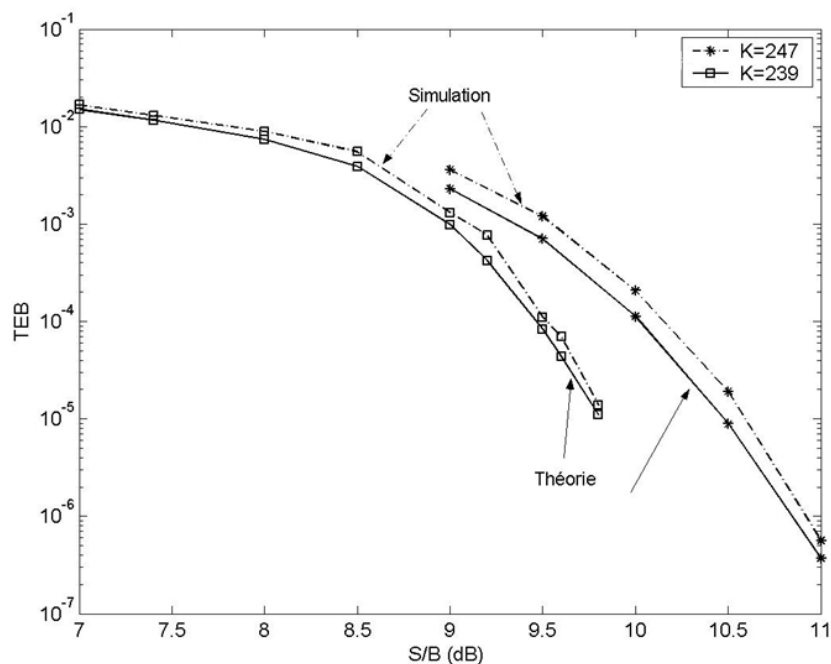


FIG. 2.12 – Comparaison des TEB simulés avec la probabilité d'erreur théorique pour des codes RS (255, K)

2.4.4 Performances avec les codes LDPC

2.4.4.1 Contexte

Pour les codes LDPC, il n'est pas simple de formuler le TEB de manière théorique en fonction des paramètres du code. En effet, les performances dépendent non seulement des dimensions du code, du nombre d'itérations, mais aussi de la structure de la matrice de parité. Ainsi, les performances présentées par la suite sont obtenues uniquement par simulation. Le principe général de la chaîne de simulation en C est identique à celui décrit pour les codes RS à la différence que le bloc de K bits est multiplié par la matrice d'encodage et non par le polynôme pour former le bloc codé de N bits.

Au niveau de la réception, l'architecture du décodeur (variables d'entrée, sortie) est différente de celle des codes RS et basée sur l'algorithme de décodage BP pour les

codes LDPC, pr sent  dans le chapitre 1. De plus, comme nous l'avons vu au cours du chapitre 1, l' tape d'initialisation de l'algorithme de d codage constitue une  tape importante ayant un impact sur l'efficacit  du d codage des codes LDPC. En simulation, on utilise le d codage souple   propagation de croyance, dont la phase d'initialisation consiste   calculer les rapports logarithmiques de vraisemblance, ou LLR : Logarithmic Likelihood Ratio. Apr s un paragraphe dans lequel l'expression des LLR pour un canal optique AWGN est pr sent e, les performances des codes LDPC sur ce type de canal sont ensuite analys es.

2.4.4.2 Expressions des rapports logarithmiques de vraisemblance (LLR)

En notant e la donn e  mise et Y la variable d cisionnelle, rappelons que la d finition g n rale du LLR est :

$$LLR = \ln \left(\frac{p(e = 0 | Y)}{p(e = 1 | Y)} \right) \quad (2.56)$$

Sachant que $Y = e + n$, avec n la variable de bruit gaussienne dont on conna t la densit  de probabilit  (2.4), les probabilit  recherch es dans (2.56) s'expriment de la mani re suivante :

$$p(e = 0 | Y) = p(n = Y | Y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y)^2}{2\sigma^2}} \quad (2.57)$$

$$p(e = 1 | Y) = p(n = Y - 1 | Y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y-1)^2}{2\sigma^2}} \quad (2.58)$$

Le rapport logarithmique des deux expressions conduit   :

$$LLR = \ln \left(\frac{\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y-1)^2}{2\sigma^2}}}{\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y)^2}{2\sigma^2}}} \right) \quad (2.59)$$

Ce qui se simplifie en :

$$\begin{aligned} LLR &= \ln \left(e^{-\frac{(Y-1)^2}{2\sigma^2}} e^{+\frac{(Y)^2}{2\sigma^2}} \right) \\ LLR &= \ln \left(e^{\frac{Y^2 - (Y-1)^2}{2\sigma^2}} \right) \end{aligned} \quad (2.60)$$

Finalement, les LLR prennent une expression simple en fonction de la variable reçue Y et de la variance du bruit gaussien σ^2 :

$$LLR = \frac{2Y - 1}{2\sigma^2} \quad (2.61)$$

Cette expression est identique à celle classiquement formulée pour les canaux bipolaires avec $Y_b = 2Y - 1$ et $\sigma_b^2 = (2\sigma)^2$, en notant Y_b et σ_b^2 les variables et variances associées dans le cas bipolaire. Une fois les LLR initialisés, le décodage est réalisé selon le principe énoncé dans le chapitre 1 avec un nombre d'itérations maximal fixé à 100 dans un premier temps. En effet, on est certain dans ce cas que les performances ne dépendent plus du nombre d'itérations.

2.4.4.3 Analyse des performances

Résultats de simulations

Différentes simulations ont été réalisées sur la chaîne de transmission avec un modèle de canal AWGN pour des codes LDPC construits selon la méthode BIBD présentée chapitre 1 [50]. Dans un premier temps, pour valider la chaîne de transmission avec codage / décodage LDPC, les TEB obtenus sont comparés à ceux précédemment publiés [50], obtenus sur un canal AWGN bipolaire. La figure 2.13 représente les performances obtenues à l'aide des codes LDPC BIBD (2715,2535) et (1448,1268) de redondances respectives : 7,1% et 14%.

Par comparaison aux résultats obtenus avec les mêmes codes sur un canal bipolaire, présentés dans [50] (Figure 1), les résultats sont identiques, mais décalés de 3 dB par

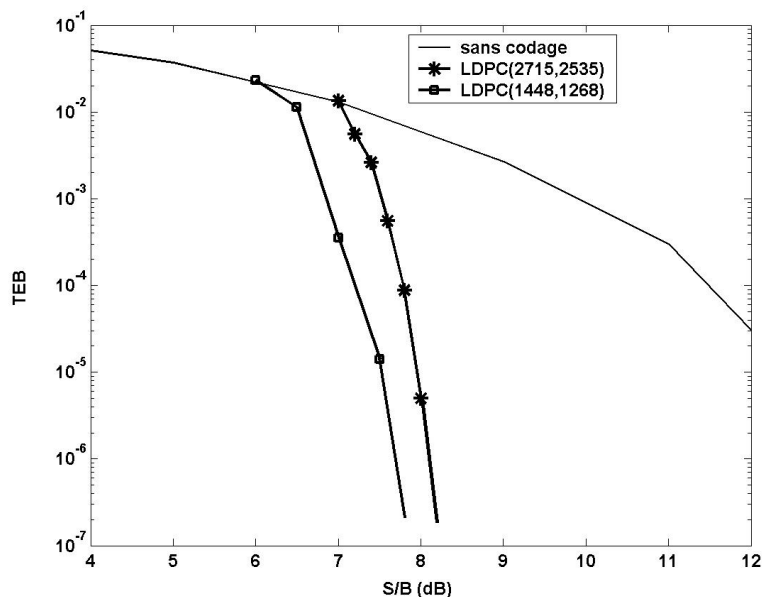


FIG. 2.13 – TEB obtenus par simulation avec les codes LDPC construits selon la méthode BIBD

rapport au S/B suite à la conversion unipolaire / bipolaire, comme dans le cas non codé. Ceci permet de valider l'encodage réalisé avec des codes LDPC-BIBD ainsi que l'algorithme de décodage utilisé pour les codes LDPC.

De plus, on peut remarquer à partir de la figure 2.13 que le code LDPC(1448,1268) de redondance deux fois plus élevée que le code LDPC(2715,2535) conduit à de meilleures performances. En effet, il a un pouvoir de correction plus élevé. Cependant, la pente des deux courbes de performance n'est pas la même : celle du LDPC(2715,2535) est plus importante. Ceci est dû au fait que la longueur des deux codes est différente, et le code le plus long possède une pente plus forte. Une étude paramétrique des codes LDPC-BIBD est plus particulièrement détaillée dans la dernière partie de ce chapitre, illustrant le dimensionnement des codes correcteurs au réseau ECOFRAME.

Les paragraphes suivants analysent de manière générale les résultats obtenus à l'issue de différentes simulations.

Influence du nombre d'itérations maximal

Afin d'illustrer et d'évaluer la convergence de l'algorithme de décodage, les performances obtenues pour différents nombres maximaux d'itérations sont reportées figure 2.14 pour le code LDPC(6176,5983) construit à partir de la méthode BIBD. Notons alors que le nombre d'itérations réalisées dans chacun des cas n'est pas nécessairement égal au nombre maximal d'itérations fixé, dans la mesure où il existe deux conditions d'arrêt : le mot décodé est un mot de code ou bien le nombre maximal d'itérations est atteint.

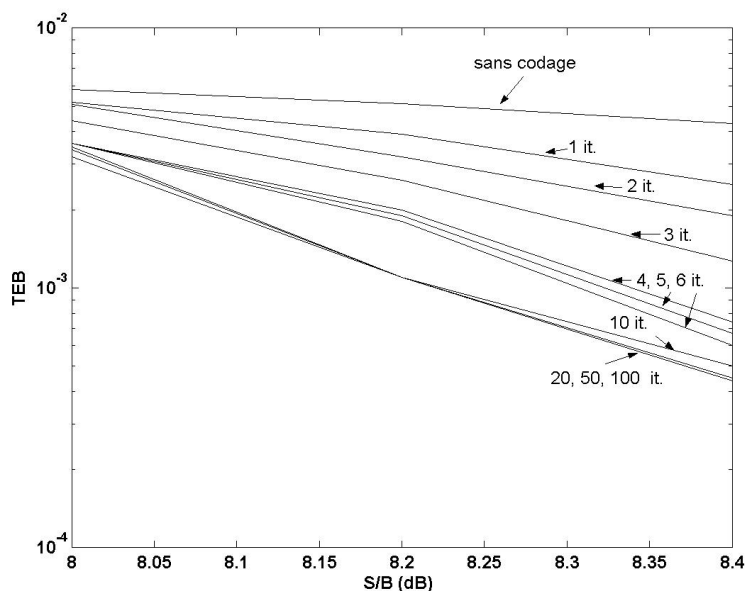


FIG. 2.14 – TEB pour différents nombres maximaux d'itérations avec le LDPC(6176,5983)

A partir de l'exemple présenté, on peut conclure que l'algorithme de décodage converge relativement rapidement avec ce type de codes LDPC, puisque dans cet exemple, 10 à 20 itérations suffisent pour obtenir les TEB finaux. Ce paramètre est important puisque le temps pris pour réaliser le décodage est proportionnel au nombre d'itérations réalisées.

*Comparaison des performances pour des codes obtenus par les méthodes
BIBD ou aléatoire*

Les performances obtenues à partir du code LDPC BIBD (2715,2535) ont été comparées à celles obtenues avec un code LDPC de mêmes dimensions, mais construit à partir de la méthode aléatoire incluant la suppression des cycles d'ordre 4. La figure 2.15 présente les TEB obtenus par simulation avec ces deux codes.

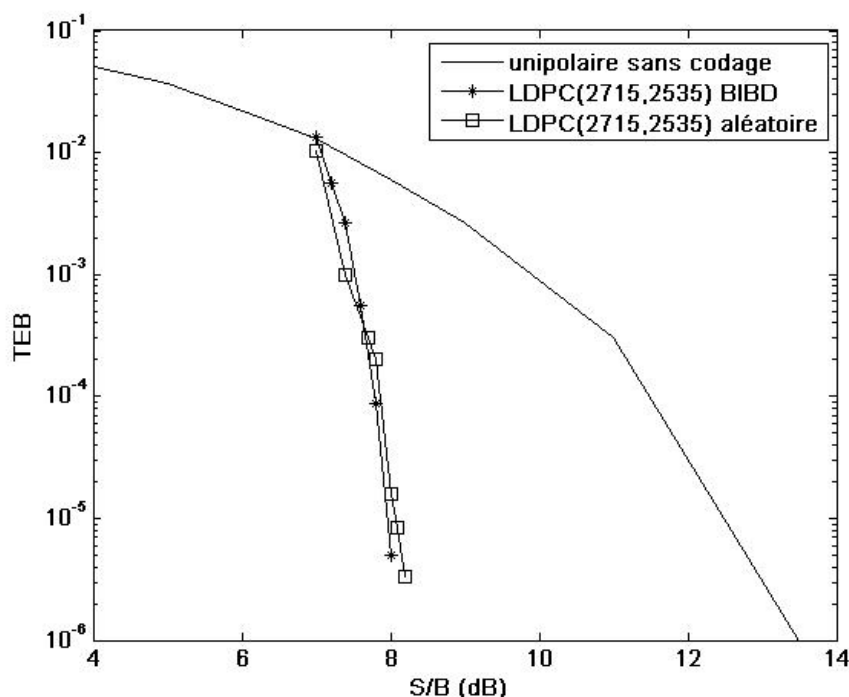


FIG. 2.15 – Comparaison des performances avec les codes LDPC(2715,2535) obtenus avec la méthode BIBD et la méthode aléatoire

On se rend compte que les TEB obtenus avec les deux codes correcteurs sont quasiment identiques. Ainsi, l'intérêt d'utiliser la construction aléatoire n'est pas vraiment significatif. En effet, pour le cas de codes très peu redondants (ici 7% de redondance), la matrice de parité possède peu de lignes, et le choix de la position des '1' dans la matrice est

restreint. Ainsi, les structures aléatoires sont peu différentes de celles issues des méthodes déterministes, et les performances ne sont pas nécessairement meilleures.

Ainsi, par la suite, comme nous ne considérerons que l'utilisation de codes peu redondants dans le cadre des transmissions optiques haut débit, les codes LDPC envisagés seront tous issus de la méthode BIBD.

Comparaison codes RS

Un dernier critère d'évaluation des performances des codes LDPC-BIBD est la comparaison avec les codes RS, de redondance et de taille (en nombre de bits) équivalentes. Par exemple, le code LDPC(6176,5983) de redondance égale à 3,2% est comparé au code RS(511,495) qui possède en fait $511 \times 9 = 4599$ bits par bloc dont 4401 bits utiles. La figure 2.16 représente les TEB obtenus pour ces deux codes en fonction du rapport signal à bruit sur canal AWGN. Un autre exemple avec les codes LDPC(2034,1921) et RS(255,241) est représenté 2.17. Dans ce cas, la redondance des deux codes est égale à 5,8 %. Le code RS possède une taille de $255 \times 8 = 2040$ bits et est donc tout à fait comparable au code LDPC envisagé.

Sur chacun des deux exemples, on peut remarquer que les performances sont meilleures avec les codes LDPC. En terme de gain de codage, on note une différence de 2 dB environ pour un TEB de 10^{-9} . Etant donné que cette différence augmente lorsque le TEB diminue, on a une différence en gain de codage à 10^{-12} légèrement supérieure à 2 dB. Une étude plus exhaustive est présentée dans la dernière partie du chapitre, et révèle que cette différence de gain de codage à 10^{-9} , pour des codes LDPC et RS de taille et redondance équivalentes est toujours proche de 2dB.

L'amélioration du gain de codage apportée par les codes LDPC est conforme aux résultats précédemment publiés [29] et reportés figure 1.5. En effet, à partir de cette figure, on relève un gain de codage à 10^{-12} de 3 à 4 dB supérieur avec les codes LDPC par

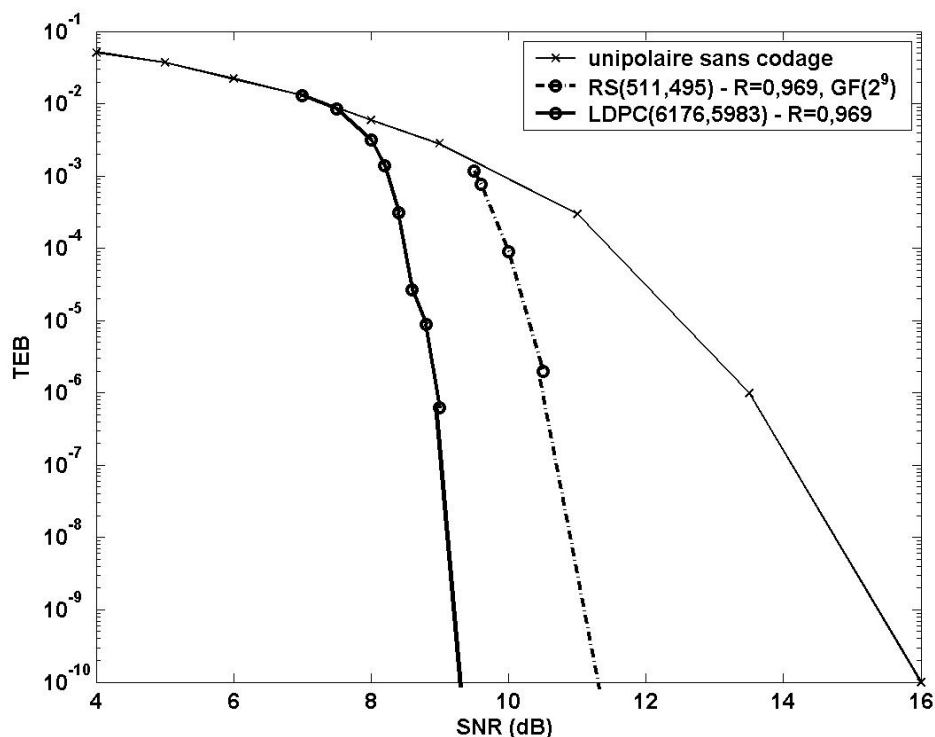


FIG. 2.16 – Comparaison des performances des codes LDPC(6176,5983) et RS(511,489) rapport au code RS(255,239) pr sent . Or, les gains de codage repr sent s figure 1.5 sont obtenus pour des codes LDPC plus redondants ($r = 23\%$) donc plus efficaces que le code de r f rence RS(255,239) pour lequel $r = 6,7\%$. Ceci explique le fait que la diff rence de gain de codage (3-4 dB) est plus importante qu'entre des codes RS ou LDPC de m me rendement comme nous l'avons  tabli dans ce paragraphe (2 dB).

2.4.4.4 Conclusions

Les codes LDPC ont une bonne capacit  de correction sur canal unipolaire AWGN. Plus particuli rement, les codes LDPC-BIBD constituent un bon compromis parmi les codes LDPC, ils poss dent une architecture simple tout en ayant un fort pouvoir de correction et une convergence rapide. De plus, ils se r v lent  tre plus int ressants que

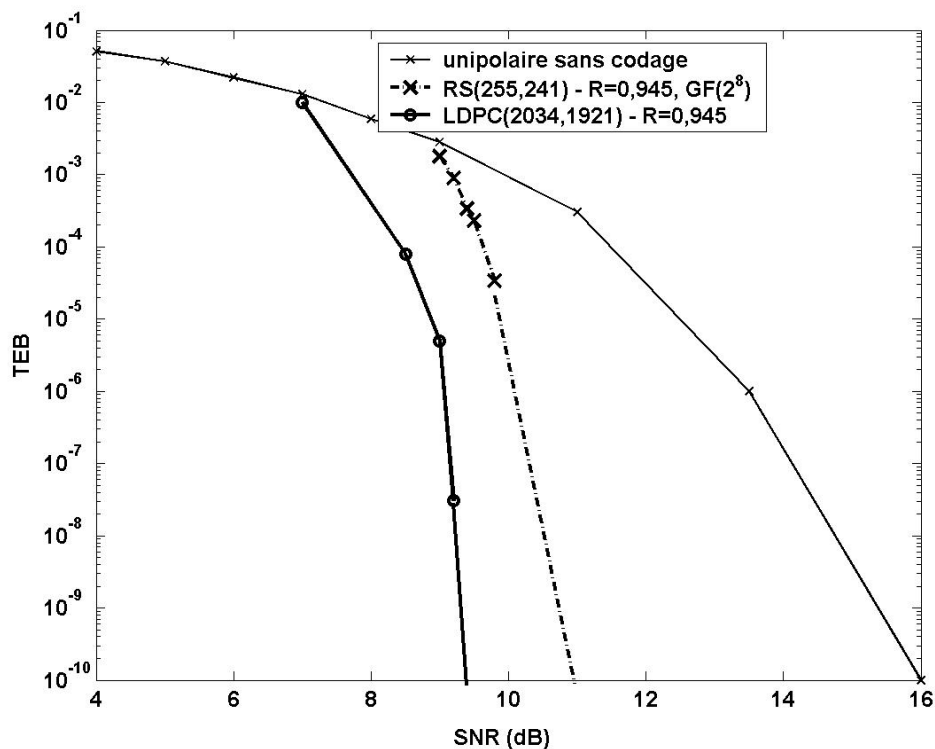


FIG. 2.17 – Comparaison des performances des codes LDPC(2034,1921) et RS(255,241)

des codes RS de caractéristiques similaires. L'étude du compromis entre efficacité de correction, perte en débit et rapidité de décodage est illustrée dans la dernière partie de ce chapitre constituant une application du dimensionnement au réseau ECOFRAME.

2.4.5 Conclusions

Les performances des codes RS et LDPC ont été étudiées sur le type de canal le plus simple à envisager : le canal AWGN. Les résultats obtenus sont en accord avec les résultats établis par les formules approchées données par l'ITU [23, 24] en ce qui concerne les codes RS. Pour les codes LDPC, même si aucune formule ne permet de traduire l'amélioration de performance qu'ils apportent, les résultats ont pu être comparés à ceux présentés sur

canal AWGN en transmission bipolaire. Ainsi, les performances obtenues avec les codes LDPC-BIBD sont conformes à celles précédemment obtenues dans le contexte bipolaire.

Parmi les autres types de codes LDPC, les codes LDPC-BIBD apparaissent finalement comme un bon compromis et seront donc les seuls envisagés par la suite. De plus, par rapport aux codes RS, on remarque qu'à redondance et taille équivalentes, ils sont plus efficaces de 2 dB environ pour un TEB de 10^{-9} .

La partie suivante est consacrée à l'étude des performances de ces codes sur un canal χ^2 , plus représentatif du canal optique comme nous l'avons précédemment justifié. L'influence des caractéristiques du canal sur les paramètres du décodage et sur les performances est particulièrement analysée.

2.5 Performances avec le modèle χ^2

2.5.1 Cas non codé

2.5.1.1 Expression de la probabilité d'erreur

Comme dans le cas AWGN, il est possible de déterminer, à partir de la densité de probabilité de la variable décisionnelle Y , la probabilité d'erreur d'une transmission sur un canal de type χ^2 . Pour chaque symbole émis ('0' ou '1'), l'amplitude A_i diffère et la densité de probabilité de la variable reçue prend alors une forme dissymétrique bien particulière. Un exemple est représenté figure 2.18 pour $ER_{in} = \frac{\delta_1}{\delta_0} = 7dB$, $\sigma^2 = 0.03$ et $n = 4$. On remarque sur cet exemple que les densités de probabilité pour un '1' ou pour un '0' émis ne se coupent pas en 0.5 comme dans le cas AWGN et que les deux courbes ne sont pas identiques (canal non symétrique). Ainsi, le seuil optimal de décision pour lequel la probabilité d'erreur est minimale, n'est pas égal à 0.5.

2.5 Performances avec le modèle χ^2

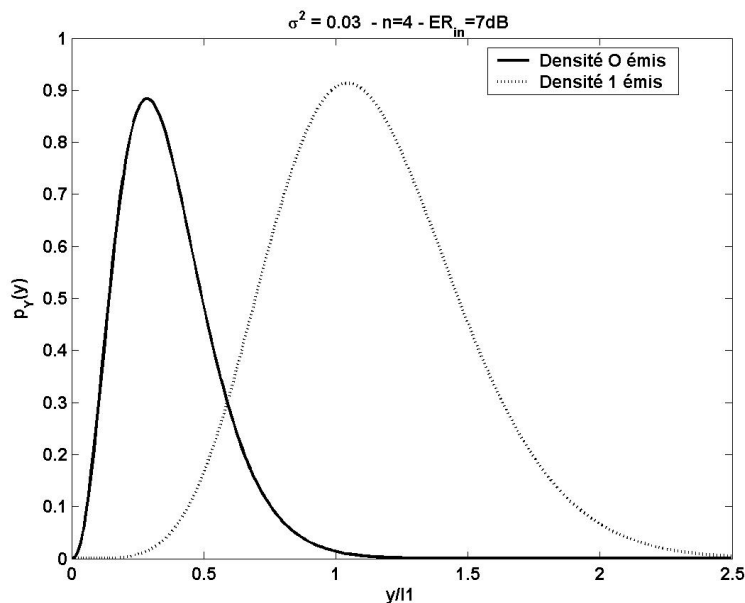


FIG. 2.18 – Exemple de densité de probabilité pour la variable reçue, avec $\sigma^2 = 0.03$ $ER_{in} = 7dB$ dans le cas d'un canal χ^2 avec $n = 4$

Pour déterminer la probabilité d'erreur P_e , il faut déterminer la probabilité d'erreur sur les '1' P_{e1} et celle sur les '0' P_{e0} . En supposant l'émission équiprobable, on a :

$$P_e = \frac{1}{2}P_{e1} + \frac{1}{2}P_{e0} \quad (2.62)$$

P_{e1} est obtenue comme la probabilité que la variable décisionnelle Y soit plus petite que le seuil tandis qu'un '1' a été émis et inversement pour P_{e0} c'est-à-dire :

$$\begin{aligned} P_{e1} &= p(Y < S | i = 1) = \int_{-\infty}^S p_1(y) dy \\ P_{e0} &= p(Y > S | i = 0) = \int_S^{+\infty} p_0(y) dy \end{aligned} \quad (2.63)$$

Rappelons que la densité de probabilité de la variable reçue Y s'écrit de la manière

suivante, pour y positif, en notant A_i l'amplitude correspondant   la donn e i :

$$p_Y^{(i)}(y) = \frac{1}{2\sigma^2} \left(\frac{y}{A_i^2} \right)^{\left(\frac{n-2}{4}\right)} e^{-\frac{1}{2\sigma^2}(y+A_i^2)} I_{n/2-1} \left(\sqrt{y} \frac{A_i}{\sigma} \right) \quad (2.64)$$

L'int grale de cette fonction s' crit gr ce   la fonction de Marcum g n ralis e, d finie pour $0 < a < b$ par [63] :

$$Q_m(a, b) = \int_b^{+\infty} x \left(\frac{x}{a} \right)^{m-1} e^{-\frac{x^2+a^2}{2}} I_{m-1}(ax) dx \quad (2.65)$$

Ainsi, en effectuant un changement de variables, et sachant que $p_Y^{(i)}(y)$ n'est d finie que pour des valeurs positives de y , on obtient pour Pe_1 :

$$Pe_1 = 1 - Q_{n/2} \left(\frac{A_1}{\sigma}, \frac{\sqrt{S}}{\sigma} \right) \quad (2.66)$$

Et pour Pe_0 :

$$Pe_0 = Q_{n/2} \left(\frac{A_0}{\sigma}, \frac{\sqrt{S}}{\sigma} \right) \quad (2.67)$$

Ainsi, la probabilit  d'erreur s' crit de la mani re suivante :

$$Pe = \frac{1}{2} \left(1 + Q_{n/2} \left(\frac{A_0}{\sigma}, \frac{\sqrt{S}}{\sigma} \right) - Q_{n/2} \left(\frac{A_1}{\sigma}, \frac{\sqrt{S}}{\sigma} \right) \right) \quad (2.68)$$

La relation (2.68) peut aussi  tre exprim e en fonction des param tres de non-centralit  normalis s $\delta_i = \frac{A_i^2}{\sigma^2}$ et avec A_1 normalis    1, (2.68) devient :

$$Pe = \frac{1}{2} \left(1 + Q_{n/2} \left(\sqrt{\delta_0}, \sqrt{S\delta_1} \right) - Q_{n/2} \left(\sqrt{\delta_1}, \sqrt{S\delta_1} \right) \right) \quad (2.69)$$

A partir de (2.69), on d termine pour chaque configuration le seuil optimal S

2.5 Performances avec le modèle χ^2

permettant de minimiser P_e .

Pour représenter les performances, rappelons que les paramètres caractérisant le canal χ^2 sont le rapport d'extinction ER_{in} , le facteur de qualité Q ainsi que l'ordre de la distribution χ^2 : $n = 2B_o/B_e$. Les figures 2.19 et 2.20 représentent les performances théoriques issues de (2.69) sur un canal χ^2 en fonction de Q . La figure 2.19 représente les TEB obtenus pour différentes valeurs de ER_{in} lorsque n est fixé ($n = 4$) tandis qu'ils sont représentés figure 2.20 pour différentes valeurs de n lorsque ER_{in} est fixé (cas limite ER_{in} infini). Dans chaque cas, les TEB tracés sont ceux obtenus pour le seuil optimal.

A titre de comparaison, les performances obtenues sur canal AWGN (avec $Q = S/B$ dans ce cas (2.39)) sont aussi représentées.

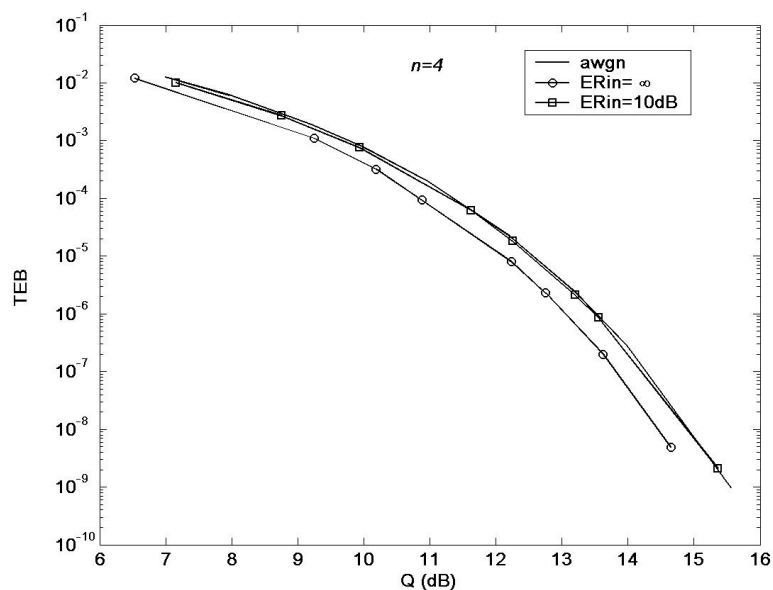


FIG. 2.19 – TEB en fonction du facteur de qualité Q sur un canal χ^2 avec $n = 4$ pour différentes valeurs de ER_{in}

A partir de la figure 2.19, on remarque que pour une valeur de n fixée, le cas ER_{in} infini correspond à un cas limite, pour lequel le TEB est plus faible que pour les autres

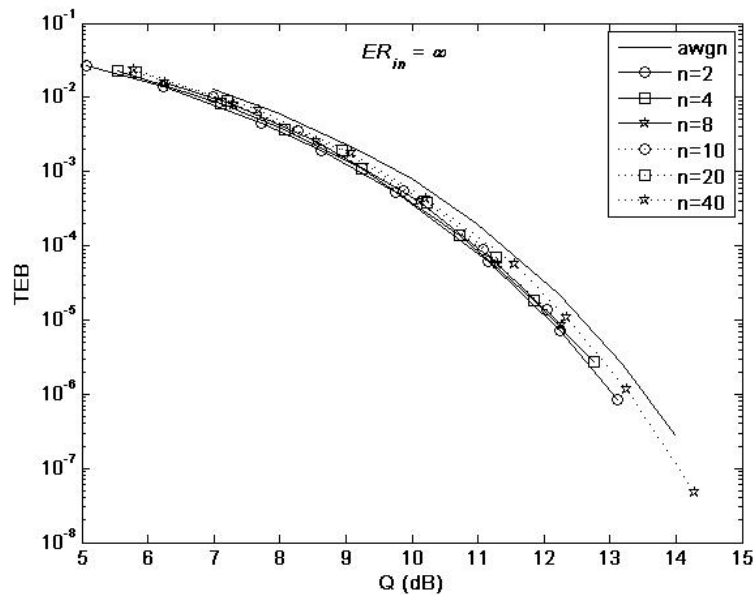


FIG. 2.20 – TEB en fonction du facteur de qualité Q sur un canal χ^2 pour différentes valeurs de n , avec ER_{in} infini

valeurs de ER_{in} . Lorsque ER_{in} diminue, les performances tracées en fonction du facteur Q tendent vers celles obtenues sur canal AWGN. En effet, dans ce cas, les variances des distributions χ^2 résultant des '1' et des '0' tendent à être égales, donc le canal χ^2 se symétrise. Ainsi, les courbes obtenues pour ER_{in} infini et sur canal AWGN forment les bornes inférieure et supérieure des performances obtenues sur un canal χ^2 pour n'importe quelle valeur de ER_{in} .

Le modèle simple et classique AWGN est alors suffisant lorsqu'on souhaite caractériser les performances du système optique. Cependant, comme nous le verrons par la suite, en vue de l'utilisation de codage correcteur d'erreurs, il est important de connaître précisément le comportement du signal sur le canal afin d'obtenir les performances idéales.

A partir de la figure (2.20), on note que pour ER_{in} fixé (pris infini dans cet exemple), plus n augmente, plus les performances tendent vers celles du cas AWGN. En effet, comme

on a déjà pu l'observer sur les courbes (2.4) et (2.5), lorsque n augmente, les densités de probabilité se symétrisent et tendent vers le cas symétrique AWGN.

2.5.1.2 Résultats de simulations

Des simulations ont été effectuées dans les deux cas présentés c'est-à-dire lorsque le rapport d'extinction est infini ($A_0 = 0$) puis dans le cas réel où $A_0 \neq 0$ afin de valider la formule précédemment présentée. Dans tous les cas, rappelons que l'amplitude correspondant à un '1' émis est normalisée : $A_1 = 1$.

La différence par rapport au cas AWGN réside dans le choix du seuil optimal (toujours égal à 0,5 dans le cas AWGN). Il est obtenu numériquement à partir de l'expression de la densité de probabilité de la variable décisionnelle (2.64) en déterminant la valeur de y telle que $p_Y^{(0)}(y) = p_Y^{(1)}(y)$. Il peut également être déterminé en recherchant la valeur du seuil minimisant la probabilité d'erreur théorique donnée par (2.69).

Afin de comparer les performances, les TEB obtenus pour $n = 2$ sont reportés figure 2.21 en fonction du paramètre Q pour ER_{in} infini et ER_{in} égal à 10 dB. Les performances sont comparées à celles issues de la formule théorique P_e (2.69). Dans chaque cas, on a considéré le seuil optimal.

Notons que les résultats sont concordants, ce qui valide la chaîne de simulation pour le canal χ^2 . Par la suite, ce modèle de simulation sera utilisé pour la transmission des données codées, avec les codes RS et les codes LDPC.

Tout comme dans le cas du canal AWGN, où les valeurs de S/B codé ou non codé sont différentes pour des paramètres de simulation fixés, la valeur de Q pour un canal χ^2 diffère selon les cas codé et non codé. En effet, tandis que pour le cas non codé, on considère que $A_1^2 = 1$, dans le cas codé $A_1^2 = N/K$. De même, la valeur de A_0^2 est multipliée par un facteur N/K par rapport au cas non codé. La valeur du facteur de qualité dans le cas

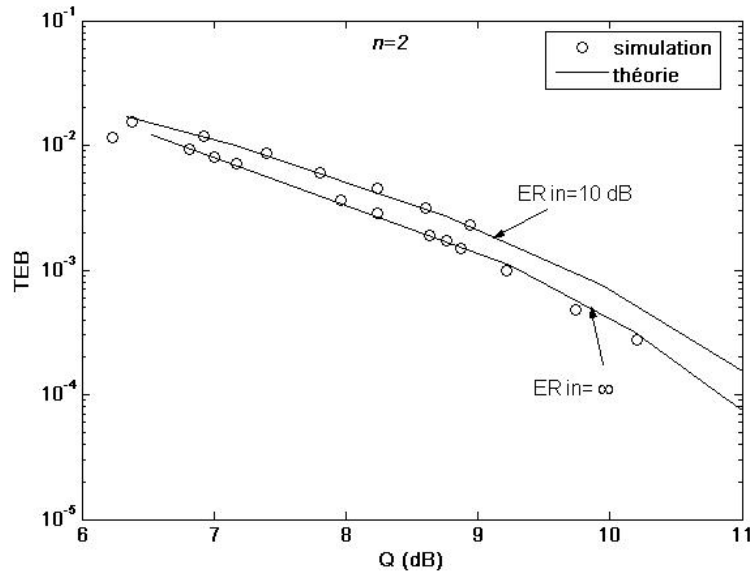


FIG. 2.21 – TEB obtenus par simulation sur canal χ^2 , $n = 2$

cod  Q_c s'exprime par :

$$Q_c = \frac{N}{K} \frac{\delta_1 - \delta_0}{\sqrt{2n} \left(\sqrt{1 + 2\delta_1 \frac{N}{K \cdot n}} + \sqrt{1 + 2\delta_0 \frac{N}{K \cdot n}} \right)} \quad (2.70)$$

Cette expression peut se simplifier si on consid re le cas o  K/N est proche de 1 c'est- -dire pour des codes peu redondants :

$$Q_c \approx \sqrt{\frac{N}{K}} \frac{\delta_1 - \delta_0}{\sqrt{2n} \left(\sqrt{1 + 2\delta_1/n} + \sqrt{1 + 2\delta_0/n} \right)} \quad (2.71)$$

Ce qui correspond   :

$$Q_c \approx \sqrt{\frac{N}{K}} \cdot Q \quad (2.72)$$

Soit en dB :

$$Q_c \approx Q + 10 \cdot \log \left(\frac{N}{K} \right) \quad (2.73)$$

2.5.2 Performances avec les codes RS

2.5.2.1 TEB sur canal χ^2

La probabilité d'erreur présentée pour le canal AWGN (2.47) ne tenant pas compte du type de canal, les performances théoriques codées sont donc obtenues de la même manière que sur canal AWGN. Pour des paramètres systèmes donnés, il suffit dans le cas χ^2 de déterminer la probabilité de la transmission sans codage TEB_{IN} à partir de (2.69), puis de l'insérer dans le calcul de la probabilité d'erreur.

Des résultats issus de la simulation d'une chaîne de transmission utilisant les codes RS sur canal χ^2 sont présentés figure 2.22. Dans les simulations réalisées, le seuil de décision avant l'entrée du décodeur est choisi optimal de manière numérique. Les courbes sont présentées pour un rapport d'extinction infini, pour un ordre $n=2$ et pour trois codes différents RS(511,495), RS(511,479) et RS(255,239), de rendement respectivement égaux à : $R=0,97$ et $R=0,937$ pour les deux derniers. Les résultats sont représentés en fonction du paramètre Q .

Sur la figure 2.22, on retrouve des résultats attendus, c'est-à-dire que pour les codes RS(511, K), de même taille, les performances sont meilleures pour un rendement plus faible (c'est-à-dire une redondance plus élevée). De la même manière, pour un même rendement ($R=0,937$), on retrouve le fait que le pouvoir de correction du code est meilleur pour une longueur plus grande.

Afin de compléter l'étude des codes RS sur canal χ^2 , les performances obtenues sur canal χ^2 par simulation sont par la suite comparées aux résultats issus de la formule théorique, ainsi qu'aux résultats précédemment obtenus sur canal AWGN.

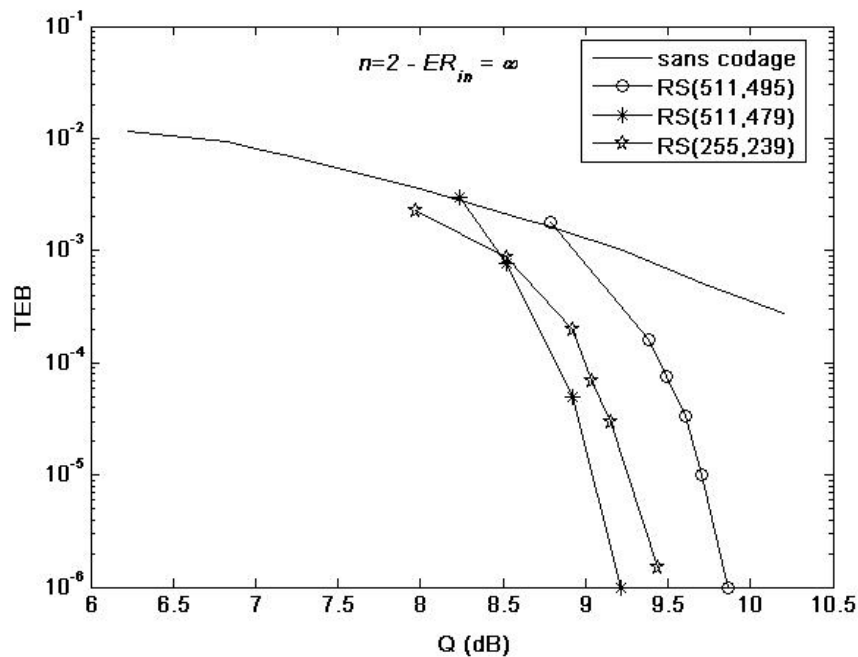


FIG. 2.22 – TEB obtenus sur canal χ^2 idéal (ER_{in} infini et $n = 2$) pour différents codes RS

2.5.2.2 Comparaison performances simulées et théoriques

Les courbes sur la figure 2.23 permettent de comparer les TEB obtenus par simulation avec ceux issus de la formule théorique, qui pour rappel, fournit le TEB codé à partir du TEB non codé. Ainsi, pour un cas de configuration donné (Q et ER_{in} fixés), on détermine théoriquement le TEB non codé à partir de (2.69) pour le seuil optimal, puis le TEB codé à partir de (2.53), (2.54) et (2.55). En simulation, on considère toujours l'utilisation du seuil optimal avant l'entrée dans le décodeur.

Les courbes révèlent, comme dans le cas du canal AWGN, une légère différence entre les performances théoriques et les performances obtenues par simulation. On peut en déduire que la formule théorique des performances des codes RS s'applique bien quelle que soit la statistique du canal et que le pouvoir de correction du code est donc indépendant de la

2.5 Performances avec le modèle χ^2

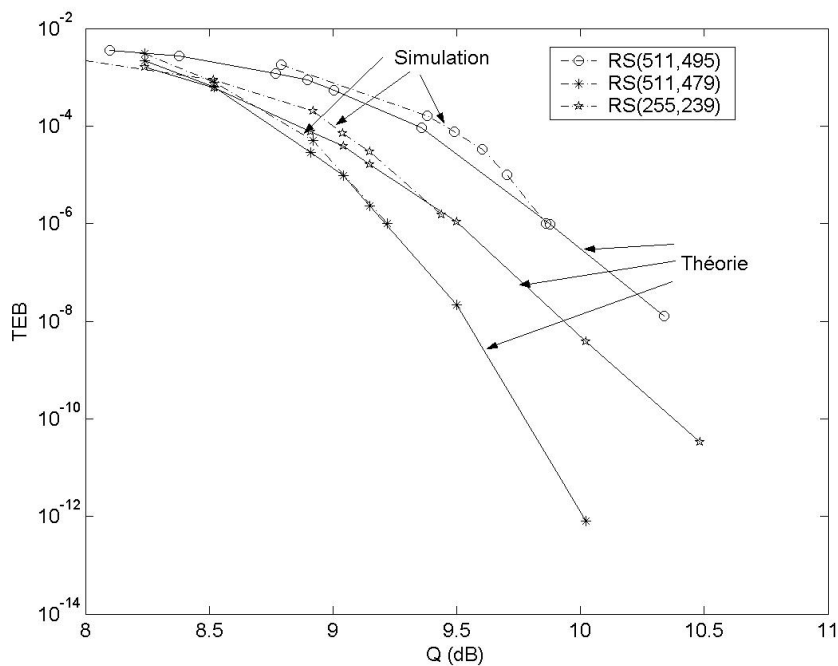


FIG. 2.23 – TEB obtenus sur canal χ^2 idéal (ER_{in} infini), $n = 2$ pour différents codes RS

statistique, à condition de choisir correctement le seuil optimal.

Des simulations ont été réalisées en appliquant un seuil classique de 0,5 avant l'entrée du décodeur. Les résultats sont reportés figure 2.24 pour le code RS(255,239) dans le cas d'un canal χ^2 avec ER_{in} infini et $n = 2$. Les TEB obtenus dans le cas du seuil optimal sont aussi reportés à titre de comparaison.

On remarque alors qu'avec le seuil gaussien de 0,5, les performances sont fortement dégradées par rapport au cas avec seuil optimal, et qu'elles le sont aussi par rapport au cas sans codage mais avec le seuil optimal. Ceci justifie l'importance de la prise en compte de la statistique du canal dans le choix du seuil de décision.

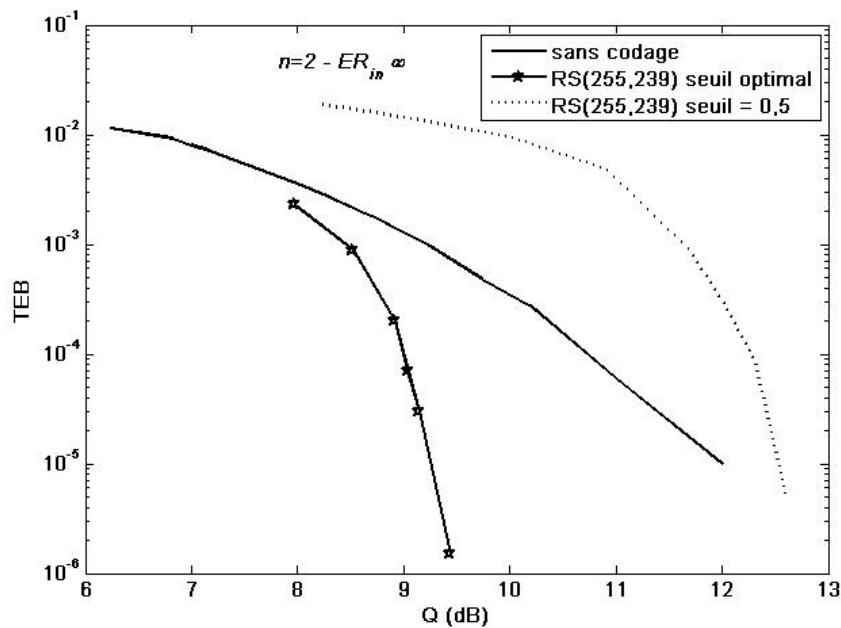


FIG. 2.24 – TEB obtenus avec le code RS(255,239) sur canal χ^2 (ER_{in} infini, $n = 2$) avec le seuil optimal et avec le seuil gaussien de 0,5

2.5.2.3 Comparaison performances RS sur canal χ^2 et AWGN

Afin de conclure sur les performances des codes RS sur un canal χ^2 , les performances sont comparées sur la figure 2.23 à celles obtenues sur canal AWGN avec le code RS(511,495).

A partir de ces courbes, on peut remarquer que les TEB codés et non codés sur canal χ^2 suivent une évolution identique à ceux obtenus sur canal AWGN, avec un décalage par rapport à l'axe Q . Ceci est conforme avec le fait que le pouvoir de correction est indépendant de la statistique du canal.

2.5.2.4 Conclusions

L'analyse des performances des codes RS a montré que ces codes sont aussi performants sur canal χ^2 en terme de réduction du TEB que sur canal AWGN. De plus, les

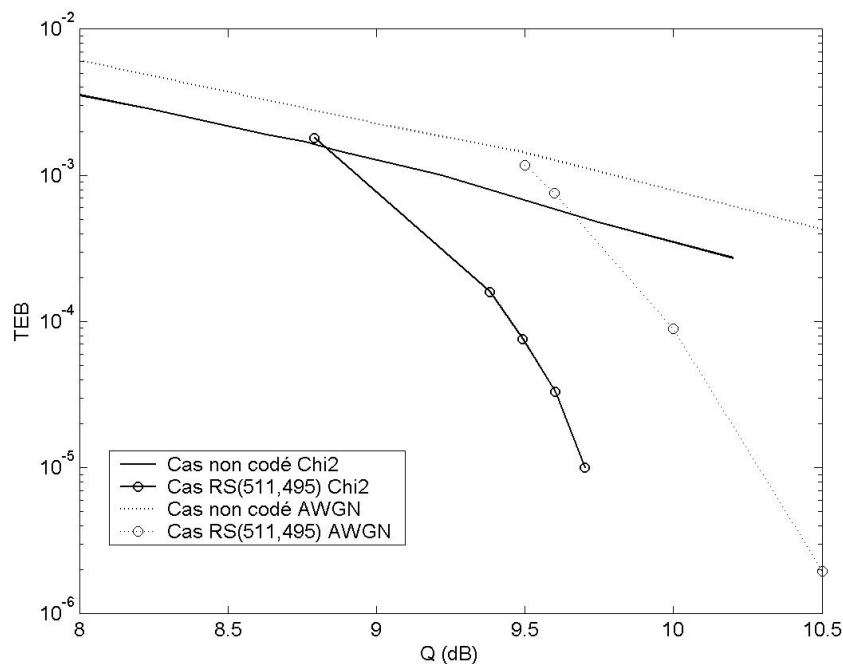


FIG. 2.25 – TEB obtenus avec le code RS(511,495) sur canal χ^2 (ER_{in} infini, $n = 2$) et canal AWGN

performances obtenues sont en accord avec les TEB approchés donnés par les formules théoriques ITU [23] qui ne spécifient pas le canal de transmission.

Ceci signifie que pour réaliser un dimensionnement, une première étape peut être effectuée à l'aide des formules théoriques et en considérant n'importe quel type de canal. L'étape suivante sera la validation par simulation sur canal χ^2 pour confirmer les prévisions théoriques.

2.5.3 Performances avec les codes LDPC-BIBD

2.5.3.1 Expression des LLR

Le rapport logarithmique de vraisemblance (ou LLR) s'évaluant à partir de la densité de probabilité de la variable reçue (2.56), il est nécessaire de le recalculer dans le cas

d'un canal χ^2 . Suivant que l'on se place dans le cas id al ou r el (c'est- -dire $A_0^2 = 0$ ou $A_0^2 \neq 0$), la quantit  $p(e = 0|Y)$ ne s'exprimera pas de la m me mani re.

Consid rons tout d'abord le cas g n ral r el o  $A_0^2 \neq 0$. On utilise alors les expressions des densit s de probabilit s (2.64) pour effectuer le rapport des deux valeurs et obtenir l'expression du LLR :

$$LLR = \ln \left(\frac{\left(\frac{1}{2\sigma^2} \left(\frac{Y}{A_0^2}\right)\right)^{(n-2)/4} e^{-(Y+A_0^2)/(2\sigma^2)} I_{n/2-1} \left(\sqrt{Y} \frac{A_0}{\sigma^2}\right)}{\left(\frac{1}{2\sigma^2} \left(\frac{Y}{A_1^2}\right)\right)^{(n-2)/4} e^{-(Y+A_1^2)/(2\sigma^2)} I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2}\right)} \right) \quad (2.74)$$

Certains termes communs au num rateur et au d nominateur permettent de simplifier l'expression en :

$$LLR = \ln \left(\left(\frac{A_1^2}{A_0^2}\right)^{\frac{n-2}{4}} e^{-\frac{Y+A_0^2}{2\sigma^2}} e^{\frac{Y+A_1^2}{2\sigma^2}} \frac{I_{n/2-1} \left(\sqrt{Y} \frac{A_0}{\sigma^2}\right)}{I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2}\right)} \right) \quad (2.75)$$

Finalement, le LLR peut s' crire dans le cas r el :

$$LLR = \frac{A_1^2 - A_0^2}{2\sigma^2} + (n/2 - 1) \ln \left(\frac{A_1}{A_0}\right) + \ln \left(\frac{I_{n/2-1} \left(\sqrt{Y} \frac{A_0}{\sigma^2}\right)}{I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2}\right)}\right) \quad (2.76)$$

Dans le cas id al o  $A_0^2 = 0$, l'expression de $p(e = 0|Y)$ prend une  criture plus simple. Le rapport des deux probabilit s donne alors :

$$LLR = \ln \left(\frac{\frac{1}{2^{n/2}\sigma^n \Gamma(\frac{1}{2}n)} Y^{n/2-1} e^{-Y/(2\sigma^2)}}{\left(\frac{1}{2\sigma^2} \left(\frac{Y}{A_1^2}\right)\right)^{(n-2)/4} e^{-(Y+A_1^2)/(2\sigma^2)} I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2}\right)} \right) \quad (2.77)$$

2.5 Performances avec le modèle χ^2

Comme précédemment, certains termes peuvent se simplifier et le LLR s'exprime :

$$LLR = \ln \left(\frac{Y^{(n-2)/4} A_1^{n/2-1} e^{A_1^2/(2\sigma^2)}}{2^{n/2-1} \sigma^{2(n/2-1)} (n/2-1)! I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2} \right)} \right) \quad (2.78)$$

D'où :

$$LLR = \ln \left(\left(\frac{\sqrt{Y} A_1}{2\sigma^2} \right)^{n/2-1} \frac{e^{A_1^2/(2\sigma^2)}}{(n/2-1)! I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2} \right)} \right) \quad (2.79)$$

Et, finalement :

$$LLR = \frac{A_1^2}{2\sigma^2} + \ln \left(\frac{\left(\frac{\sqrt{Y} A_1}{2\sigma^2} \right)^{n/2-1}}{(n/2-1)! I_{n/2-1} \left(\sqrt{Y} \frac{A_1}{\sigma^2} \right)} \right) \quad (2.80)$$

Les expressions des LLR dans les cas réel (2.76) et idéal (2.80) permettent d'initialiser le décodeur LDPC. Le reste du décodage s'effectue comme dans le cas classique.

2.5.3.2 Analyse des performances

a- Résultats de simulations

Des simulations ont été effectuées sur un canal χ^2 réel et idéal avec différents codes LDPC. Les résultats reportés sur la figure 2.26 ont été obtenus par simulation sur un canal réel χ^2 caractérisé par un rapport ER_{in} égal à 10 dB et $n = 4$.

Les courbes sont conformes aux résultats attendus, puisque les performances sont fortement améliorées, à partir d'une valeur minimale du TEB, dépendant du code envisagé. De plus, on remarque comme attendu que le code le plus redondant, c'est-à-dire le code LDPC(1810,1630) de redondance égale à 11% conduit à de meilleures performances que le code LDPC(2715,2535) de 7% de redondance.

D'autres résultats obtenus avec d'autres codes conduisent à des conclusions similaires.

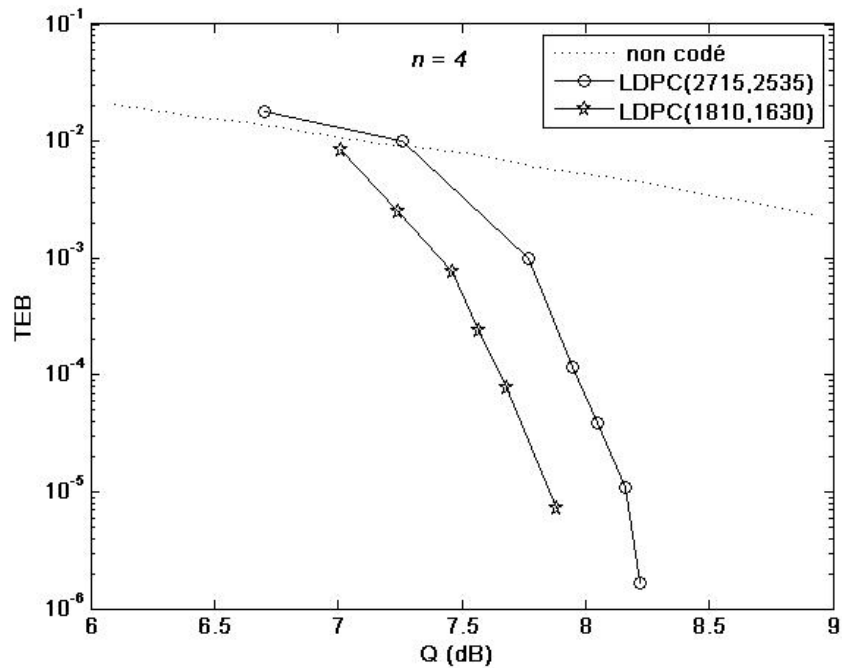


FIG. 2.26 – Performances obtenues avec les codes LDPC(2715,2535) et LDPC(1810,1630) sur un canal χ^2 avec $ER_{in}=10$ dB et $n=4$

Afin de compl ter l'茅valuation des performances des codes LDPC sur un canal χ^2 , les r sultats sont compar s dans les paragraphes suivants avec ceux obtenus sur canal AWGN, mais aussi sur canal χ^2 avec un d codeur non adapt  c'est- -dire bas  sur une statistique AWGN pour le calcul des LLR.

b- Comparaison avec les performances des codes LDPC sur canal AWGN

La figure 2.27 repr sente les performances obtenues par simulation en fonction du facteur Q sur un canal χ^2 avec $ER_{in} = 10$ dB et ER_{in} infini, et $n=4$, dans le cas non cod  et le cas avec LDPC(6176,5983). Les performances obtenues par simulation sur canal AWGN sont  galement report es.

Les courbes montrent que les performances sont  quivalentes, transpos es sur l'axe Q ,

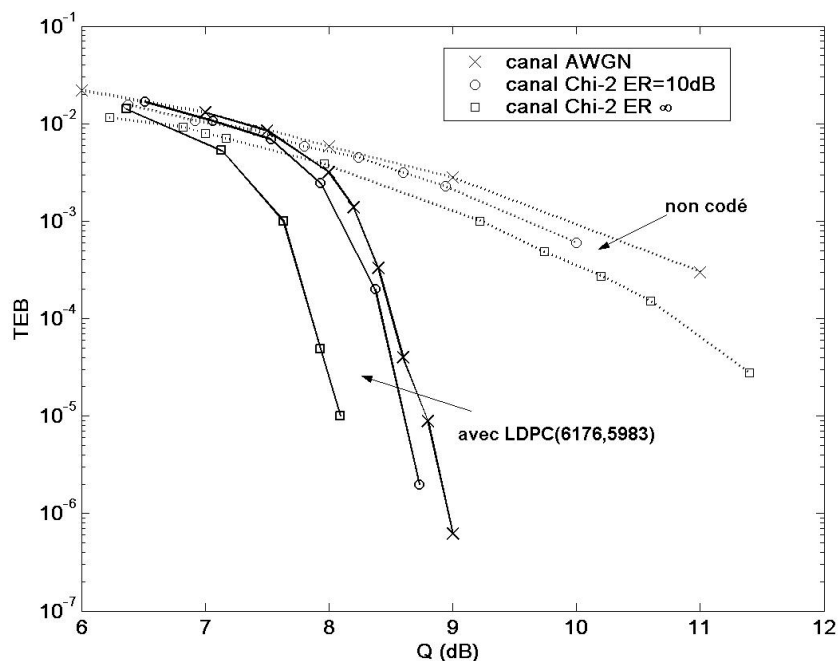


FIG. 2.27 – Comparaison des performances obtenues avec le code LDPC(6176,5983) sur canal χ^2 et canal AWGN pour $n = 4$

entre les cas AWGN et χ^2 . Cela permet de conclure que le pouvoir de correction des codes LDPC est le même sur un canal χ^2 que sur un canal AWGN.

De telles conclusions sont valables dans le cas où le calcul des LLR a été adapté au canal (2.76). En effet, ce n'est pas le cas si on considère l'initialisation du décodage basée sur une statistique gaussienne, comme illustré dans le paragraphe suivant.

c- Importance de l'initialisation des LLR

Des simulations ont été réalisées sans effectuer l'adaptation au niveau de l'initialisation des LLR, c'est-à-dire en supposant une statistique gaussienne. Les résultats de simulation sont présentés figure 2.28 pour le code LDPC(6176,5983) sur un canal χ^2 avec ER_{in} infini et $n = 4$ en fonction du facteur Q dans les deux cas suivants :

- d codeur AWGN : le d codeur est bas  sur une statistique gaussienne et les LLR sont calcul s par (2.61) (tandis que le canal est mod lis  par une statistique χ^2),
- d codeur Chi-2 : le d codeur est adapt  en utilisant les LLR calcul s pour une statistique χ^2 (2.76) (le canal est toujours mod lis  par une statistique χ^2).

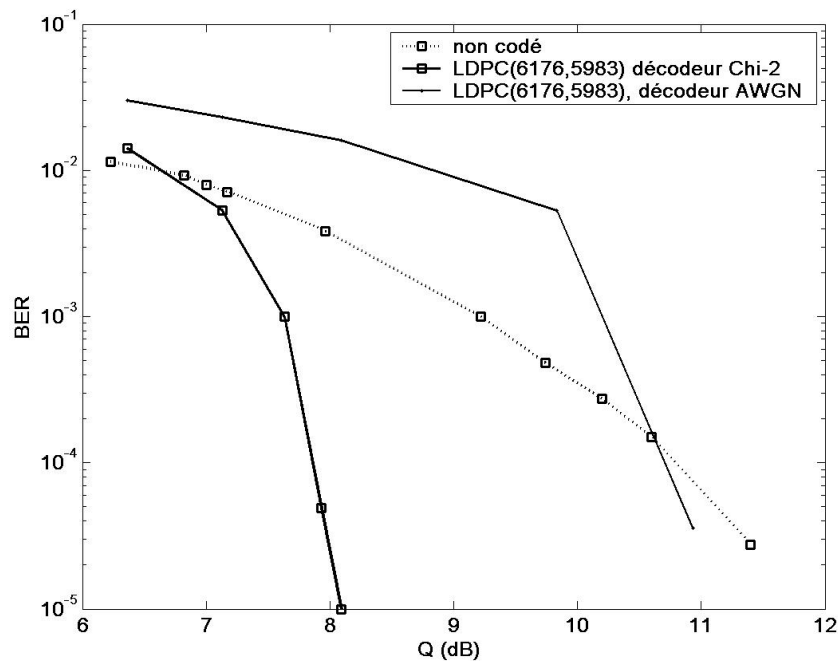


FIG. 2.28 – Performances avec le code LDPC (6176,5983) sur canal χ^2 avec ER_{in} infini et $n = 4$, avec calcul du LLR adapt  au canal χ^2 et avec le mod le classique AWGN

On remarque que sans adaptation du calcul des LLR, les performances obtenues avec le code LDPC sont s v rement d grad es. En effet, par exemple, un TEB de 10^{-5} est obtenu pour $Q = 11$ dB avec un d codeur bas  sur une statistique AWGN tandis qu'un facteur de qualit  de 8 dB est suffisant dans le cas d'un d codeur adapt . De plus, sans l'adaptation, ce n'est qu'  partir de $Q \approx 10,5$ dB (c'est   dire   partir d'un TEB non cod  de 10^{-4} que le codage apporte une correction. Or, avec le calcul ad quat des LLR, le code LDPC(6176,5983) am liore le TEB   partir de $Q = 6,5$ dB soit un TEB non cod 

de 10^{-2} .

De plus, pour des S/B plus faibles que celui correspondant au seuil de correction, le fait d'utiliser un décodeur "mal adapté" dégrade les performances par rapport au cas non codé. Ainsi, il paraît indispensable dans ce cas de prendre en compte la statistique du canal dans les paramètres nécessaires au décodage.

2.5.3.3 Conclusions

Les performances des codes LDPC sur canal χ^2 sont similaires à celles obtenues sur canal AWGN. Les codes LDPC sont alors, de la même manière que les codes RS, aussi performants quel que soit le type de canal. Cependant, une modification au niveau du calcul des LLR est à apporter pour le cas d'un canal χ^2 . En effet, si le décodeur est basé sur une statistique gaussienne, les performances sont dégradées.

Finalement, pour réaliser un dimensionnement, il est possible, comme pour les codes RS, d'utiliser un modèle AWGN, avant de transposer et de vérifier par simulation les résultats sur canal χ^2 .

2.5.4 Conclusions

Les performances d'une chaîne de transmission optique unipolaire ont été évaluées à partir d'une modélisation à statistique de type χ^2 , plus précise que la statistique gaussienne. Nous avons pu établir que le seuil optimal de détection pour une liaison simple sans codage correcteur d'erreurs est différent du seuil classique de 0,5 optimal pour un canal AWGN.

Par conséquent, afin d'obtenir les performances optimales avec un codage de type RS à décodage dur, il est important d'ajuster le seuil de décision. En ce qui concerne les performances obtenues avec les codes LPDC utilisant un décodage souple basé sur la

propagation de croyance, il est important d'initialiser correctement les LLR à partir de la statistique.

Les résultats portant sur l'évaluation des LLR ont fait l'objet de différentes publications [65–68]. Dans [65, 66], les résultats sont présentés dans le cadre de l'étude du réseau ECOFRAME, pour une valeur de $n = 4$ et en supposant que ER_{in} est infini. Dans [67, 68], l'étude est étendue au cas général où n peut prendre n'importe quelle valeur et où la statistique au niveau des '0' émis est associée à une distribution χ^2 non-centrée, c'est-à-dire avec ER_{in} quelconque. En tenant compte des spécificités relatives au canal χ^2 , les performances en terme de pouvoir de correction pour les codes RS ou LDPC sont identiques quel que soit le type de canal. Ceci signifie que pour réaliser un dimensionnement de code correcteur lié à des spécifications données en terme de TEB, il est possible de considérer le canal AWGN avant de transposer les résultats sur canal χ^2 .

Un exemple de dimensionnement paramétrique réalisé dans le cadre du projet de recherche ANR ECOFRAME [59] donné en annexe E permet d'illustrer ce point.

Les conclusions de cette étude sont tout d'abord que, le canal AWGN, plus simple à modéliser que le canal χ^2 , permet de dimensionner correctement les codes correcteurs, de type RS ou de type LDPC. Plus précisément, l'étude de dimensionnement des codes correcteurs sur canal AWGN a permis de mettre en évidence que :

- les codes RS peuvent être utilisés et que parmi eux le code (511,495) permet de maximiser le débit utile, tout en respectant le gain de codage minimal,
- les codes LDPC permettent de gagner, par rapport aux codes RS, 2dB de gain de codage. Ainsi, leur utilisation permet de garder une marge sur la valeur du S/B sur la liaison, ce qui permet de relâcher certaines contraintes de qualité sur les composants opto-électroniques et de diminuer ainsi le coût du réseau.

De plus, parmi les codes LDPC étudiés, tous permettent de satisfaire le gain de

2.6 Conclusion

codage minimal requis, et le choix peut se faire soit en maximisant le débit utile (LDPC(12742,12465)), soit en limitant la taille de la trame FEC et en choisissant le code LDPC(6176,5963). Ce dernier conduit en effet au même débit utile que le code RS retenu tout en assurant un gain de codage plus important.

Les résultats portant sur l'application des codes LDPC au réseau ECOFRAME ainsi que l'importance de la prise en compte de la statistique du canal au niveau de la réception ont fait l'objet de publications [65,66].

2.6 Conclusion

Dans ce chapitre, nous avons étudié les performances d'un système de transmission sur fibre optique, prenant en compte différentes perturbations sur la chaîne de transmission. Dans ce cadre, deux modèles (AWGN et à statistique χ^2) ont été étudiés pour décrire la transmission d'une modulation OOK sur canal unipolaire et basé sur une détection quadratique en réception. En tenant compte d'un bruit global additif gaussien sur la chaîne, nous avons vu que le canal à statistique χ^2 traduit plus fidèlement la distribution du signal reçu après photo-détection. Ce résultat a été confirmé lors des travaux de l'ANR ECOFRAME par des simulations physiques de la statistique du canal réel. Après une analyse théorique de chacun des deux canaux de transmission et des performances obtenues dans le cas non codé, deux types de codes correcteurs d'erreurs ont ensuite été envisagés : les codes RS et les codes LDPC construits à partir de la méthode BIBD. Les performances en terme de correction d'erreurs montrent que, comme attendu, les codes LDPC sont plus performants que les codes RS, et ce sur les deux types de canaux. En effet, on note un gain de codage à 10^{-12} de 2 dB supérieur pour les codes LDPC par rapport aux codes RS.

De plus, concernant l'étude des performances sur canal χ^2 , le résultat important porte sur la nécessité de prendre en compte la statistique au niveau du décodage. Cela se traduit pour le cas non codé et le cas avec les codes RS par une adaptation du seuil de décision au seuil optimal obtenu à partir de la description statistique du signal reçu. Concernant le décodage des codes LDPC utilisant un algorithme à propagation de croyance, la prise en compte du canal se traduit par le calcul de nouveaux rapports de vraisemblance (LLR), déterminés eux-aussi à partir de la statistique.

Finalement, en prenant en compte la statistique χ^2 particulière, les performances de chacun de ces codes se sont révélées conformes à celles obtenues sur canal AWGN. Cependant, les résultats de simulations ont permis de mettre en évidence l'importance de l'adaptation du décodeur, qui n'est plus efficace s'il est basé sur une statistique classique gaussienne. Le dimensionnement des codes RS et LDPC par rapport au contexte ECOFRAME, visant à réduire un TEB de 10^{-5} à 10^{-12} , a permis d'illustrer ces résultats et de conclure ce chapitre.

Le chapitre suivant porte sur l'application des codes correcteurs à une transmission optique utilisant la technique d'accès multiple par répartition de code (ou OCDMA : Optical Code Division Multiple Access). Cette technique est envisagée pour l'accès haut débit voire multimédia, le code servant alors à différencier à la fois les utilisateurs et le service. Dans ce contexte, le modèle χ^2 développé dans ce chapitre servira de base pour traduire fidèlement le processus de la photo-détection qui joue un rôle important, comme nous le verrons, sur les performances d'un tel système d'accès multiple.

Chapitre 3

Codage correcteur d'erreurs pour des transmissions sur fibre optique utilisant la technique OCDMA

Sommaire

3.1	Introduction	127
3.2	Principe de l'OCDMA	129
3.2.1	OCDMA incohérent	129
3.2.2	Emission	131
3.2.3	Réception	137
3.2.4	Application de l'OCDMA aux transmissions multimédia	142
3.3	Limitations	144
3.3.1	L'Interférence d'Accès Multiple	144
3.3.2	Le beat noise	148
3.3.3	Les autres limitations	175
3.3.4	Conclusions	179
3.4	Amélioration des performances avec codes correcteurs d'erreurs	180
3.4.1	Contexte	180
3.4.2	Application des codes LDPC à la chaîne de transmission OCDMA	182

3.4.3	Comparaison des performances des codes LDPC avec les codes RS199	
3.5	Application des LDPC à un réseau multiservice utilisant la technique OCDMA	201
3.5.1	Description du système	201
3.5.2	Performances du système multimédia avec codes LDPC	206
3.5.3	Conclusions	212
3.6	Conclusion	213

3.1 Introduction

Le principe de la technique d'Accès Multiple par Répartition de Code (AMRC ou CDMA : Code Division Multiple Access) est d'affecter à chaque utilisateur un code particulier et unique. En multipliant à l'émission les données de chaque utilisateur par le code correspondant, cela permet de différencier les données. Au niveau de la réception, les informations émises par l'utilisateur désiré sont récupérées en effectuant une corrélation du signal reçu par son code.

L'application de la technique CDMA pour l'optique s'appelle OCDMA (Optical CDMA) et a été envisagée dès les années 80 [69, 70], notamment comme alternative aux techniques d'accès classiques pour les réseaux d'accès à très haut débit. Récemment, la technique OCDMA a été également étudiée pour différencier les TEB (donc les services), constituant ainsi une solution pour les transmissions multimédia [71–73].

D'une manière générale, les codes OCDMA sont définis en une dimension (1D) ou en deux dimensions (2D) dans les domaines temporel et/ou spectral. Les codes en une ou deux dimensions, [69, 70, 74–76] sont caractérisés par une longueur temporelle F qui correspond à la division du temps bit en F temps chips, sur lesquels seront émis ou non, des impulsions en fonction du code. Pour les codes en 2D, la deuxième dimension fait référence au nombre de longueurs d'onde nécessaires à l'émission.

Pour la mise en place d'un système OCDMA, on distingue deux catégories : l'OCDMA cohérent et l'OCDMA incohérent. L'OCDMA cohérent correspond au cas où l'émission et la réception sont basées sur l'amplitude et la phase du signal optique. Il est dans ce cas important qu'il n'y ait pas de fluctuation d'amplitude durant le temps chip. Or, du fait de la combinaison, partielle ou entièrement destructive, de différentes composantes spectrales, l'amplitude du signal peut varier sur le temps chip. Pour éviter ce phénomène, il est nécessaire que les composantes du signal restent en phase pendant un temps au

moins égal au temps chip T_c . Le temps de cohérence de la source τ_c , défini comme le temps maximal pendant lequel le signal reste en phase doit ainsi respecter la relation $\tau_c \geq T_c$. Il convient alors d'utiliser des sources très cohérentes. De telles contraintes sur la source rendent le système OCDMA cohérent relativement coûteux à l'émission, mais aussi en réception où une détection de phase est nécessaire. L'OCDMA incohérent, est au contraire uniquement basé sur la puissance optique du signal transmis et constitue alors une solution plus simple à mettre en oeuvre puisque la contrainte sur la cohérence de la source est relâchée.

Le système OCDMA étudié dans ce chapitre est un système OCDMA incohérent. Tandis que pour l'OCDMA cohérent, les codes peuvent être bipolaires (de part la prise en compte de la phase en réception), ils sont unipolaires dans le cas incohérent. Cependant, afin de garantir une cardinalité raisonnable (c'est-à-dire un nombre d'utilisateurs significatif), les codes unipolaires ne peuvent pas être strictement orthogonaux entre eux. Par conséquent, des interférences apparaissent entre les différents utilisateurs. Ces interférences constituent l'IAM (Interférence d'Accès Multiple), première limitation inhérente d'un système de transmission OCDMA incohérent [70].

De plus, en pratique, les interférences ne se combinent pas de manière parfaitement incohérente au niveau du récepteur mais partiellement ou entièrement cohérente. Ce phénomène de combinaison conduit après la photodétection à des battements, appelés dans le cadre de l'OCDMA, bruits de battements ou "beat noise". Le beat noise constitue la deuxième limitation majeure à prendre en compte lors d'une transmission OCDMA [77–85].

Il existe également d'autres sources de perturbation le long de la chaîne de transmission, provenant des composants opto-électroniques, et pouvant être modélisées par un bruit global de nature gaussienne qui s'ajoute au signal avant la photodétection.

3.2 Principe de l'OCDMA

Afin de compenser les erreurs introduites par l'IAM, le bruit gaussien et le beat noise, nous avons envisagé l'utilisation de codes correcteurs de type LDPC qui ont déjà montré un gain en performances significatif sur un canal optique unipolaire. Dans le cadre d'une transmission optique, nous avons vu au cours du chapitre 2, qu'il est important d'avoir un modèle statistique précis du signal reçu pour obtenir des performances optimales. Dans le cadre de l'OCDMA, et vu les différentes limitations liées à cette technique, la variable décisionnelle suit une statistique bien particulière qu'il est alors nécessaire d'étudier.

Dans la première partie du chapitre, le système OCDMA incohérent ainsi que son application aux transmissions multimédia sont présentés. L'impact des différentes limitations propres au canal OCDMA (IAM, beat noise, bruit gaussien) est ensuite abordé. Les performances théoriques sont déterminées dans les cas où on ne considère que l'IAM, puis l'IAM et le beat noise. Le bruit gaussien est ensuite pris en compte. Dans chaque cas, la statistique du signal photodéfecté est modélisée mathématiquement. Ces résultats sont par la suite utilisés pour évaluer les paramètres nécessaires à l'initialisation du décodeur LDPC. Les performances des codes LDPC sont ensuite évaluées dans différents cas de configuration. Finalement, les potentialités du système OCDMA utilisant ces codes correcteurs sont illustrées par une application à une transmission multimédia.

3.2 Principe de l'OCDMA

3.2.1 OCDMA incohérent

L'OCDMA incohérent est caractérisé par le fait que la détection est basée uniquement sur la puissance du signal. Il n'est donc pas nécessaire dans ce cas, et contrairement au cas de l'OCDMA cohérent, d'assurer une amplitude constante durant le temps chip, donc d'utiliser des sources possédant un grand temps de cohérence.

Ainsi, dans le cadre de systèmes OCDMA incohérents, on s'intéresse aux sources telles que $\tau_c \leq T_c$. Plus le temps de cohérence est faible, plus le nombre de composantes spectrales présentes dans le signal augmente et plus les fluctuations d'amplitude du signal durant le temps d'émission sont importantes. En effet, les différentes composantes possèdent des phases variables et peuvent se combiner de manière destructive (partiellement ou complètement dans le pire des cas, c'est-à-dire en opposition de phase). Ceci implique que la durée effective (amplitude non nulle) T_p de l'impulsion émise dans l'intervalle de durée T_c est d'autant plus faible que le nombre de composantes est élevé (source large bande). Pour les sources laser envisagées, ce temps effectif T_p est environ égal au temps de cohérence τ_c et on aura donc : $T_p = \tau_c = 1/\Delta f$ où Δf correspond au spectre d'émission de la source ou encore à sa bande passante optique.

L'OCDMA purement incohérent correspond au cas où T_p est très faible devant T_c ($T_c/T_p \gg 1$). On considère de plus que le reste du système de transmission est idéal, de telle sorte que le signal reçu ait toujours la même durée effective T_p (fibre optique idéale, de bande passante infinie). Dans ce cas, la sommation en puissance des différentes interférences sur les chips peut ainsi avoir lieu et on dit alors que les interférences se combinent de manière incohérente d'où l'appellation OCDMA incohérent et régime incohérent.

En pratique, le rapport T_c/T_p n'est pas toujours très grand devant 1 puisque le spectre d'émission ainsi que la bande passante optique d'émission ou de réception sont limités. Ainsi, même si le système OCDMA est incohérent (c'est-à-dire basé sur la puissance optique), le régime n'est pas toujours parfaitement incohérent. En fonction de la valeur du rapport T_c/T_p on parlera donc de régime incohérent ($T_c/T_p \gg 1$), partiellement cohérent ($T_c/T_p > 1$), ou cohérent ($T_c/T_p \leq 1$). Avec les sources envisagées telles que $\tau_c = T_p \leq T_c$, le régime cohérent se limite donc au cas $T_c/T_p = 1$. Comme nous le verrons

3.2 Principe de l'OCDMA

par la suite, ce rapport a un impact sur la manière dont se combinent les interférences, et conduit à une limitation nommée beat noise, dont l'impact est plus ou moins important en fonction du rapport T_c/T_p [77–86].

3.2.2 Emission

3.2.2.1 Principe

Pour l'OCDMA incohérent, les codes utilisés qui peuvent être en une ou deux dimensions, sont unipolaires c'est-à-dire composés uniquement de '1' et de '0'. Au niveau de l'émission on considère une modulation OOK (On Off Keying) qui consiste à multiplier directement la donnée utile d'un utilisateur ('0' ou '1') par son code.

Ainsi, à l'issue de la multiplication, lorsque la donnée '1' est à émettre, le code de l'utilisateur est transmis (ce qui correspond à plusieurs intervalles de durée T_c). A l'inverse, lorsqu'il s'agit d'un '0', rien n'est transmis.

Si U utilisateurs transmettent simultanément des informations dans le réseau, U codeurs sont nécessaires et les données codées, notées $s_j(t)$ sur la figure 3.1, sont sommées pour former le signal $e(t)$ émis sur le canal de transmission. Les données aléatoires émises par les différents utilisateurs sont notées b_j et les codes 1D ou 2D sont représentés par $c_j(t + l_j)$, où l_j correspond au décalage temporel relatif à l'utilisateur j . Ce décalage aléatoire traduit le fait que les différents utilisateurs ne sont pas synchrones en temps bits mais en temps chips, et commencent à émettre à des temps différents.

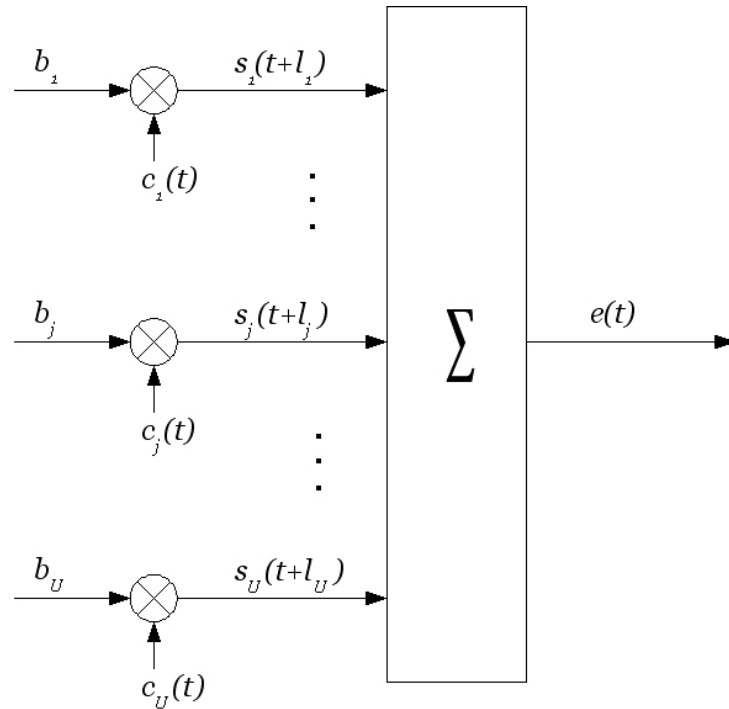


FIG. 3.1 – Principe d'émission OCDMA

3.2.2.2 Familles de codes

3.2.2.3 Présentation

Différentes familles de codes ont été envisagées pour l'OCDMA incohérent. Les plus connues sont celles issues des codes premiers (Prime Codes PC) [87] et des codes optiques orthogonaux (Optical Orthogonal Codes OOC) [70].

De plus, les codes sont définis soit en une dimension (1D) soit en deux dimensions (2D), dans les domaines temporel et /ou spectral. Si le code possède une dimension temporelle F , le temps bit T_b est décomposé en plusieurs temps chips de durée $T_c = T_b/F$. Si le code possède une dimension spectrale L , les impulsions de durée T_c sont émises sur L longueurs d'onde différentes, selon la règle de codage. Ainsi, sur un temps chip donné, un signal sera émis aux longueurs d'onde pour lesquelles le code 2D possède un '1' lorsque la donnée '1'

3.2 Principe de l'OCDMA

est émise tandis que rien n'est émis sur les longueurs d'onde affectées à '0'. Le poids W du code correspond au nombre de 1 dans la séquence de code. De plus chaque famille de code est caractérisée par l'autocorrélation ha qui traduit le degré de ressemblance d'un code avec lui-même, et l'intercorrélation maximale hc qui traduit la valeur maximale de ressemblance entre une séquence de code et toutes les autres de la famille de codes.

Parmi ces codes, l'étude se limite par la suite aux codes temporels OOC [70] pour les codes en 1D et aux codes temporels/spectral MWOOC (Multi-Wavelength OOC) [76] pour les codes en 2D, obtenus à partir des codes OOC. De plus, on considère que l'émission et la réception s'effectuent de manière synchrone en temps chips, ce qui correspond à la maximisation de l'Interférence d'Accès Multiple et donc à une borne supérieure lors de l'évaluation des performances.

Pour des codes parfaitement orthogonaux, l'intercorrelation entre deux codes distincts est nulle, ce qui peut s'exprimer pour tout i, j, l , avec $i \neq j$:

$$I_{i,j} = \sum_{t=0}^{F-1} c_i(t) \cdot c_j(t+l) = 0 \quad (3.1)$$

Pour $i = j$, on a :

$$I_{i,i} = \sum_{t=0}^{F-1} c_i(t) \cdot c_i(t+l) \begin{cases} = W & \text{si } l = 0 \\ \leq W & \text{si } l \neq 0 \end{cases} \quad (3.2)$$

Cependant, pour les codes optiques unipolaires envisagés, une orthogonalité parfaite n'est pas possible. L'intercorrélation, n'est alors pas nulle et l'équation (3.1) n'est pas toujours vérifiée. On définit alors la valeur de l'intercorrélation maximale h_c entre deux codes différents $i \neq j$ telle que pour tout i, j : $I_{i,j} \leq h_c$.

3.2.2.4 Les codes 1D OOC

Un exemple de représentation de code 1D affecté à un utilisateur est donné sur la figure 3.2, pour $F = 16$ et $W = 3$.

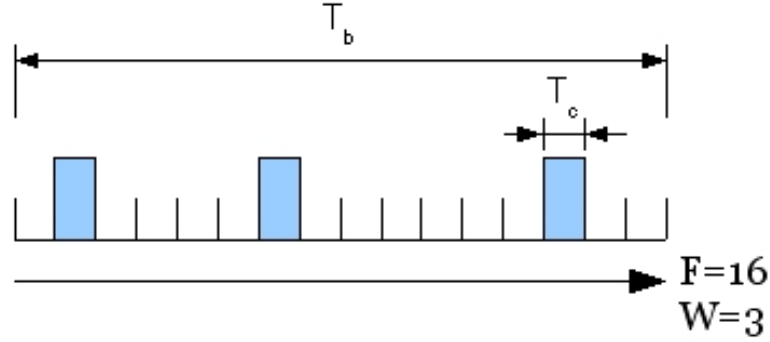


FIG. 3.2 – Exemple de séquence de code en 1D

Les codes OOC (Optical Orthogonal Codes), généralement notés $OOC(F, W, ha, hc)$, peuvent avoir des valeurs d'autocorrélation et d'intercorrélation quelconques. Cependant, le nombre maximum de séquences de codes N_{OOC} est lié aux valeurs d'autocorrélation et d'intercorrélation maximale $ha = hc = h$, par [88] :

$$N_{OOC} = \left\lfloor \frac{1}{W} \left\lfloor \frac{F-1}{W-1} \left\lfloor \frac{F-2}{W-2} \cdots \left\lfloor \frac{F-h}{W-h} \right\rfloor \right\rfloor \right\rfloor \right\rfloor \quad (3.3)$$

où $\lfloor x \rfloor$ désigne la partie entière par valeur inférieure de x . Pour limiter l'IAM, la valeur d'intercorrélation maximale, c'est-à-dire la valeur maximale prise par $I_{i,j}$, $i \neq j$, est fixée à sa valeur non nulle la plus faible, c'est-à-dire $hc = 1$.

Dans ce cas la cardinalité de la famille de code $OOC(F, W, 1, 1)$, que l'on notera aussi $OOC(F, W)$ est simplifiée en :

$$N_{OOC} = \left\lfloor \frac{F-1}{W \cdot (W-1)} \right\rfloor \quad (3.4)$$

3.2 Principe de l'OCDMA

En d'autres termes, cela permet à U utilisateurs, $U \leq N_{OOC}$ de transmettre des informations simultanément en utilisant U codes parmi les N_{OOC} choisis de manière aléatoire.

Comme l'intercorrélacion n'est pas nulle entre les codes, deux séquences peuvent interférer avec la probabilité d'interférence donnée par [70] :

$$\frac{W^2}{F} \tag{3.5}$$

Pour obtenir la probabilité d'interférence d'un utilisateur sur l'utilisateur désiré, il faut prendre en compte le fait que le code d'un utilisateur peut interférer uniquement lorsque la donnée émise est un '1'. En considérant une émission équiprobable, la probabilité d'interférence est alors donnée par :

$$p_{OOC} = \frac{W^2}{2F} \tag{3.6}$$

Les interférences limitent les performances du système, d'autant plus que la quantité (3.6) est élevée. Ainsi, pour un poids W donné, il faut augmenter la longueur temporelle F pour diminuer la probabilité d'interférence. Cependant, l'augmentation de la longueur entraîne une diminution du débit utile D pour une bande passante électrique B_e fixée selon la relation :

$$D_c = F \cdot D \leq B_e \tag{3.7}$$

avec $D_c = \frac{1}{T_c}$ le débit chip. Ainsi, il existe un compromis entre débit utile et performances, ce qui limite l'utilisation des codes 1D. Le paragraphe suivant présente les codes 2D, qui permettent de relâcher les contraintes et donc d'augmenter les potentialités de la technique OCDMA.

3.2.2.5 Les codes 2D MWOOC

Parmi les différentes méthodes de construction de codes en 2D dérivés des codes en 1D OOC [74–76], nous nous focaliserons sur les codes 2D précédemment étudiés dans le groupe de recherche [76] qui possèdent une flexibilité sur le choix des paramètres F, W , et L . En effet, les codes MWOOC issus de [74] doivent posséder un nombre de longueurs d'onde L égal à la dimension temporelle du code $L = F$. Ceux présentés dans [75] possèdent une dimension spectrale L fixée par $h + 2$. Les codes proposés dans [76] ont, quant à eux, une dimension spectrale égale au poids : $L = W$, ce qui permet pour une taille F fixée d'avoir moins de longueurs d'onde utilisées qu'avec les codes issus de [74] tout en étant souple sur les valeurs prises par L , contrairement à [75]. De plus, avec cette méthode de construction, l'intercorrélacion et l'autocorrélacion maximales sont égales à 1, tout comme pour les codes 1D OOC.

La cardinalité est alors donnée par :

$$N_{MWOOC} = F + L \tag{3.8}$$

Un code noté MWOOC($F, L, W = L$) est alors défini par une matrice de dimension $L \times F$ contenant des '1' et des '0'. Avec cette méthode de construction, chaque ligne, correspondant chacune à une longueur d'onde différente, possède un seul '1', ce qui établit que $W = L$. Ainsi, lorsque la donnée à émettre est un '1', après multiplication par la matrice de code, un signal est émis sur les L longueurs d'onde à un instant différent en fonction de la position du '1' sur chaque ligne.

Un exemple de matrice de code est représenté figure 3.3 pour $F = 23$ et $W = L = 5$.

Comme les codes OOC en 1D, on peut établir la probabilité d'interférence entre deux

3.2 Principe de l'OCDMA

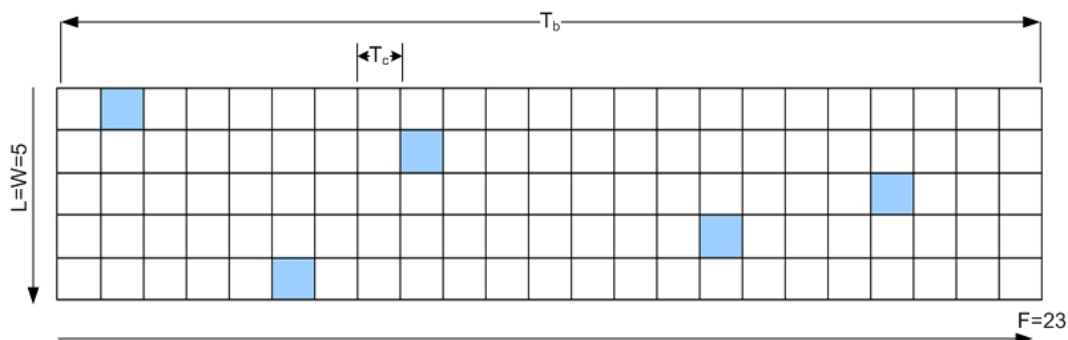


FIG. 3.3 – Exemple de séquence de code en 2D

utilisateurs différents en considérant une émission équiprobable [76] :

$$p_{MWOOC} = \frac{W^2}{2FL} = \frac{W}{2F} \quad (3.9)$$

Par rapport aux codes 1D, pour une même dimension temporelle F et un même poids W , la probabilité d'interférence est divisée par $L = W$. Ainsi, le code 2D se comporte d'un point de vue probabiliste comme un code 1D ayant une dimension temporelle équivalente égale à $L \times F$, soit L fois plus grande. Cette nouvelle dimension spectrale permet donc de relâcher la contrainte sur la longueur temporelle pour viser une performance donnée. Pour la même raison, les débits utiles seront plus grands en utilisant des codes 2D plutôt que des codes 1D.

3.2.3 Réception

3.2.3.1 Introduction

En réception, le signal détecté est composé de la somme des données codées $s_j(t)$ des différents utilisateurs. On considère la réception d'un seul utilisateur, appelé par la suite utilisateur désiré. Dans le cadre de l'OCDMA, les récepteurs mono-utilisateurs c'est-à-dire n'utilisant que le code de l'utilisateur désiré pour décoder l'information sont préférés aux

récepteurs multi-utilisateurs. En effet, ces derniers, qui nécessitent la connaissance des codes de tous les utilisateurs du réseau permettent une meilleure estimation des données de l'utilisateur désiré mais sont de fait plus complexes.

Dans tous les schémas de réception envisagés pour l'OCDMA, on retrouve le principe de la corrélation du signal reçu avec le code de l'utilisateur désiré. Ce principe de base, nommé réception conventionnelle par corrélation (CCR : Conventional Correlation Receiver) constitue la plus simple des techniques de réception. Après la présentation du CCR dans le paragraphe suivant, d'autres schémas de réception toujours basés sur ce principe sont également présentés.

3.2.3.2 Réception par corrélation

En réception, on considère que le signal reçu $r(t)$ est corrélé avec le code de l'utilisateur désiré $c_d(t)$, avec ici une synchronisation en temps bit pour l'utilisateur désiré (le décalage l_d est nul). Une fois l'intégration réalisée sur le temps bit T_b , une décision est prise sur chaque donnée à l'aide d'un comparateur à seuil, dont la valeur est notée S . La figure 3.4 représente le principe de la réception par corrélation.

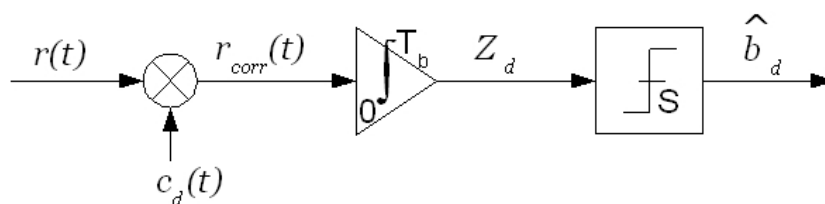


FIG. 3.4 – Réception par corrélation CCR

Chacun des signaux représentés sur la figure 3.4 est détaillé ci-après de manière mathématique. Dans le cas idéal, on considère que le signal reçu à l'entrée du récepteur $r(t)$ est égal au signal formé à l'émission c'est-à-dire à la somme des données codées des

3.2 Principe de l'OCDMA

différents utilisateurs, pour $j \in [1, U]$:

$$r(t) = \sum_{j=1}^U s_j(t) \quad (3.10)$$

En explicitant chaque donnée codée $s_j(t)$ comme le produit de la donnée émise par le code de l'utilisateur correspondant, $r(t)$ peut aussi s'écrire comme :

$$r(t) = \sum_{j=1}^U b_j \cdot c_j(t + l_j) \quad (3.11)$$

Ce signal est ensuite multiplié au code de l'utilisateur désiré $c_d(t)$ pour constituer le signal reçu corrélé $r_{corr}(t)$:

$$\begin{aligned} r_{corr}(t) &= r(t) \cdot c_d(t) = \left(\sum_{j=1}^U b_j \cdot c_j(t + l_j) \right) \cdot c_d(t) \\ &= b_d \cdot c_d(t) \cdot c_d(t) + \sum_{j=1/j \neq d}^U b_j \cdot c_j(t + l_j) \cdot c_d(t) \end{aligned} \quad (3.12)$$

L'intégration de ce signal sur le temps bit permet de constituer la variable décisionnelle correspondant à l'utilisateur désiré d et s'écrivant comme :

$$Z_d = \int_0^T r_{corr}(t) dt = \sum_{t=0}^{F-1} r_{corr}(t) \quad (3.13)$$

Avec la notation $I_{i,j}$ définie précédemment $I_{i,j} = \sum_{t=0}^{F-1} c_i(t)c_j(t+l)$, la variable

décisionnelle peut s'écrire :

$$\begin{aligned} Z_d &= b_d \cdot I_{d,d} + \sum_{j=1/j \neq d}^U b_j \cdot I_{d,j} \\ &= b_d \cdot W + \sum_{j=1/j \neq d}^U b_j \cdot I_{d,j} \end{aligned} \quad (3.14)$$

Dans le cas de codes parfaitement orthogonaux (3.1), la quantité $I_{d,j}$ est nulle pour tout $j \neq d$ donc la variable décisionnelle est uniquement exprimée par :

$$Z_d = b_d \cdot W \quad (3.15)$$

Seules les données relatives à l'utilisateur désiré sont présentes dans la variable décisionnelle, et la réception est réalisée sans erreur.

Dans le cas de quasi-orthogonalité, on définit l'interférence relative aux données de l'utilisateur désiré par :

$$I_d = \sum_{j=1/j \neq d}^U b_j \cdot I_{d,j} \quad (3.16)$$

La variable décisionnelle s'écrit alors :

$$Z_d = b_d \cdot W + I_d \quad (3.17)$$

On peut noter que, comme les données sont unipolaires, le terme d'interférence et la variable décisionnelle sont des termes positifs. La variable décisionnelle est ensuite comparée à un seuil S (compris entre 0 et W [89]) pour prendre une décision selon la

3.2 Principe de l'OCDMA

règle :

$$\hat{b}_d = \begin{cases} 1 & \text{si } Z_d \geq S \\ 0 & \text{si } Z_d < S \end{cases} \quad (3.18)$$

Dans ce cas, comme la variable décisionnelle comporte un terme d'IAM I_d , il peut exister des erreurs lors de la réception. La probabilité d'erreur correspondant à ce cas de figure est développée dans la partie 3.3.1.

3.2.3.3 Les autres récepteurs

a- Récepteur CCR avec limiteur

Afin de réduire la quantité d'IAM présente dans le signal reçu, la solution la plus simple consiste à ajouter un module limiteur avant le CCR, comme représenté sur la figure 3.5.

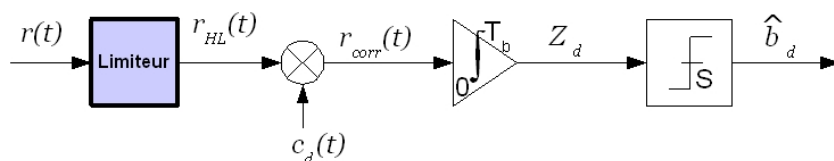


FIG. 3.5 – Récepteur CCR avec limiteur

Le fonctionnement de ce module "limiteur", généralement réalisé en optique (et dans ce cas appelé HL : Hard Limiter) peut être traduit par la fonction suivante :

$$hl(x) = \begin{cases} x & \text{si } x < 1 \\ 1 & \text{si } x \geq 1 \end{cases} \quad (3.19)$$

Ce module revient donc à écrêter à 1 le signal reçu sur chaque chip. Ainsi, la quantité d'interférence est diminuée par rapport au cas sans limiteur, ce qui a pour conséquence une amélioration de la probabilité d'erreur par rapport au CCR.

b- Récepteurs multi-utilisateurs

Les performances peuvent être encore améliorées si on utilise des récepteurs multi-utilisateurs. Un récepteur multi-utilisateurs nécessite par définition la connaissance au niveau du récepteur des codes des différents utilisateurs actifs. Le principe repose alors sur l'estimation des interférences des utilisateurs non désirés avant de soustraire au signal utile cette estimation de sorte à annuler les interférences. Parmi les différents récepteurs multi-utilisateurs, on peut citer les récepteurs ayant une structure parallèle (PIC : Parallel Interference Cancellation receiver) ou série (ou successif SIC : Successive Interference Cancellation receiver) [89]. Cependant, la complexité de tels schémas au niveau de la réception, qui nécessitent la connaissance des codes de tous les utilisateurs actifs, ainsi que le calcul des corrélations relatives à chacun des utilisateurs rend le système plus lourd. Ainsi, l'utilisation de récepteurs multi-utilisateurs ne sera pas envisagée par la suite, et l'étude se limitera au CCR.

3.2.4 Application de l'OCDMA aux transmissions multimédia

La technique OCDMA peut être utilisée dans le cadre d'applications multimédia, pour effectuer par exemple, la différenciation de TEB et la différenciation de débits. Afin de fournir des valeurs significatives de débits et de nombre d'utilisateurs, nous ne considérerons par la suite, pour les applications multimédia, que l'utilisation de codes 2D.

Pour différencier les TEB, la technique la plus simple est d'utiliser des codes de poids différents. En effet, pour une longueur temporelle fixée, plus le poids du code est élevé plus les TEB sont faibles. A partir des codes MWOOC, il est possible de former des codes multi-poids [76, 90] et de les affecter à des utilisateurs de classes différentes.

Considérons un code MWOOC($F, L, W = L$) comportant un nombre N_{MWOOC} de matrices de code différentes. Pour obtenir des codes multi-poids, on décompose le poids

3.2 Principe de l'OCDMA

L en différentes valeurs L_j telles que $\sum_{j=2}^m L_j = L$. Le nombre m correspond au nombre de poids différents ($m - 1$ sous-poids dans la décomposition plus le poids d'origine L). Sachant qu'il existe différentes manières d'effectuer la décomposition, m peut prendre plusieurs valeurs pour une valeur de L donnée.

Ainsi, m correspond au nombre de classes de services (caractérisées par le TEB) différentes.

La figure 3.6 représente deux exemples de répartitions possibles pour un code MWOOC avec 7 longueurs d'onde. Dans le cas de gauche, le poids est décomposé en 4 et 3 tandis que dans le cas de droite la décomposition est faite en sous-poids égaux à 5 et 2. Dans chacun des cas $m = 2$, mais les qualités de service (en termes de TEB) sont différentes, puisque les poids sont différents.

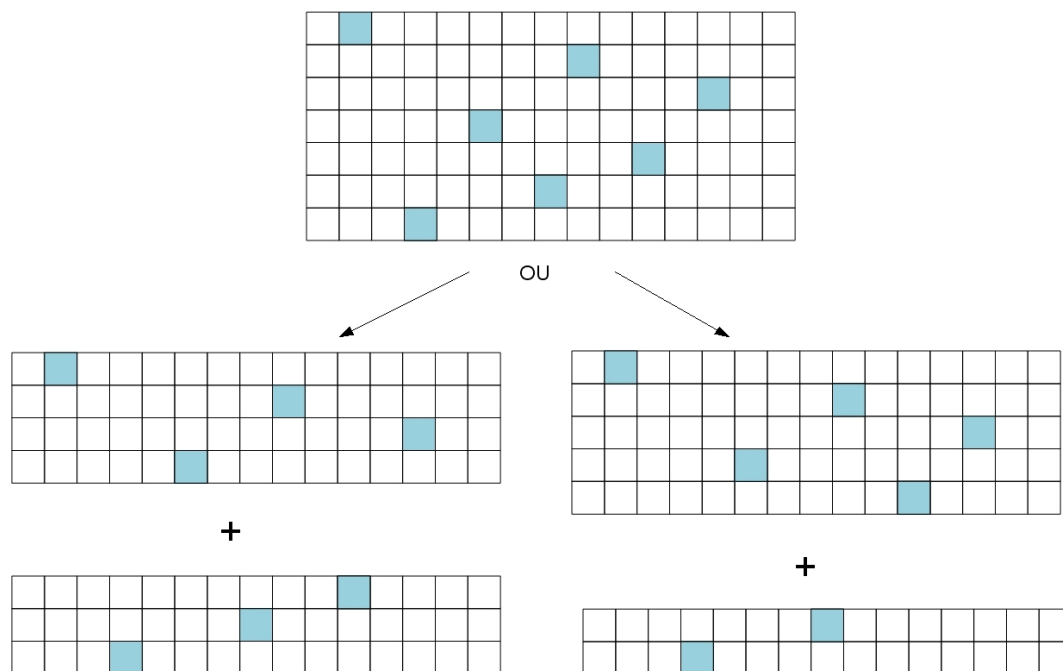


FIG. 3.6 – Exemple de partitionnement pour un code MWOOC avec $L = 7$

Concernant la différenciation de débits, une solution simple appelée "parallel mapping"

consiste à affecter plusieurs matrices de codes à un seul utilisateur qui de ce fait, peut envoyer simultanément plusieurs informations. Ainsi, le débit par utilisateur sera donné par le débit matrice multiplié par le nombre de matrices qui lui sont affectées.

En réception, il est alors nécessaire d'avoir autant de récepteurs par utilisateur que de matrices qui lui sont affectées, ce qui rend préférable l'utilisation de récepteurs le plus simples possibles, c'est-à-dire un CCR avec éventuellement un limiteur.

On peut également effectuer la différenciation de débit en utilisant des codes de longueurs temporelles différentes [76]. Cependant, cette technique a des performances limitées du fait de l'interférence introduite par les codes longs sur les codes plus courts.

L'application multimédia envisagée dans cette étude se focalise donc sur le parallel mapping pour la différenciation de débits.

3.3 Limitations

3.3.1 L'Interférence d'Accès Multiple

L'IAM constitue une limitation inhérente dans une transmission OCDMA utilisant des codes quasi-orthogonaux. Due à l'intercorrélation non nulle, et à la sommation en puissance des données sur chaque temps chip, la variable décisionnelle qui est un terme toujours positif, dépend du terme d'interférence I_d , dont la valeur est elle-même liée au nombre d'utilisateurs actifs ainsi qu'à la famille de code utilisée.

On peut montrer que la variable I_d représentant l'IAM sur le temps bit, suit une loi binomiale de paramètres ($p = p_{OOC}$ ou $p = p_{MWOOC}$, $U - 1$) [70,89]. Ainsi, I_d peut prendre

3.3 Limitations

des valeurs i entre 0 et $U - 1$ avec la probabilité suivante :

$$p(I_d = i) = \binom{U-1}{i} p^i (1-p)^{U-1-i} \quad (3.20)$$

La densité de probabilité de la variable I_d est donc de la forme :

$$p_{I_d} = \sum_{i=0}^{U-1} \binom{U-1}{i} p^i (1-p)^{U-1-i} \delta(I_d - i) \quad (3.21)$$

La variable décisionnelle, donnée par (3.17) : $Z_d = b_d \cdot W + I_d$, avec une émission équiprobable de '1' et de '0', a quant à elle une densité de probabilité exprimée par la demi-somme des densités de probabilités conditionnelles notées : $p_{Z_d}^{(0)}$ pour $b_d = 0$ et $p_{Z_d}^{(1)}$ pour $b_d = 1$.

Pour $b_d = 0$, $Z_d = I_d$ donc $p_{Z_d}^{(0)} = p_{I_d}$. Pour $b_d = 1$, $Z_d = W + I_d$ donc sachant que p_{I_d} s'exprime avec des fonctions de dirac, cela signifie que $p_{Z_d}^{(1)}$ est égale p_{I_d} décalée de W . Ainsi, on peut écrire la densité de probabilité de Z_d comme :

$$\begin{aligned} p_{Z_d} &= \frac{1}{2} \left(p_{Z_d}^{(0)} + p_{Z_d}^{(1)} \right) = \frac{1}{2} \left(\sum_{i=0}^{U-1} \binom{U-1}{i} p^i (1-p)^{U-1-i} \delta(Z_d - i) \right. \\ &\quad \left. + \sum_{i=0}^{U-1} \binom{U-1}{i} p^i (1-p)^{U-1-i} \delta(Z_d + W - i) \right) \end{aligned} \quad (3.22)$$

Un exemple de représentation des densités de probabilité conditionnelles $p_{Z_d}^{(0)}$ et $p_{Z_d}^{(1)}$ est donné figure 3.7 pour la famille OOC(70,3) et $U = 10$ utilisateurs.

On remarque alors que les densités $p_{Z_d}^{(0)}$ et $p_{Z_d}^{(1)}$ sont identiques, et décalées sur l'axe des

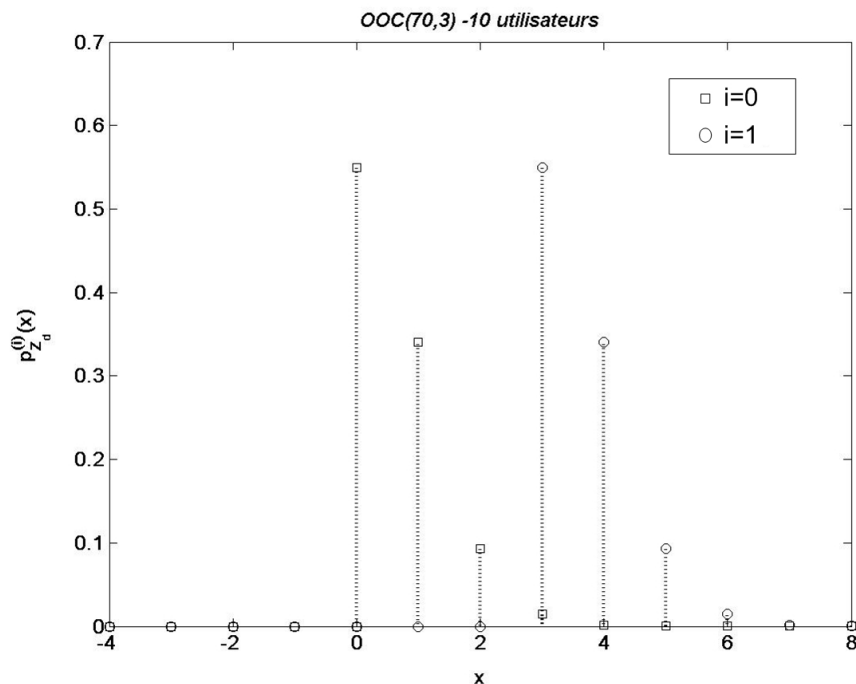


FIG. 3.7 – Exemple de densité de probabilité de la variable décisionnelle en présence d'IAM pour la famille OOC(70,3) avec 10 utilisateurs

x de la valeur $W = 3$. Ainsi, par exemple, $p^{(0)}(Z_d = 0) = p^{(1)}(Z_d = 3) = p(I_d = 0)$. Par conséquent, la densité de probabilité globale p_{Z_d} , s'écrivant comme la somme pondérée des deux, sera maximale en $x = 3$ où : $p(Z_d = W = 3) = \frac{1}{2} (p(I_d = 0) + p(I_d = 3)) \approx \frac{1}{2} \cdot p(I_d = 0) = p(Z_d = 0)$.

On remarque de plus dans cet exemple où l'IAM est peu importante (faible nombre d'utilisateurs), que la répartition binomiale décroît rapidement en fonction de x .

La moyenne m_I et la variance σ_I^2 de la variable binomiale d'IAM I_d sont données par [70] :

$$m_I = (U - 1) \cdot \frac{W^2}{2F} \tag{3.23}$$

$$\sigma_I^2 = (U - 1) \cdot \frac{W^2}{2F} \cdot \left(1 - \frac{W^2}{2F}\right) \approx (U - 1) \cdot \frac{W^2}{2F} \tag{3.24}$$

3.3 Limitations

A partir de ces considérations, il est possible d'évaluer la probabilité d'erreur du système OCDMA en considérant l'impact de l'IAM. Avec une réception CCR, en supposant une émission equiprobable, la probabilité d'erreur est exprimée d'une manière générale par :

$$P_{e,CCR} = \frac{1}{2}p(Z_d \geq S|b_d = 0) + \frac{1}{2}p(Z_d < S|b_d = 1) \quad (3.25)$$

En remplaçant Z_d par son expression (3.17), il vient :

$$P_{e,CCR} = \frac{1}{2}p(I_d \geq S) + \frac{1}{2}p(W + I_d < S) \quad (3.26)$$

Comme $I_d \geq 0$, la quantité $p(W + I_d < S)$ est toujours nulle pour $S \leq W$, ce qui signifie qu'aucune erreur n'est faite sur les '1' émis. En ce qui concerne le terme $p(I_d > S)$, il sera d'autant plus faible que le seuil est élevé. Ainsi, le seuil optimal est $S = W$. Dans les cas $0 < S \leq W$, la probabilité d'erreur s'exprime alors :

$$P_{e,CCR} = \frac{1}{2}p(I_d \geq S) = \frac{1}{2} \sum_{i \geq S} p(I_d = i) \quad (3.27)$$

Ainsi, la probabilité d'erreur avec une réception CCR s'exprime finalement, avec p donné par (3.6) pour les codes OOC ou (3.9) pour les codes MWOOC :

$$P_{e,CCR} = \frac{1}{2} \sum_{i \geq S} \binom{U-1}{i} p^i (1-p)^{U-1-i} \quad (3.28)$$

En considérant l'ajout du module limiteur en amont du CCR, la probabilité d'erreur

peut être bornée par l'expression suivante [70,89] :

$$P_{e, Lim+CCR} \leq \frac{1}{2} \binom{W}{S} \prod_{i=0}^{S-1} (1 - (1-p)^{U-1-i}) \quad (3.29)$$

Pour une même famille de code et un même nombre d'utilisateurs actifs, la probabilité du système est améliorée par rapport au cas du CCR [70,89].

3.3.2 Le beat noise

3.3.2.1 Contexte

Tout comme dans le cas d'une transmission optique filaire sans OCDMA, la loi quadratique de la photodétection qui revient à élever au carré le signal reçu conduit à des battements entre le signal utile et le bruit composé ici de l'IAM. Ces battements entre les interférences constituent le beat noise (ou bruit de battement). En utilisant la notation complexe, le champ reçu $E(t)$ peut s'écrire sur un intervalle de durée égale au temps chip T_c :

$$E(t) = \sum_{i=1}^U E_i \cdot e^{j\omega t + \phi_i} \quad (3.30)$$

où ϕ_i correspond à la phase relative de l'utilisateur i et U représente toujours le nombre d'utilisateurs actifs.

E_i correspond à l'amplitude de la contribution du i^{ime} utilisateur sur le temps chip considéré. Pour $i = d$ (l'utilisateur désiré), cela correspond à sa donnée utile, tandis que toutes les autres contributions (E_i pour $i \neq d$) constituent l'IAM. Le photocourant (ou la puissance optique) est défini à un facteur près (la sensibilité du photodétecteur,

3.3 Limitations

normalisée à 1) comme le module au carré du champ reçu (2.2). Il est donc de la forme :

$$|E(t)|^2 = \left(\sum_{i=1}^U E_i e^{j\omega t + \phi_i} \right) \cdot \left(\sum_{k=1}^U E_k^* e^{-j\omega t - \phi_k} \right) \quad (3.31)$$

Comme les amplitudes E_i sont des valeurs réelles, on a, en séparant les termes relatifs à l'utilisateur désiré d :

$$\begin{aligned} |E(t)|^2 = & E_d^2 + \sum_{i=1; i \neq d}^U E_i^2 + 2E_d \sum_{i=1; i \neq d}^U E_i \cdot \cos(\phi_d - \phi_i) \\ & + \sum_{i=1; i \neq d}^U \sum_{k=1; k \neq d; k \neq i}^U E_i E_k \cdot \cos(\phi_i - \phi_k) \end{aligned} \quad (3.32)$$

Les différents termes de l'expression (3.32) représentent :

- l'information utile émise par l'utilisateur désiré : E_d^2 ,
- le terme d'Interférence d'Accès Multiple : $\sum_{i=1; i \neq d}^U E_i^2$. L'IAM est présente si au moins un utilisateur non désiré interfère c'est-à-dire s'il existe i tel que $E_i \neq 0$,
- le beat noise primaire (entre les données de l'utilisateur désiré et celles des autres utilisateurs),
- le beat noise secondaire (entre les données des utilisateurs non désirés).

Les deux termes de beat noise dépendent de la différence de phase entre les champs provenant de différents utilisateurs sur un temps chip. Cette différence de phase a pour conséquence que les données se combinent de manière plus ou moins destructive. Ainsi, les termes de beat noise s'ajoutent (ou se soustraient) à la variable décisionnelle, ce qui modifie sa valeur et donc la prise de décision.

D'une manière similaire au cas sans accès multiple, pour quantifier les battements dus à la photodétection, on peut définir le rapport entre la bande passante optique du système B_o et la bande passante électrique B_e : $M = B_o/B_e$, avec M entier > 0 . La bande passante

optique correspond à la bande la plus étroite imposée par le système de transmission c'est-à-dire entre la bande passante de la source (spectre d'émission), la bande passante des éventuels amplificateurs en ligne et celle de l'éventuel filtre optique de réception. Elle est directement reliée à la durée effective T_p du signal émis durant un temps chip par la relation : $B_o = 1/T_p$. Si B_o correspond à la bande passante de la source (au spectre d'émission), $T_p = \tau_c$ avec τ_c le temps de cohérence de la source.

La bande passante électrique correspond au débit maximum des informations transmises sur la chaîne. Ce débit est limité par la capacité du récepteur électronique à échantillonner le signal reçu à une certaine fréquence. En pratique, cette période est égale au temps chip T_c et la bande passante électrique est fixée à sa valeur minimale : $B_e = 1/T_c = F/T_b$ où T_b représente le temps bit.

Ainsi on a la relation :

$$M = B_o/B_e = T_c/T_p \in \mathbb{N}^* \quad (3.33)$$

Par rapport aux études réalisées au cours du chapitre 2, nous avons pu établir que lorsque le rapport des bandes passantes M augmente, la statistique du canal χ^2 tend vers celle du canal symétrique gaussien. Ceci signifie que lorsque M augmente, les effets de battements dus à la photodétection deviennent peu importants. Ceci est en accord avec les résultats précédemment publiés [77–85], qui affirment que le beat noise a un impact d'autant plus important que la quantité M ($M \in \mathbb{N}^*$) est faible.

Ainsi, il est indispensable d'en tenir compte dans les systèmes OCDMA incohérents dès lors qu'on est en régime partiellement cohérent ou cohérent, c'est-à-dire lorsque la condition $M = T_c/T_p \gg 1$ n'est pas respectée. Cependant, pour des régimes dits incohérents avec des valeurs de M élevées, il est intéressant d'avoir une formulation

3.3 Limitations

générale de l'impact du beat noise afin de caractériser les dégradations du TEB.

Pour modéliser le canal en prenant en compte l'effet du beat noise, plusieurs études ont été menées, avec des modèles de distributions approchées différents. Dans [77, 78, 81, 86], le beat noise dans les systèmes OCDMA, à une ou deux dimensions, est modélisé par un bruit gaussien, ce qui est peu représentatif du phénomène de battements compte tenu de la loi de détection quadratique. Une autre approximation, appelée méthode du point col ("saddle point approximation"), est utilisée dans [80]. L'impact du beat noise n'y est étudié que pour des régimes cohérents ou partiellement cohérents c'est-à-dire pour des valeurs de M faibles, et il est négligé pour le régime incohérent. De la même manière, seul le régime cohérent a été traité dans [86]. Cependant, même si l'impact du beat noise est plus important dans les régimes cohérents et partiellement cohérents, il existe quand même pour le régime incohérent [81].

Afin de compléter ces travaux et d'obtenir une formulation générale de l'impact du beat noise sur les performances d'un système OCDMA, quelle que soit la nature du régime (c'est-à-dire quelle que soit la valeur de M), deux approches différentes ont été menées. La première est basée sur des considérations temporelles du signal transmis, tandis que la deuxième s'appuie sur une représentation spectrale et conduit à un modèle statistique de la variable décisionnelle. Ces deux approches ainsi que les modèles de canal correspondants sont développés dans les paragraphes suivants. Chacune de ces approches conduit à l'obtention d'une expression théorique de la probabilité d'erreur, valable pour toute valeur de M .

3.3.2.2 Premier modèle : approche temporelle

Description

Cette première approche se base sur l'expression analytique du photocourant (3.32) qui

fait apparaître des termes en $\cos(\delta\phi)$, où $\delta\phi = \phi_i - \phi_j$ correspond à la différence de phase entre des champs d'utilisateurs différents. Ce déphasage se traduit par un décalage dans le temps des impulsions dans le temps chip. En effet, si les champs correspondants à deux impulsions différentes ont la même phase, celles-ci débutent au même endroit dans le temps chip et la combinaison est entièrement constructive. Le terme de beat noise est dans ce cas maximal. Si les impulsions sont décalées dans le temps, le terme de beat noise s'annule en partie. Ce phénomène peut aussi se traduire par le fait que la durée effective des impulsions T_p n'est pas égale au temps chip T_c .

Dans ce modèle, on considère que le beat noise se traduit par l'élévation au carré des amplitudes résultantes de chacune des impulsions contenues dans des intervalles T_p . Ainsi, cela revient à considérer qu'il apporte une contribution uniquement positive, mais qu'elle est toujours maximale. Ce modèle simple permet alors d'aboutir à une borne supérieure des performances [82,83].

Un exemple est représenté figure 3.8. Sur chacun des chips de l'utilisateur désiré, le nombre d'interférents est 1, 4 et 3. Si on ne considère que l'impact de l'IAM sans prendre en compte la détection quadratique, cela conduit à une variable décisionnelle $Z_d = 8$. Cependant, si on tient compte de la conversion quadratique avec le modèle décrit plus haut, les différents interférents occupent seulement une partie du temps chip, et peuvent, ou non, interférer réellement avec les autres impulsions présentes. Dans l'exemple présenté figure 3.8, on suppose que la première impulsion occupe une position aléatoire dans le premier temps chip, tandis que sur le deuxième temps chip, les quatre impulsions sont supposées réparties sur deux positions conduisant à deux impulsions d'amplitudes résultantes égales à 1 et 3 (elles pourraient aussi être réparties sur quatre positions : 1 1 1 1, sur trois : 1 1 2 ou encore sur une seule : 4). Enfin, sur le dernier chip de l'utilisateur désiré, les trois impulsions interférentes sont réparties sur trois positions

3.3 Limitations

différentes. La variable décisionnelle est alors obtenue comme la somme au carré des amplitudes résultantes des différents groupes, c'est-à-dire $Z_d = 12$ dans cet exemple.

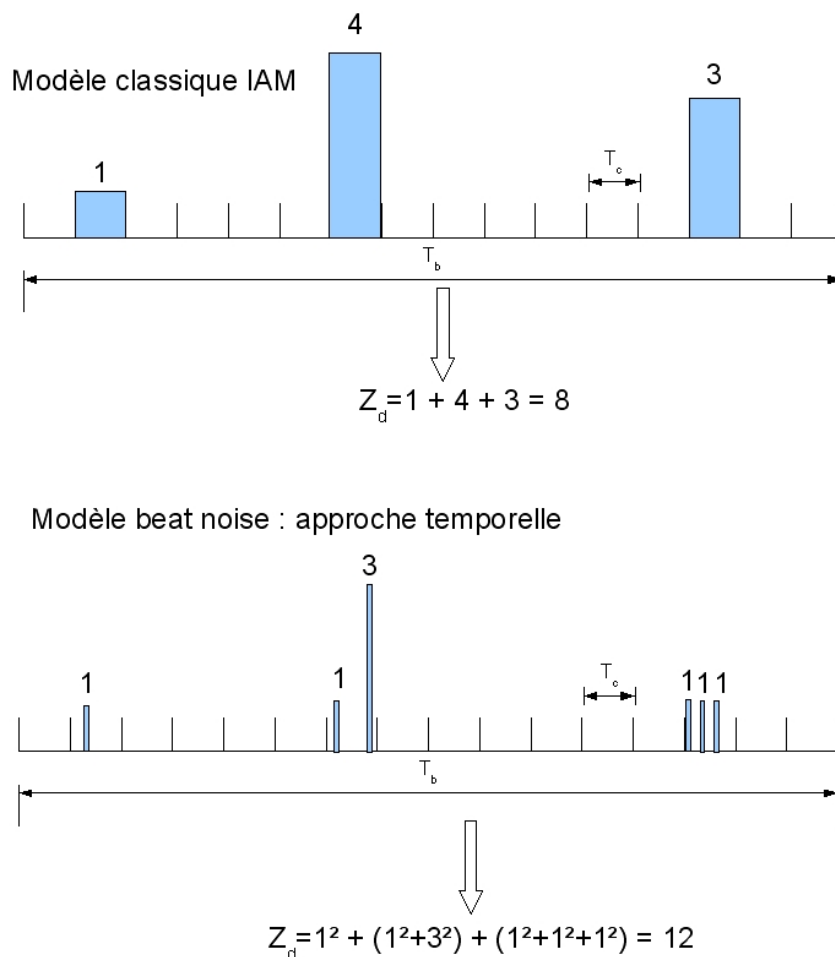


FIG. 3.8 – Modèle de beat noise, approche temporelle

Avec les notations précédentes (3.33), on peut écrire $T_c = M \cdot T_p$. Lorsque M devient grand, T_p est très petit face à T_c et les chips interférents ont une probabilité très faible de se superposer. Dans ce cas, l'élévation au carré en réception ne générera pas (ou peu) de beat noise. Dans l'exemple précédent, cela correspondrait au cas où toutes les impulsions sont sur des positions différentes ($Z_d = 1^2 + (1^2 + 1^2 + 1^2 + 1^2) + (1^2 + 1^2 + 1^2) = 8$).

Probabilité d'erreur

Pour évaluer la probabilité d'erreur, il faut partir de la formule générale (3.25) avec une variable décisionnelle telle que :

$$Z_d = b_d \cdot W + I_d + k_d = \sum_{m=1}^Q a_m^2 \quad (3.34)$$

avec

- k_d le terme de beat noise sur le temps bit, relatif à l'utilisateur désiré,
- Q le nombre de positions occupées par des groupes d'impulsions dans le temps bit, sur les chips de l'utilisateur désiré uniquement (dans l'exemple figure 3.8 : $Q = 6$),
- a_m l'amplitude des impulsions résultantes sur chacune de ces positions (dans l'exemple figure 3.8 : $a_1 = a_2 = 1$, $a_3 = 3$, $a_4 = a_5 = a_6 = 1$).

Notons alors que le terme d'IAM, sans prendre en compte les battements est donné par :

$$I_d = \sum_{m=1}^Q a_m \quad (3.35)$$

Avec (3.34), la probabilité d'erreur s'exprime de la manière suivante :

$$P_{e,CCR} = \frac{1}{2}p(I_d + k_d \geq S) + \frac{1}{2}p(W + I_d + k_d < S) \quad (3.36)$$

Sachant que dans le modèle, le beat noise n'apporte qu'une contribution positive (ou nulle), le terme $p(W + I_d + k_d < S)$ est nul pour $S \leq W$, et les erreurs n'auront donc lieu que sur les données '0' émises. Concernant le terme $p(I_d + k_d \geq S)$, comme I_d et k_d sont

3.3 Limitations

positifs, il peut se décomposer en :

$$\begin{aligned} p(I_d + k_d \geq S) &= p(I_d \geq S) + p(I_d < S \cap I_d + k_d \geq S) \\ &= p(I_d \geq S) + p(I_d < S) \cdot p(I_d + k_d \geq S | I_d < S) \end{aligned} \quad (3.37)$$

Le premier terme de (3.37) correspond à la probabilité d'erreur en présence d'IAM avec un CCR, et est donnée par (3.28). Le deuxième terme quant à lui peut s'écrire :

$$\begin{aligned} p(I_d < S) \cdot p(I_d + k_d \geq S | I_d < S) &= \sum_{j=0}^{S-1} p(I_d = j) \cdot p(j + k_d \geq S | j < S) \\ &= \sum_{j=0}^{S-1} \binom{U-1}{j} (p)^j (1-p)^{U-1-j} \cdot p(j + k_d \geq S | j < S) \end{aligned} \quad (3.38)$$

Finalement, la probabilité d'erreur se résume à :

$$\begin{aligned} P_{e,CCR} &= \frac{1}{2} \sum_{i \geq S} \binom{U-1}{i} p^i (1-p)^{U-1-i} \\ &+ \frac{1}{2} \sum_{j=0}^{S-1} \binom{U-1}{j} (p)^j (1-p)^{U-1-j} \cdot p(j + k_d \geq S | j < S) \end{aligned} \quad (3.39)$$

Ainsi, pour estimer l'impact du beat noise sur les performances du système, il reste à déterminer les différents cas possibles de répartition des impulsions sur le temps bit puis déterminer ceux qui conduisent à $j + k_d \geq S$, sachant que $j < S$, où pour rappel j représente la valeur prise par le terme d'IAM I_d .

Etant donné que les impulsions sont de durée $T_p = T_c/M$, chaque impulsion peut se positionner sur M places différentes dans un temps chip. Sachant qu'on ne considère que les W chips de l'utilisateur désiré, le nombre de places disponibles sur le temps bit pour

chaque impulsion est $W \cdot M$. Ainsi, le nombre de positions Q sur lesquelles les j impulsions interférentes se répartissent est au maximum égal à $W \cdot M$, et reste forcément inférieur au nombre total d'impulsions j . Ceci se traduit par : $Q \leq \min \{j, W \cdot M\}$. Or, sachant que $M \geq 1$ et que dans la somme considérée (3.39), $j \leq S - 1$, avec $S \leq W$ on a toujours : $j \leq W \cdot M$. Donc la condition sur Q est finalement : $Q \leq j$.

Rappelons que dans le cas considéré (erreurs sur les '0' émis donc $b_d = 0$), les amplitudes des impulsions résultantes $a_m, m \in [1, Q]$ sur chacune des Q positions occupées vérifient $I_d = \sum_{m=1}^Q a_m$ (3.35) et $I_d + k_d = \sum_{m=1}^Q a_m^2$ (3.34).

Ainsi, la condition $I_d + k_d \geq S$ se traduit par $\sum_{m=1}^Q a_m^2 \geq S$. Il reste donc à déterminer tous les cas possibles de combinaisons des a_m pour en déduire ceux qui conduisent à $\sum_{m=1}^Q a_m^2 \geq S$.

Listons les valeurs pouvant être prises par les a_m c'est-à-dire les répartitions possibles des j impulsions interférentes sur Q positions :

- Sur la position 1, le nombre d'impulsions superposées, noté a_1 peut varier de 1 à j .
Pour une valeur fixée de a_1 , il y a alors $\binom{j}{a_1}$ combinaisons différentes.
- Sur la position 2, le nombre d'impulsions superposées, noté a_2 peut varier de 1 à $j - a_1$. Pour une valeur fixée de a_1 , il y a alors $\binom{j - a_1}{a_2}$ combinaisons différentes.
- Sur la position 3, le nombre d'impulsions superposées, noté a_3 peut varier de 1 à $j - a_1 - a_2$. Pour une valeur fixée de a_1 , il y a alors $\binom{j - a_1 - a_2}{a_3}$ combinaisons différentes.
- ...
- Sur la position Q , telle que $a_Q = j - \sum_{q=1}^{Q-1} a_q$, il n'y a alors qu'une combinaison

3.3 Limitations

possible.

Notons alors que le nombre de positions occupées Q ne peut pas être déterminé à l'avance, puisqu'il dépend des valeurs prises par chaque a_m . Il peut être obtenu de manière récursive avec la condition d'arrêt $j = \sum_{q=1}^Q a_q$.

Pour un nombre fixé Q , le nombre de combinaisons possibles pour répartir ces Q positions sur les $W \cdot M$ positions disponibles est $\binom{W \cdot M}{Q}$.

Ce décompte conduit à l'expression du nombre de cas possibles de répartition des j chips parmi les $W \cdot M$ places disponibles :

$$\begin{aligned}
 N(j) &= \sum_{a_1=1}^j \sum_{a_2=1}^{j-a_1} \dots \sum_{a_m=1}^{j-\sum_{q=1}^{m-1} a_q} \dots \sum_{a_{Q-1}=1}^{j-\sum_{q=1}^{Q-2} a_q} \binom{j}{a_1} \binom{j-a_1}{a_2} \dots \\
 &\dots \binom{j-\sum_{q=1}^{m-1} a_q}{a_m} \dots \binom{W \cdot M}{Q} = \sum_{Q \geq 1} N^{(Q)}(j) \quad (3.40)
 \end{aligned}$$

Chacun de ces cas arrive avec la probabilité $(\frac{1}{W \cdot M})^j$ puisqu'il y a j impulsions interférentes à positionner sur $W \cdot M$ places. Pour chacun de ces cas, l'interférence plus le terme de beat noise vaut : $I_d + k_d = \sum_{m=1}^Q a_m^2$. La probabilité pour un cas d'engendrer une interférence globale (IAM + beat noise) supérieure au seuil s'écrit alors : $p_s = p \left(\sum_{m=1}^Q a_m^2 \geq S \right)$ et vaut 0 ou 1 en fonction des quantités a_m .

On en déduit finalement que :

$$\begin{aligned}
 p(j + k_d \geq S | j < S) &= \sum_{Q \geq 1} N^{(Q)}(j) \cdot \left(\frac{1}{W \cdot M} \right)^j \cdot p_s \\
 &= \sum_{a_1=1}^j \sum_{a_2=1}^{j-a_1} \dots \sum_{a_m=1}^{j-\sum_{q=1}^{m-1} a_q} \dots \sum_{a_{Q-1}=1}^{j-\sum_{q=1}^{Q-2} a_q} \binom{j}{a_1} \binom{j-a_1}{a_2} \dots \\
 &\dots \binom{j-\sum_{q=1}^{m-1} a_q}{a_m} \dots \binom{W \cdot M}{Q} \cdot p_s \cdot \left(\frac{1}{W \cdot M} \right)^j
 \end{aligned} \tag{3.41}$$

D'où, en remplaçant dans (3.39), on obtient pour $0 < S \leq W$:

$$\begin{aligned}
 P_{e,CCR} &= \frac{1}{2} \sum_{i \geq S}^{U-1} \binom{U-1}{i} p^i (1-p)^{U-1-i} \\
 &+ \frac{1}{2} \sum_{j=0}^{S-1} \binom{U-1}{i} (p)^j (1-p)^{U-1-j} \cdot \sum_{a_1=1}^j \sum_{a_2=1}^{j-a_1} \dots \sum_{a_m=1}^{j-\sum_{q=1}^{m-1} a_q} \dots \\
 &\dots \sum_{a_{Q-1}=1}^{j-\sum_{q=1}^{Q-2} a_q} \binom{j}{a_1} \binom{j-a_1}{a_2} \dots \binom{j-\sum_{q=1}^{m-1} a_q}{a_m} \dots \binom{W \cdot M}{Q} \cdot p_s \cdot \left(\frac{1}{W \cdot M} \right)^j
 \end{aligned} \tag{3.42}$$

Sachant que dans ce modèle, la contribution globale (IAM+ beat noise) est considérée comme étant toujours positive, la probabilité d'erreur (correspondant à la probabilité d'erreurs faites sur les '0') est, de la même manière qu'en ne considérant que l'IAM, d'autant plus faible que le seuil S , compris entre 0 et W , est élevé. Ainsi, par la suite, on considèrera en fait toujours $S = W$.

Validation du modèle

Lorsque M tend vers l'infini, il a été vérifié numériquement que le deuxième terme de

3.3 Limitations

(3.42) tend vers 0. Ainsi, la probabilité d'erreur tend vers le premier terme de la somme, qui est aussi égal à la probabilité d'erreur du CCR ne considérant que l'impact de l'IAM. A l'opposé, plus M est faible, plus la valeur prise par le deuxième terme est importante. Ceci se traduit par le fait que le beat noise est d'autant plus important que le rapport des bandes passantes $M = B_o/B_e$ est faible alors qu'il est négligeable pour M infini, ce qui valide le modèle aux limites.

De plus, les résultats issus de (3.42) pour $S = W$ ont été comparés à des résultats précédemment publiés [81] et utilisant un modèle de beat noise basé sur une distribution gaussienne. Afin de se mettre dans les mêmes configurations que celles présentées dans [81], nous avons comparé les performances obtenues avec le code OOC ($F=2197$, $W=4$) et $M = 50$. Les résultats sont représentés figure 3.9 en fonction du nombre d'utilisateurs actifs U . A titre de comparaison, les performances obtenues sans prendre en compte le beat noise, mais en ne considérant que l'IAM sont aussi reportées figure 3.9.

On peut remarquer que les résultats issus du modèle développé et ceux présentés dans [81] sont tout à fait concordants, ce qui est normal, vu qu'ici $M = 50$ et le modèle gaussien est suffisant. Les TEB tendent dans ce cas vers une valeur limite, correspondant à la probabilité d'erreur sans beat noise, mais en considérant uniquement l'IAM.

3.3.2.3 Deuxième modèle : approche spectrale et modèle statistique

Description

Dans ce modèle, on ne fait plus d'hypothèse sur les valeurs prises par le déphasage. En effet, la base du raisonnement n'est plus fondée sur l'expression temporelle du signal reçu, mais sur des considérations spectrales, analogues à celles présentées au cours du chapitre 2. Etant donné que la photodétection revient à élever au carré le champ reçu, le modèle χ^2 peut s'avérer représentatif du beat noise, si on considère que l'IAM se comporte

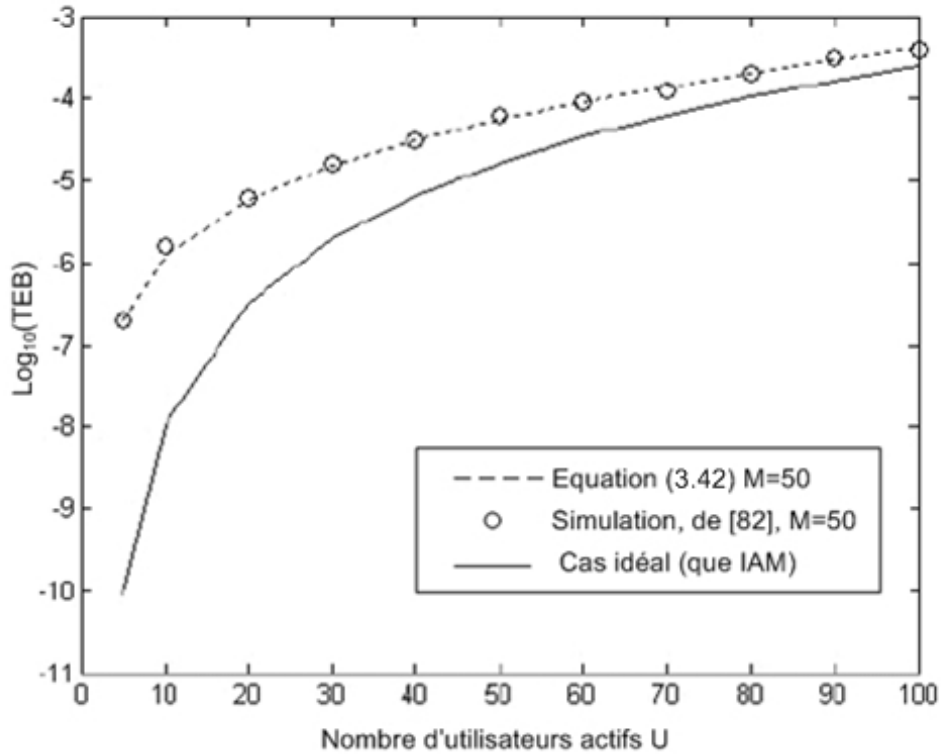


FIG. 3.9 – Performances obtenues avec la formule théorique (3.42), comparées à celles issues de [82] en fonction du nombre d'utilisateurs U pour $M = 50$

localement sur chaque temps chip comme une variable de type gaussienne. Comme le beat noise n'apparaît que lorsque plusieurs champs se superposent, le modèle χ^2 ne peut être utilisé que lorsque la puissance du chip est strictement supérieure à 1. Ainsi, si r_i représente la puissance (normalisée) du chip i , $i \in [0, F - 1]$ avant la détection et z_i sa valeur après, le modèle envisagé peut se représenter par la fonction suivante :

$$f : \begin{cases} r_i = 0 & \mapsto f(r_i) = z_i = 0 \\ r_i = 1 & \mapsto f(r_i) = z_i = 1 \\ r_i & \mapsto f(r_i) = z_i = \sum_{j=1}^n x_{ij}^2 \end{cases} \quad (3.43)$$

3.3 Limitations

Avec n l'ordre de la distribution χ^2 , $n = 2B_o/B_e = 2M$ et x_{ij} , variable gaussienne ayant une variance var et une moyenne m_{ij} données. Les valeurs prises par var et m_{ij} dépendent de la quantité d'IAM ainsi que de celle du beat noise caractérisé par le paramètre M :

- La moyenne m_{ij} de chaque variable x_{ij} est choisie identique pour toutes les valeurs de j , donc $m_{ij} = m_i$. Elle est déterminée de manière à respecter la puissance totale du chip : $\sum_{j=1}^n m_{ij}^2 = r_i$. Comme toutes les moyennes sont identiques, on a $\sum_{j=1}^n m_{ij}^2 = n \cdot m_i^2$ d'où :

$$m_i = \sqrt{\frac{r_i}{n}} = \sqrt{\frac{r_i}{2M}} \quad (3.44)$$

- La variance var des n variables x_{ij} dépend de la quantité d'IAM, caractérisée par sa variance σ_I^2 ainsi que du nombre de variables :

$$var = \frac{\sigma_I^2}{n} = \frac{(U-1)W^2}{2F \cdot 2M} \quad (3.45)$$

Le modèle étant entièrement décrit, il est maintenant possible d'établir la statistique du signal détecté afin de déterminer la probabilité d'erreur.

Probabilité d'erreur

L'expression (3.25) donne une définition générale de la probabilité d'erreur en fonction de la variable décisionnelle Z_d . Afin de déterminer les termes qui la constituent, la statistique de Z_d est recherchée. La variable décisionnelle est définie dans le cadre de l'OCDMA par (3.13), qui peut aussi s'écrire :

$$Z_d = \sum_{i=1/c_d(i)=1}^F z_i \quad (3.46)$$

En distinguant les cas où $r_i > 1$ des autres cas, l'expression de la variable décisionnelle devient :

$$Z_d = \alpha + \sum_{i=1}^{W'} \sum_{j=1}^n x_{ij}^2 \quad (3.47)$$

en notant α le nombre de chips tels que $r_i = 1$ et W' le nombre de chips tels que $r_i > 1$. Pour une valeur fixée de α , et pour $W' > 0$, la variable $Z - \alpha$ est définie comme une double somme de variables gaussiennes au carré et possède donc une statistique de type χ^2 . L'ordre de la distribution est dans ce cas donné par le nombre de termes dans la somme c'est-à-dire nW' . Le paramètre de non-centralité est défini par :

$$s^2(W', \alpha) = \sum_{i=1}^{W'} \sum_{j=1}^n m_{ij}^2 \quad (3.48)$$

Or, par définition (3.44), $\sum_{j=1}^n m_{ij}^2 = r_i$, ce qui simplifie l'expression du paramètre de non centralité (3.48) :

$$s^2(W', \alpha) = \sum_{i=1}^{W'} r_i \quad (3.49)$$

Ainsi, on peut en conclure que la variable décisionnelle Z_d suit également une distribution de nature χ^2 , d'ordre nW' et de paramètre de non-centralité donné par :

$$s_Z^2(W', \alpha) = \alpha + s^2(W', \alpha) = \alpha + \sum_{i=1}^{W'} r_i \quad (3.50)$$

Cette quantité (3.50) correspond à la somme de α (c'est-à-dire la somme des r_i tels que $r_i = 1$) et de la somme des r_i tels que $r_i > 1$. C'est donc finalement la somme des r_i , quelles que soient leurs valeurs soit :

$$s_Z^2(W', \alpha) = \sum_{i=1}^W r_i = b_d \cdot W + I_d \quad (3.51)$$

Ainsi, on peut exprimer la densité de probabilité de Z_d pour s_Z^2 fixé et $W' \neq 0$ à partir

3.3 Limitations

d'une fonction χ^2 non-centrée :

$$p_{s_Z^2, W' \neq 0}(Z_d) = \frac{1}{2var} \left(\frac{Z_d}{s_Z^2} \right)^{(nW'-2)/4} e^{-(s_Z^2 + Z_d)/(2var)} I_{nW'/2-1} \left(\sqrt{Z} \frac{s_Z}{var} \right) \quad (3.52)$$

W' prend des valeurs aléatoires entre 1 et W . Afin de simplifier l'étude et sachant que dans les cas considérés W est faible, les valeurs de nW' sont approchées par la suite par nW dans les cas où $s_Z^2 = b_d \cdot W + I_d > W$ et par n dans les autres cas.

Notons que lorsque $W' = 0$, la variable décisionnelle Z_d est une valeur entière, (égale à α) et sa densité de probabilité peut s'écrire sous forme de fonction de dirac :

$$p_{s_Z^2, W'=0}(Z_d) = \delta(Z_d - \alpha) \quad (3.53)$$

Finalement, à partir des expressions (3.52) et (3.53), la densité de probabilité de la variable décisionnelle est obtenue pour s_Z^2 fixé par :

$$p_{s_Z^2}(Z_d) = p(W' = 0 | s_Z^2) \cdot p_{s_Z^2, W'=0}(Z_d) + p(W' \neq 0 | s_Z^2) \cdot p_{s_Z^2, W' \neq 0}(Z_d) \quad (3.54)$$

En considérant maintenant que s_Z^2 peut prendre différentes valeurs (en fonction du terme d'interférence et de la donnée émise), l'expression générale de la densité de probabilité peut s'exprimer comme une somme pondérée, c'est-à-dire :

$$p(Z_d) = \sum_i p(s_Z^2 = i) p_i(Z_d) \quad (3.55)$$

Cette expression fait apparaître une somme pondérée de fonctions χ^2 , donc la variable décisionnelle se comporte finalement comme une mixture de χ^2 .

Afin de caractériser entièrement la densité de probabilité de la variable décisionnelle, il

reste à définir la quantité $p(W' = 0)$, qui dépend de la donnée émise. En effet, pour $b = 1$, chaque chip de l'utilisateur désiré r_i , i t.q. $c_i = 1$, possède une amplitude supérieure ou égale à 1, puisqu'il y a au minimum les chips correspondant à l'utilisateur désiré. Il existe alors au moins un chip d'amplitude supérieure à 1 (c'est à dire $W' \neq 0$) dès que le terme d'interférence I_d est non nul. Ainsi, on peut écrire :

$$p(W' = 0) = \begin{cases} 1 & \text{si } I_d = 0 \\ 0 & \text{si } I_d \geq 1 \end{cases} \quad (3.56)$$

Pour des données '0' émises, la détermination de $p(W' = 0)$ demande plus de considérations. Lorsque I_d vaut 1 ou 0, aucun chip ne peut avoir une amplitude strictement supérieure à 1, ce qui se traduit par :

$$p(W' = 0) = 1 \quad \text{si } I_d \in \{0, 1\} \quad (3.57)$$

Un autre cas évident est quand I_d est strictement supérieur à W . En effet, il existe alors nécessairement au moins un des W chips d'amplitude strictement supérieure à 1, ce qui s'écrit :

$$p(W' = 0) = 0 \quad \text{si } I_d > W \quad (3.58)$$

Pour les cas $2 \leq I_d \leq W$, le raisonnement est le suivant : la probabilité recherchée $p(W' = 0)$ étant la probabilité de n'avoir aucun chip d'amplitude strictement supérieure à 1, nous cherchons à évaluer pour I_d et W fixés, le rapport entre le nombre de cas possibles conduisant à n'avoir aucun chip d'amplitude strictement supérieure à 1 : n_0 , et le nombre total de cas de répartitions pouvant se présenter : n_t .

Le nombre n_0 correspondant aux cas où $I_d \leq W$, est donc égal au nombre de manières différentes pour choisir I_d éléments parmi W (qui seront les positions occupées par une

3.3 Limitations

seule impulsion). Il est donc donné par : $n_0 = \binom{W}{I_d}$.

Le nombre total de cas est quant à lui donné par le nombre de possibilités de choisir I_d éléments parmi $W + I_d - 1$, soit $n_t = \binom{W + I_d - 1}{I_d}$. En effet, les I_d impulsions peuvent se disposer n'importe où par rapport aux W positions, donc en fixant le premier chip à la première position, il reste $I_d + W - 1$ positions sur lesquelles ils peuvent se répartir.

Ainsi, la probabilité recherchée est donnée par le rapport n_0 sur n_t soit :

$$p(W' = 0) = \frac{\binom{W}{I_d}}{\binom{W + I_d - 1}{I_d}} \quad \text{si } 2 \leq I_d \leq W \quad (3.59)$$

Les équations (3.57), (3.58) et (3.59) décrivent entièrement la probabilité recherchée, c'est-à-dire $p(W' = 0)$ lorsque la donnée émise est un '0'. On peut ainsi obtenir également $p(W' \neq 0) = 1 - p(W' = 0)$.

En remplaçant ces expressions dans (3.54) puis dans (3.55), en utilisant le fait que $s_Z^2 = W + I_d$ pour $b_d = 1$ et $s_Z^2 = I_d$ pour $b_d = 0$, la densité de probabilité est entièrement déterminée. Avec la notation $p^{(1)}(Z_d)$ et $p^{(0)}(Z_d)$ pour la densité de probabilité pour un 1

ou un 0 émis, on a (voir développement en annexe F) :

$$p^{(1)}(Z_d) = p(I_d = 0) \cdot \delta(Z_d - W) + \sum_{i=1}^{U-1} p(I_d = i) \cdot p_{i+W, W' \neq 0}(Z_d) \quad (3.60)$$

$$p^{(0)}(Z_d) = p(I_d = 0) \cdot \delta(Z_d) + p(I_d = 1) \cdot \delta(Z_d - 1) + \sum_{i=2}^W p(I_d = i) \cdot \left(\frac{\binom{W}{i}}{\binom{W+i-1}{i}} \cdot \delta(Z_d - i) + \left(1 - \frac{\binom{W}{i}}{\binom{W+i-1}{i}} \right) \cdot p_{i, W' \neq 0}(Z_d) \right) + \sum_{i=W+1}^{U-1} p(I_d = i) \cdot p_{i, W' \neq 0}(Z_d) \quad (3.61)$$

Pour rappel, les expressions de $p(I_d = i)$ et de $p_{i, W' \neq 0}(Z_d)$ sont respectivement données par (3.20) et (3.52).

En considérant une émission équiprobable de '1' et de '0', la densité de probabilité de Z_d est obtenue avec (3.60) et (3.61) comme :

$$p(Z_d) = \frac{1}{2} (p^{(1)}(Z_d) + p^{(0)}(Z_d)) \quad (3.62)$$

Les densités de probabilités conditionnelles $p^{(1)}(Z_d)$ et $p^{(0)}(Z_d)$ sont représentées sur la figure 3.10 pour deux valeurs de n différentes. Le cas $n = 30$ représente le cas où il y a peu de beat noise, tandis que le cas $n = 4$ représente un cas où le beat noise est important.

On peut tout d'abord remarquer que les valeurs de $p^{(0)}(0)$, $p^{(0)}(1)$ et $p^{(1)}(W = 3)$ sont quasiment identiques entre les cas $n = 4$ et $n = 30$.

En effet, ceci est dû au fait que dans le modèle proposé, le beat noise n'a lieu que lorsqu'au moins deux impulsions interfèrent sur un temps chip. Or, dans les cas où un '0' est émis mais que $Z_d = 0$ ou $Z_d = 1$, et où un '1' est émis mais que $Z_d = 3$, chaque chip est au

3.3 Limitations

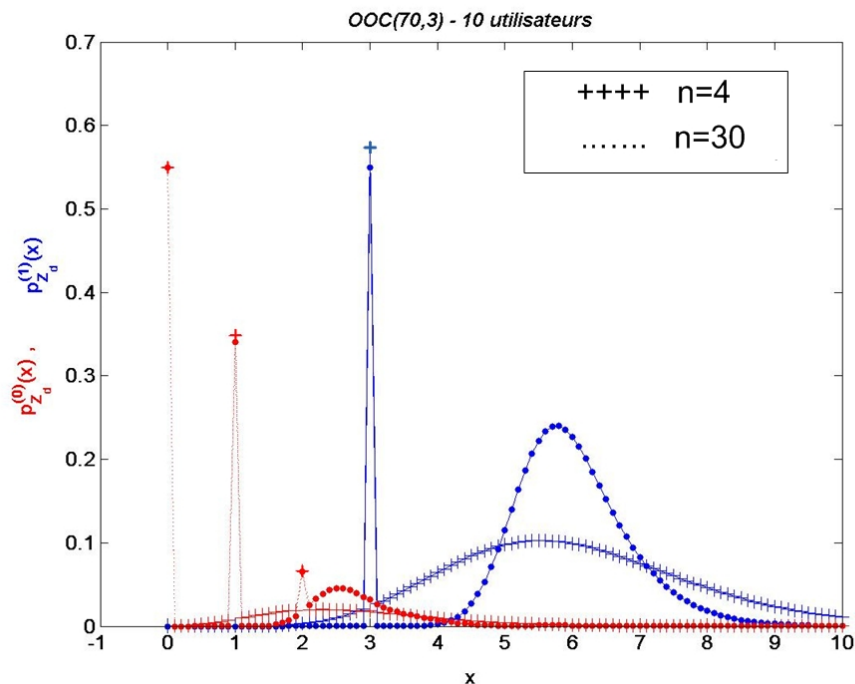


FIG. 3.10 – Densités de probabilité conditionnelles $p^{(1)}(Z_d)$ et $p^{(0)}(Z_d)$ pour $n = 4$ et $n = 30$

plus occupé par une seule impulsion, et il n'y a donc pas de beat noise, ce qui justifie que l'amplitude des raies présentes à ces positions là ne varie pas (ou peu) en fonction de la valeur de n . On retrouve en effet pour chaque valeur entière, les amplitudes des raies correspondant au cas idéal sans battements, augmentées des termes provenant des fonctions χ^2 , plus ou moins importants en fonction de n . Par exemple, dans le cas $n = 30$, l'amplitude de la raie de $p^{(1)}(W = 3)$ vaut la valeur qu'elle avait dans le cas sans beat noise (les battements ne sont plus importants), et se retrouve plus faible que dans le cas $n = 4$ où des termes traduisant les battements s'ajoutent.

Concernant les autres valeurs pouvant être prises par Z_d , on peut remarquer que le résultat de la somme pondérée de fonctions χ^2 donne une courbe relativement plate pour $n = 4$ tandis que des lobes plus étroits se forment pour $n = 30$. Ceci s'explique par le fait

que pour de faibles valeurs de n , les battements étant plus importants que pour n élevé, chacune des fonctions χ^2 constituant la mixture est plus dispersée, d'où un résultat plus plat.

Enfin, notons qu'à partir de ces courbes, qui ne sont ni symétriques, ni identiques pour les '0' et les '1' émis, il n'est pas possible de déterminer simplement le seuil optimal de détection. Le seuil optimal sera obtenu numériquement par la suite à partir de la probabilité d'erreur calculée dans les paragraphes suivants.

En effet, à partir de la densité de probabilité, il est possible de déterminer la probabilité d'erreur en partant de la définition générale donnée par (3.25), équivalente à la formulation intégrale suivante :

$$P_e = \frac{1}{2} \int_{-\infty}^S p^{(1)}(Z_d) dZ_d + \frac{1}{2} \int_S^{+\infty} p^{(0)}(Z_d) dZ_d \quad (3.63)$$

La densité de probabilité $p^{(1)}(Z_d)$ n'étant définie que pour les valeurs de Z_d positives (elle est nulle pour $Z_d < 0$), on peut simplifier : $\int_{-\infty}^S p^{(1)}(Z_d) dZ_d = \int_0^S p^{(1)}(Z_d) dZ_d$.

De plus, avec la condition $S \leq W$, les intégrales des fonctions de dirac se simplifient, puisqu'on a alors :

$$\int_0^S \delta(Z_d - \alpha) = 0 \quad \text{pour } S \leq \alpha \quad (3.64)$$

$$\int_S^{+\infty} \delta(Z_d - \alpha) = \begin{cases} 1 & \text{pour } S \leq \alpha \\ 0 & \text{pour } S > \alpha \end{cases} \quad (3.65)$$

En ce qui concerne l'intégrale des fonctions χ^2 , cela conduit, comme nous l'avons vu dans la partie 2.5.1, à des fonctions de Marcum généralisées définies d'une manière

3.3 Limitations

générale par (2.65).

Ainsi, avec ces considérations, la probabilité d'erreur s'exprime de la manière suivante, pour les '1' émis d'une part :

$$\begin{aligned}
 P_{e1} &= \int_0^S p^{(1)}(Z) dZ \\
 &= \sum_{i=1}^{U-1} p(I_d = i) \cdot \left(1 - Q_{nW/2-1} \left(\frac{\sqrt{i+W}}{\sigma}, \frac{\sqrt{S}}{\sigma} \right) \right)
 \end{aligned} \tag{3.66}$$

Et pour les '0' émis d'autre part, avec la contrainte $S > 1$ pour simplifier l'intégration des fonctions de dirac :

$$\begin{aligned}
 P_{e0} &= \int_S^{+\infty} p^{(0)}(Z) dZ \\
 &= \sum_{i=\max\{2, \lceil S \rceil\}}^W p(I_d = i) \cdot \left(\frac{\binom{W}{i}}{\binom{W+i-1}{i}} \cdot \int_S^{+\infty} \delta(Z-i) dZ \right) \\
 &\quad + \sum_{i=2}^W p(I_d = i) \cdot \left(1 - \frac{\binom{W}{i}}{\binom{W+i-1}{i}} \right) \cdot Q_{n/2-1} \left(\frac{\sqrt{i}}{\sigma}, \frac{\sqrt{S}}{\sigma} \right) \\
 &\quad + \sum_{i=W+1}^{U-1} p(I_d = i) \cdot Q_{nW/2-1} \left(\frac{\sqrt{i}}{\sigma}, \frac{\sqrt{S}}{\sigma} \right)
 \end{aligned} \tag{3.67}$$

Avec pour rappel dans les expressions (3.66) et (3.67), la probabilité $p(I_d = i)$ donnée par (3.20) : $p(I_d = i) = \binom{U-1}{i} p^i (1-p)^{U-1-i}$.

Ainsi, la probabilité d'erreur du système OCDMA avec beat noise est donnée en remplaçant (3.66) et (3.67) dans :

$$P_{e,CCR} = \frac{1}{2}P_{e1} + \frac{1}{2}P_{e0} \quad (3.68)$$

Validation du modèle

Pour valider ce modèle comme étant représentatif du beat noise, il est important tout d'abord de vérifier le comportement du système aux limites c'est-à-dire quand M tend vers l'infini correspondant au cas sans beat noise.

Pour une famille de code donnée et un nombre d'utilisateurs fixé, la variance des variables gaussiennes x_{ij} ne dépend que de M , et est inversement proportionnelle à M . Ainsi, lorsque M tend vers l'infini, la variance tend vers 0, ce qui signifie que chaque variable x_{ij} tend vers la valeur moyenne m_i .

La variable χ^2 résultante, $f(r_i) = y_i = \sum_{j=1}^n x_{ij}^2$ tend alors pour $M \rightarrow \infty$ vers $\sum_{j=1}^n m_{ij}^2 = r_i$ c'est-à-dire la valeur initiale du chip à l'émission. Ainsi, pour M infini, la fonction χ^2 ne change pas les valeurs des chips, ce qui signifie que le beat noise n'a pas d'impact.

A l'inverse, plus M devient faible, plus la variance des variables gaussiennes est importante et plus l'impact du beat noise sera important.

Afin de valider l'expression analytique obtenue, des simulations du canal décrit par (3.43) ont été réalisées. Les résultats simulés et théoriques sont comparés en termes de TEB figure 3.11 pour la famille OOC($F=534, W=3$) avec différents nombres d'utilisateurs actifs U . Les résultats sont tracés en fonction du rapport entre les bandes passantes M , avec le choix du seuil optimal minimisant la probabilité d'erreur. Les résultats simulés sont représentés par des cercles sur les courbes.

3.3 Limitations

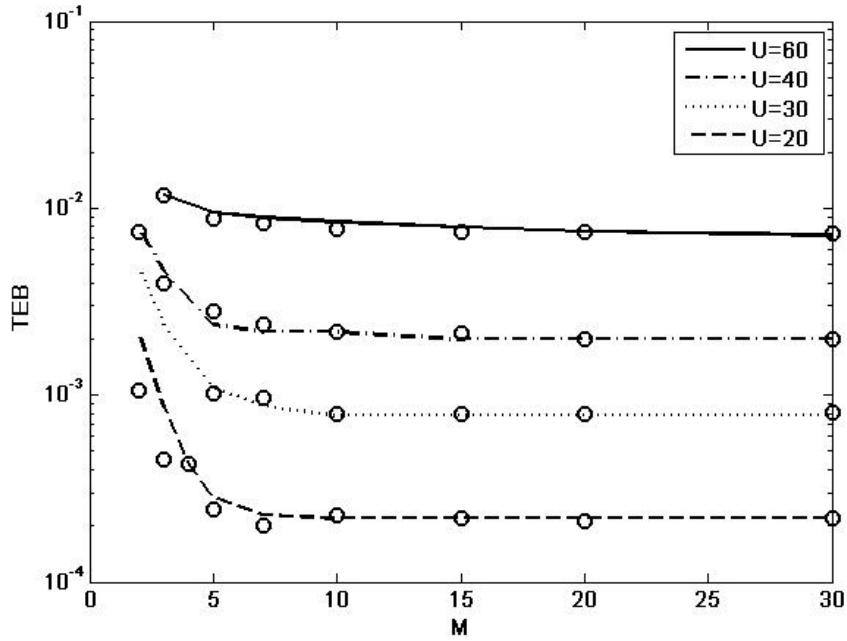


FIG. 3.11 – Performances obtenues avec la famille OOC($F=534, W=3$) avec (3.68) (trait plein) et par simulation (cercles) (3.43)

Les résultats obtenus en simulation et en théorie sont dans l'ensemble similaires. Pour un nombre limité d'utilisateurs actifs (cas $U = 20$), et pour les faibles valeurs de M , on note cependant une légère différence. Pour tous les autres cas, les résultats sont concordants, quelle que soit la valeur de M , ce qui justifie la validité de l'expression analytique (3.68) issue des considérations statistiques décrivant le canal.

Impact du seuil de décision

Afin d'illustrer l'impact du seuil de décision sur les performances, nous avons considéré la famille OOC ($F=1345, W=7$) permettant de supporter $U=32$ utilisateurs actifs à un TEB inférieur à 10^{-6} dans le cas idéal sans beat noise. Les résultats tracés sur la figure 3.12 sont obtenus pour différentes valeurs de M en fonction du seuil de décision S .

On peut tout d'abord remarquer que pour $M=5$, le seuil optimal est identique au seuil optimal sans beat noise c'est-à-dire $S_{opt} = W = 7$. Cette remarque a été faite pour

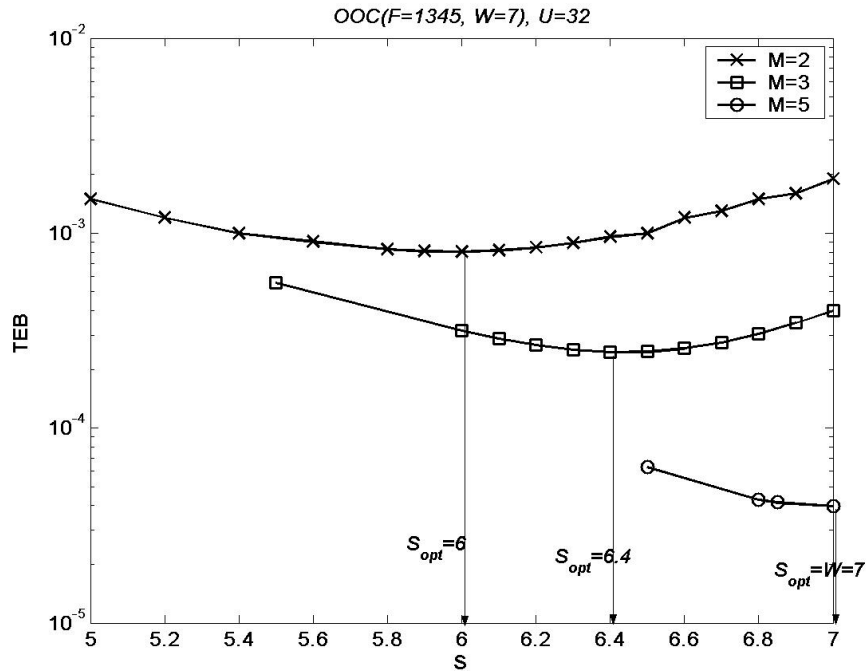


FIG. 3.12 – Influence du seuil de décision S sur le TEB pour différentes valeurs de M

tout autre valeur de M supérieure à 5 dans cet exemple, c'est pourquoi les TEB n'ont été reportés que pour des valeurs de $M \leq 5$.

On remarque alors que pour tous ces cas où le beat noise est important, le seuil optimal diffère du poids W et est d'autant plus faible que le beat noise est élevé. En effet, dans l'exemple présenté, on peut déduire de la figure 3.12 que S_{opt} vaut 6,4 pour $M=3$ et 6 pour $M=2$.

Au vu de l'importance du choix du seuil de décision sur les performances, tous les résultats présentés par la suite sont obtenus pour le seuil optimal.

Dans le paragraphe suivant, les performances sont comparées à celles obtenues avec le modèle précédent, de manière à conclure sur la validité de ce modèle en mixture de χ^2 ainsi que sur l'ensemble de l'étude réalisée sur le beat noise.

3.3 Limitations

3.3.2.4 Comparaison et conclusions

Les TEB présentés sur les courbes suivantes sont issus des formules théoriques (3.42) et (3.68) correspondant aux performances relatives aux deux modèles proposés pour traduire le canal OCDMA en présence de beat noise.

Les résultats sont présentés en fonction du rapport entre les bandes passantes M pour différents nombres d'utilisateurs, pour la famille OOC($F=70$, $W=3$) sur la figure 3.13 et pour la famille OOC($F=534$, $W=3$) sur la figure 3.14.

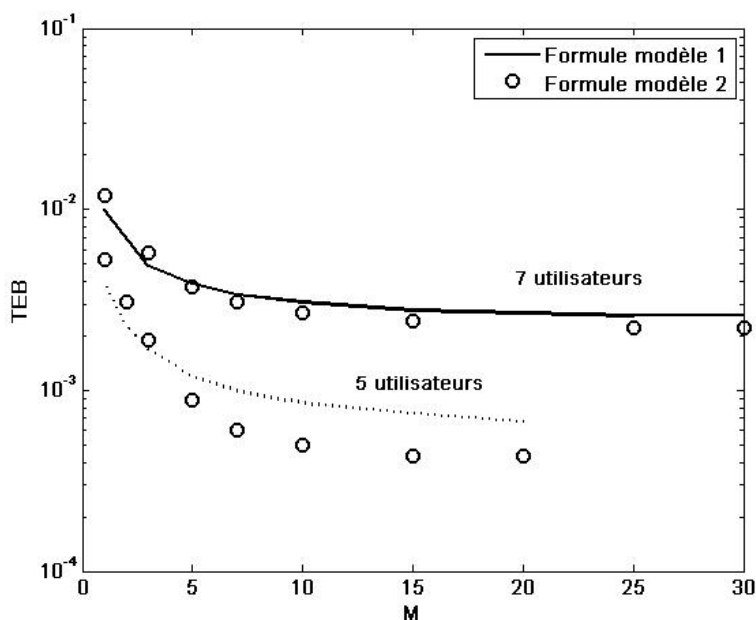


FIG. 3.13 – Comparaison des performances obtenues avec beat noise en fonction de M pour la famille ($F=70$, $W=3$) avec le premier modèle en trait plein, et le deuxième modèle avec les ronds

Sur chacune des courbes présentées, les TEB obtenus à partir du deuxième modèle (mixture de χ^2) sont meilleurs que ceux issus de la formule théorique du premier modèle, ce qui est attendu puisque le premier modèle conduit à une borne supérieure des performances. De plus, on peut remarquer que les courbes suivent la même évolution

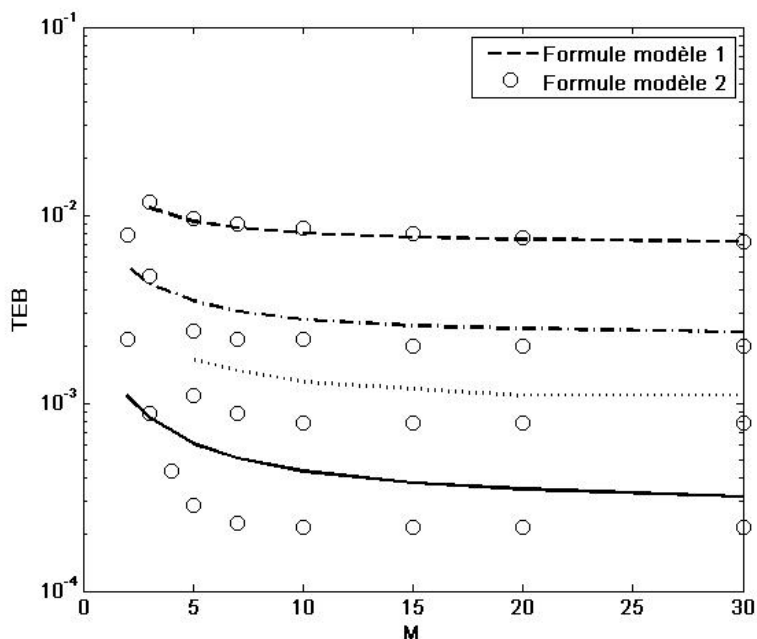


FIG. 3.14 – Comparaison des performances obtenues avec beat noise en fonction de M pour la famille ($F=534$, $W=3$) avec le premier modèle en trait plein, et le deuxième modèle avec les ronds

en fonction de M et que les performances tendent vers celles correspondant au cas sans beat noise lorsque M tend vers l'infini.

Enfin, l'écart constaté entre les deux modèles est d'autant plus faible que le nombre d'utilisateurs est élevé. Ceci est dû à l'approximation réalisée sur la statistique de l'IAM, qui est localement considérée sur chaque chip comme de nature gaussienne. Cette approximation est d'autant plus vraie que l'interférence devient importante, c'est-à-dire lorsque le nombre d'utilisateurs augmente pour une même famille.

Ce genre de courbe permet aussi de déterminer à partir de quelle valeur de M il est possible de négliger l'impact du beat noise. Cela permet donc de définir, pour une bande passante électrique B_e fixée, la bande passante optique requise, donc le temps de cohérence de la source. Par ailleurs, si ces contraintes ne peuvent pas être satisfaites, cela permet de

3.3 Limitations

prévoir le niveau de dégradation introduit en fonction des caractéristiques du système.

L'étude réalisée sur le beat noise par ces deux approches a permis de quantifier l'impact de ce phénomène sur les performances du système, quelle que soit la configuration (famille de code, nombre d'utilisateurs, valeur des bandes passantes optique et électrique). De plus, à partir du modèle en mixture de χ^2 , la statistique de la variable décisionnelle est décrite de manière détaillée, ce qui est intéressant en vue de l'utilisation de codes correcteurs d'erreurs. En effet, comme nous l'avons vu au chapitre 2, la connaissance de la statistique au niveau du décodeur est importante, afin d'obtenir les performances optimales des codes correcteurs d'erreurs.

3.3.3 Les autres limitations

L'IAM et le beat noise constituent deux limitations inhérentes dues à la technique d'accès multiple et à la loi quadratique de la photodétection. Cependant, même si on peut considérer que la fibre optique se comporte de manière idéale, dans la mesure où les distances du réseau d'accès sont faibles, les composants électroniques du circuit de décision peuvent aussi être une source de bruit. Tout comme dans le cas de la transmission optique vu au cours du chapitre 2, le modèle AWGN permet de traduire ces différents effets. Dans le cadre de l'OCDMA, on peut considérer qu'une variable de bruit ayant une distribution gaussienne centrée, s'ajoute sur chaque temps chip du signal transmis. En réception, suivant qu'on tienne compte ou non de la photodétection, cela conduit à deux modèles possibles, détaillés dans les paragraphes suivants.

3.3.3.1 Photodétection idéale

Si on ne prend pas en compte la détection quadratique, la variable décisionnelle est composée de la donnée désirée, de l'IAM et d'un terme de bruit additif gaussien, de

moyenne nulle n . Avec une réception CCR, elle s'écrit alors :

$$Z_d = b_d \cdot W + I_d + n \quad (3.69)$$

A partir de (3.69) et de la définition générale de la probabilité d'erreur du système (3.25), il est possible d'obtenir [89] la probabilité d'erreur en fonction de la variance σ^2 de la variable n , c'est-à-dire du bruit sur le temps bit :

$$P_{e,CCR} = \frac{1}{4} \sum_{i \geq 0}^{U-1} \binom{U-1}{i} p^i (1-p)^{U-1-i} \cdot \left(\operatorname{erfc} \left(\frac{S-i}{\sigma\sqrt{2}} \right) + \operatorname{erfc} \left(\frac{W+i-S}{\sigma\sqrt{2}} \right) \right) \quad (3.70)$$

Notons que dans ce cas, le S/B, défini comme le rapport E_b/N_0 est donné par :

$$S/B \text{ (dB)} = 10 \cdot \log \left(\frac{W^2}{4\sigma^2} \right) \quad (3.71)$$

En effet, on considère une normalisation en temps chip c'est-à-dire que, quel que soit le poids de la famille de code, la puissance émise par chip est identique, mais la puissance par bit varie et vaut en moyenne, avec une émission équiprobable : $E_b = W/2$. De plus, la puissance du bruit N_0 est définie par $N_0 = 2\sigma_c^2$, où σ_c^2 est la variance du bruit gaussien s'ajoutant sur chaque temps chip. Cette variance est reliée à la variance du bruit sur le temps bit σ^2 par : $\sigma^2 = W \cdot \sigma_c^2$, d'où l'obtention de (3.71).

3.3.3.2 Battements dus à la photodétection

Dans le second cas, on prend en compte les battements dus à la photodétection en plus de la présence de bruit AWGN avant la détection. La variable décisionnelle suit la même statistique que dans le cas sans bruit gaussien (c'est-à-dire une loi de répartition définie

3.3 Limitations

par une mixture de χ^2), mais les paramètres la constituant sont différents. En effet, dans ce cas, avec les mêmes notations r_i et z_i , désignant respectivement l'amplitude du chip i à l'émission et son amplitude après photodétection, on utilise le modèle suivant, traduit par la fonction g :

$$g : r_i \mapsto f(r_i) = z_i = \sum_{j=1}^n x_{ij}^2 \quad (3.72)$$

Avec n l'ordre de la distribution toujours donné par $n = 2M = 2B_o/B_e$ mais une distinction au niveau de la variance var des x_{ij} en fonction de la valeur de r_i comme suit :

$$\begin{aligned} r_i = 0 \text{ ou } 1 & : var = \sigma_b^2 \\ r_i > 1 & : var = \frac{\sigma_I^2}{n} + \sigma_b^2 \end{aligned} \quad (3.73)$$

où σ_b^2 caractérise la variance du bruit gaussien sur chacune des n composantes spectrales. Afin de relier cette variance à la variance σ^2 définie dans le cas AWGN précédemment présenté, on associe σ à la valeur moyenne des écarts types résultants des '1' et '0' émis et suivant une distribution χ^2 . En utilisant la définition de ces écarts types (2.33) $\sigma_i^2 = 2n\sigma^4 + 4\sigma^2 s_i^2$, on obtient :

$$\sigma = \frac{\sigma_1 + \sigma_0}{2} = \sigma_b^2 \sqrt{\frac{n}{2}} \left(\sqrt{1 + \frac{2}{n\sigma_b^2}} + 1 \right) \quad (3.74)$$

Avec ce modèle, la variable décisionnelle obtenue comme la somme des z_i correspondant au code de l'utilisateur désiré est à nouveau une somme de somme de variables x_{ij}^2 donc a une statistique décrite par une mixture de fonctions χ^2 . Cependant, comme dans ce cas les variances des variables de base x_{ij} ne sont pas identiques, il n'est pas évident d'obtenir une formulation de la statistique ni de la probabilité d'erreur du système. Les performances ne peuvent alors être obtenues que par simulation.

Un exemple de performances obtenues avec la famille OOC($F=534$, $W=3$) et $U = 20$ utilisateurs actifs est reporté sur la figure 3.15 pour deux valeurs de M différentes, et sont comparées au cas où on ne considère que l'impact du bruit AWGN. Dans chacun des cas, le résultat est reporté pour le seuil de détection optimal minimisant le TEB.

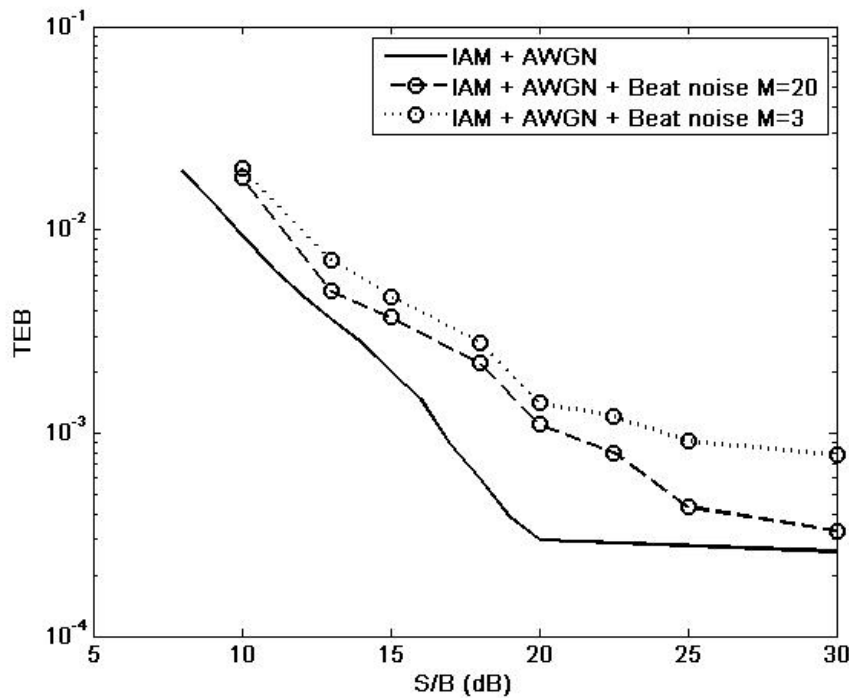


FIG. 3.15 – Performances obtenues par simulation avec la famille OOC($F=534$, $W=3$) et $U = 20$ utilisateurs, en présence de bruit AWGN et de beat noise dû à la photodétection

De la même manière que dans le cas avec beat noise et sans bruit AWGN, les performances sont d'autant plus dégradées que M est faible. De plus, cette courbe permet de mettre en évidence que le S/B requis pour atteindre un TEB dépend de la valeur de M et est d'autant plus important que M est faible. En effet, sur l'exemple présenté, un S/B de 21 dB est nécessaire pour satisfaire un TEB de 10^{-3} avec $M = 20$ tandis qu'il faut pour ce même TEB un S/B de 25 dB pour $M = 3$. Ainsi, si on souhaite caractériser l'effet

3.3 Limitations

d'un bruit gaussien sur les performances du système OCDMA, il est important de prendre en compte les battements (c'est-à-dire le beat noise) engendrés par la photodétection.

3.3.4 Conclusions

En conclusion, les performances du système OCDMA sont fortement dégradées par les différentes limitations présentées dans les paragraphes précédents : IAM, beat noise et bruit. L'ensemble de ce travail sur l'étude des battements dans le système OCDMA a fait l'objet de différentes publications, traitant du premier modèle de beat noise basé sur la décomposition du temps chip [83], du deuxième modèle de beat noise basé sur une mixture de χ^2 [84], et de la prise en compte du bruit avant la détection dans le modèle en mixture de χ^2 [85]. L'originalité de ces travaux a conduit à l'obtention d'un "best paper" à la conférence AICT 2009 [84].

Nous avons montré à travers cette étude qu'à condition de choisir une longueur temporelle de code très grande et un rapport de bande passante $M = B_o/B_e$ élevé, il est possible de limiter cette dégradation de performances. Cependant, cela signifie que les débits correspondants sont très faibles et que les contraintes sur le choix des sources sont importantes. Ainsi, il est intéressant d'envisager d'autres solutions, comme le codage correcteur d'erreurs pour compenser la dégradation de performances engendrée par les différentes limitations.

La partie suivante est consacrée à l'application au système OCDMA des codes correcteurs d'erreurs étudiés précédemment, en particulier les codes LDPC. En effet, l'étude de la statistique de la variable décisionnelle permet d'envisager l'adaptation du décodeur LDPC et donc l'optimisation des performances.

3.4 Amélioration des performances avec codes correcteurs d'erreurs

3.4.1 Contexte

3.4.1.1 Présentation

Afin de compenser les erreurs introduites par les différentes perturbations présentes sur la chaîne de transmission, une solution utilisant les codes correcteurs LDPC a été étudiée. Dans les différents cas envisagés (IAM avec ou non bruit gaussien et avec ou non beat noise), la connaissance de la statistique de la variable décisionnelle est utilisée pour définir les paramètres nécessaires au décodage des données reçues. L'étude des performances des codes LDPC sur un canal OCDMA est complétée par une comparaison avec l'utilisation de codes RS de référence.

Dans chacun des cas traités, des simulations sont réalisées afin de déterminer le TEB de la transmission avec codage correcteur d'erreurs. Le schéma de la chaîne de transmission est décrit dans le paragraphe suivant. L'étude des performances des codes LDPC est ensuite présentée dans différents cas de figures.

3.4.1.2 Chaîne de transmission non codée

La chaîne d'émission / réception OCDMA, dont les principaux éléments ont été décrits dans les paragraphes précédents, a été développée en C de manière à obtenir par simulation, les performances du système dans plusieurs cas de figures. Le principe est représenté figure 3.16.

Les différentes étapes de la simulation sont :

- Choix de U séquences de codes aléatoires c_j , $j \in [1, U]$ parmi les N_{OOC} séquences

3.4 Amélioration des performances avec codes correcteurs d'erreurs

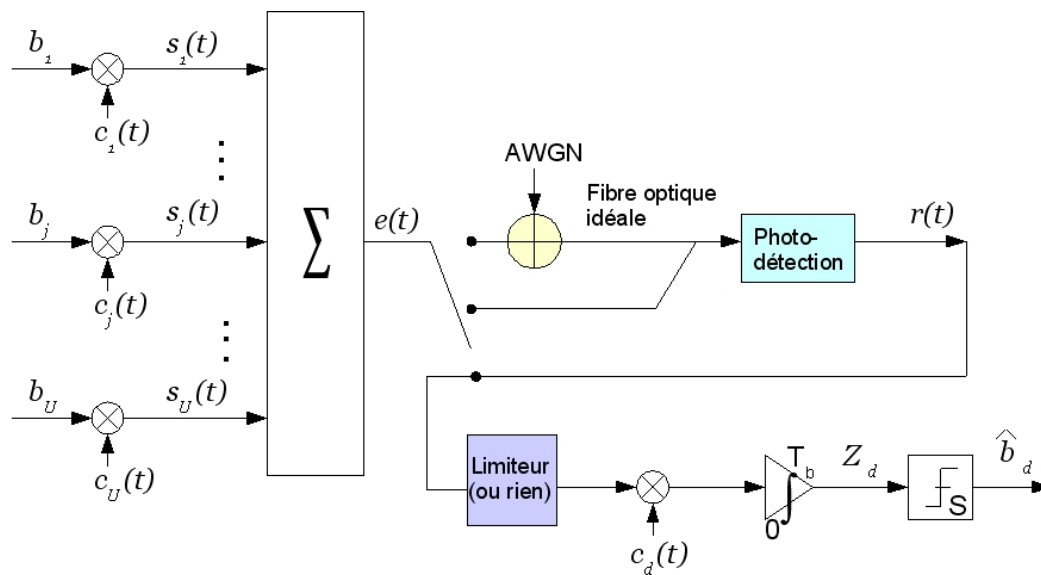


FIG. 3.16 – Chaîne de transmission complète OCDMA

possibles (N_{OOC} est donné par (3.4)), et d'un décalage aléatoire l_j pour chaque utilisateur

- Emission équiprobable de données b_j pour U utilisateurs différents
- Multiplication de chaque donnée par le code de l'utilisateur correspondant
- Sommation des données codées de tous les utilisateurs
- Ajout ou non de bruit de nature AWGN sur chaque chip (cas bruité ou idéal)
- Photodétection pour un des utilisateurs désirés, idéale ou avec prise en compte des battements
- Fonction limiteur ou non
- Réception par corrélation (CCR) pour l'utilisateur désiré
- Prise de décision par comparaison au seuil S
- Comparaison de la donnée décidée avec la donnée émise par l'utilisateur désiré
- Comptabilisation des erreurs puis du TEB

3.4.1.3 Chaîne de transmission codée

Dans le cas avec codage correcteur d'erreurs, des blocs d'encodage et de décodage sont ajoutés. De plus, les données traitées ne sont plus des simples bits mais des vecteurs, composés de N ou K bits (données codées ou non). Dans ce cas, la chaîne représentée figure 3.17 est composée de :

- Emission pour chaque utilisateur, de K bits aléatoires $u_j^{(i)}$ formant les vecteurs U_j
- Codage des informations par multiplication avec la matrice génératrice ou le polynôme générateur du code pour former les vecteurs C_j de taille N
- Multiplication de chaque donnée du vecteur C_j par le code de l'utilisateur correspondant
- Somme, ajout ou non de bruit, détection, réception pour l'utilisateur désiré
- Prise de décision par algorithme de correction d'erreurs
- Récupération des K bits d'information de l'utilisateur désiré
- Comparaison des bits de données décidés avec ceux émis
- Comptabilisation des erreurs puis du TEB

Cette chaîne, également codée en C, permet de déterminer les performances du système OCDMA codé pour différents cas de configurations.

3.4.2 Application des codes LDPC à la chaîne de transmission OCDMA

3.4.2.1 Présentation

La donnée à l'entrée du décodeur LDPC pour l'utilisateur désiré d est directement le mot reçu V_d composé des variables décisionnelles $Z_d^{(i)}$ pour i allant de 0 à $N - 1$. En effet, comme on effectue un décodage souple, afin d'obtenir les meilleures performances

3.4 Amélioration des performances avec codes correcteurs d'erreurs

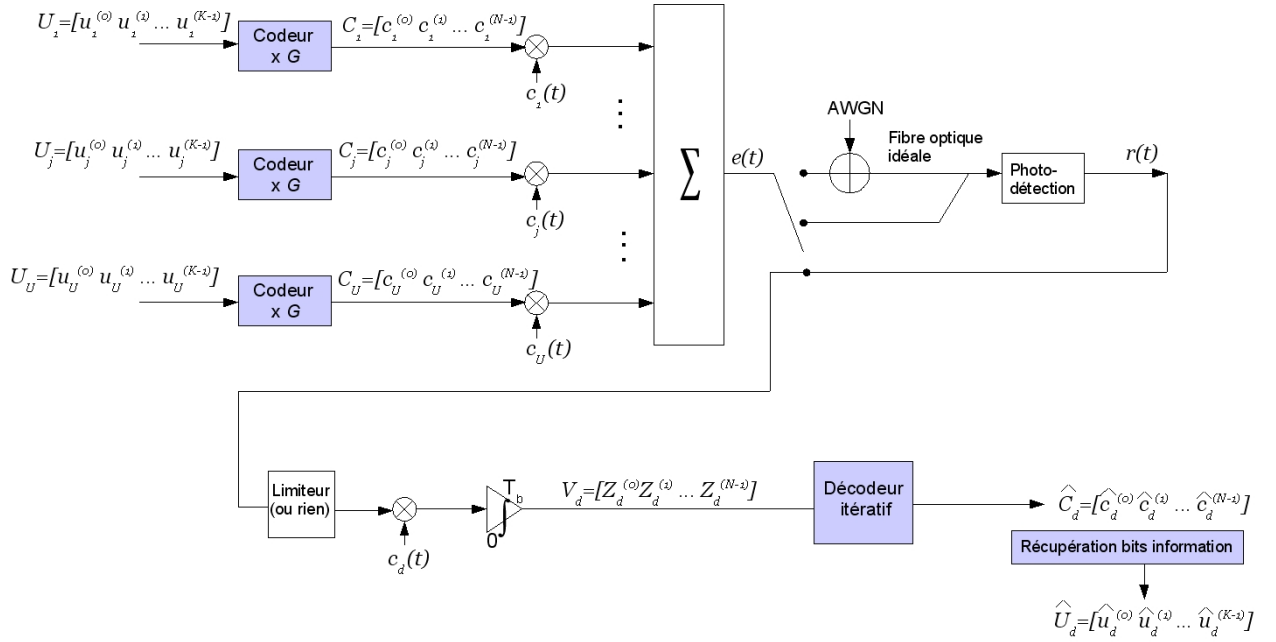


FIG. 3.17 – Chaîne de transmission complète OCDMA avec codage correcteur d'erreurs

possibles, il faut que les valeurs d'entrées contiennent le maximum d'informations donc ne soient pas seuillées. A partir de ces valeurs d'entrées, l'étape essentielle lors de l'initialisation du décodeur consiste à déterminer les LLR pour i allant de 0 à $N - 1$, qui ont, avec les notations employées dans ce contexte, la définition suivante :

$$LLR = \ln \left(\frac{p \left(u_d^{(i)} = 0 \mid Z_d^{(i)} \right)}{p \left(u_d^{(i)} = 1 \mid Z_d^{(i)} \right)} \right) \quad (3.75)$$

Dans les différents cas envisagés (cas idéal avec de l'IAM uniquement, cas avec en plus du bruit gaussien, cas avec en plus du beat noise) et afin de déterminer le rapport (3.75), les caractéristiques de la variable décisionnelle sont utilisées. Une fois l'expression des LLR établie, les performances sont obtenues par simulation, avec l'initialisation particulière des LLR suivi de l'algorithme de décodage classique.

3.4.2.2 Evaluation des performances en présence d'IAM (cas idéal)

Lorsqu'on ne considère que l'impact de l'IAM sur les performances, le terme d'interférence suit une loi de distribution binomiale de moyenne m_I (3.23) et de variance σ_I^2 (3.24). La variable décisionnelle ne prend donc que des valeurs entières positives et le modèle simple gaussien est de ce fait inapproprié. Afin de simplifier le décodage dans ce cas, on fait cependant l'approximation suivante :

- pour toute valeur positive ou nulle, la densité de probabilité de I_d est approchée par une portion de loi gaussienne ayant pour paramètres la variance σ_I^2 (3.24) et la moyenne m_I (3.23) issues de la loi binomiale,
- pour toute valeur négative, la densité de probabilité est nulle.

La fonction $f_{I_d}(x)$, représentant la densité de probabilité de la variable d'IAM I_d , s'écrit alors avec le modèle approché :

$$f_{I_d}(x) = \begin{cases} \frac{\alpha}{\sqrt{2\pi}\sigma_I} e^{-\frac{(x-m_I)^2}{2\sigma_I^2}} & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (3.76)$$

Avec α un coefficient permettant de normaliser la fonction de densité de probabilité à 1, c'est-à-dire tel que :

$$\int_{-\infty}^{+\infty} f_{I_d}(t) dt = 1 \quad (3.77)$$

En résolvant (3.77), il vient que $\alpha = 1/\phi(m/\sigma_I)$ où $\phi(x)$ est définie pour tout $x \in \mathbb{R}$ par $\phi(x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$.

Cette densité de probabilité décrite par $f_{I_d}(x)$ permet d'avoir un modèle simple tel que les valeurs $p(I_d = i)$ pour i entre 0 et $U - 1$ données par (3.20) soient proches des valeurs $f_{I_d}(x = i)$ données par (3.76).

3.4 Amélioration des performances avec codes correcteurs d'erreurs

La figure 3.18 représente la densité de probabilité de la variable I_d , avec la loi binomiale, et l'approximation décrite par f_{I_d} , pour la famille MWOOC($67 \times 8, 7$) et différents nombres d'utilisateurs U . On peut remarquer que le modèle approché donne une valeur de $f_{I_d}(x = i)$

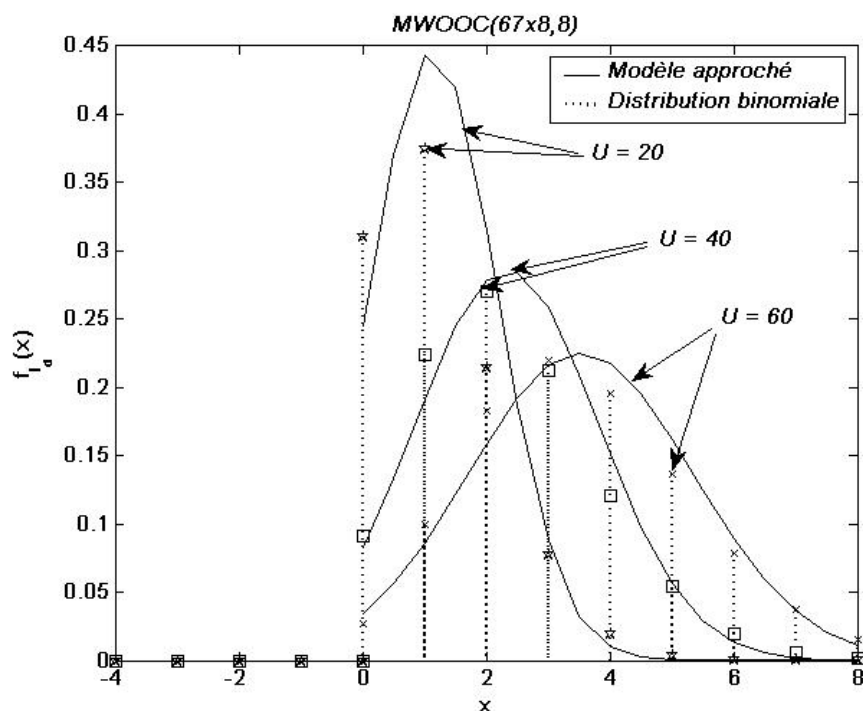


FIG. 3.18 – Comparaison des densités de probabilité entre le modèle approché et la loi de répartition binomiale pour la famille MWOOC($67 \times 8, 7$) et différents nombres d'utilisateurs U

d'autant plus proche de $p(I_d = i)$ que le nombre d'utilisateurs U est élevé. De plus, même si dans certains cas (pour $U = 20$ par exemple), l'approximation n'est pas toujours fidèle pour de faibles valeurs de I_d , on peut remarquer que la différence entre les deux modèles (approximation et binomial) est moins importante lorsque le terme pris par I_d augmente. Ceci signifie que le modèle présenté constitue une bonne approximation dès que le terme d'interférence est significatif, c'est-à-dire lorsque l'IAM engendre des erreurs.

A l'entrée du décodeur, afin de se ramener au cas d'une transmission normalisée à l'émission de '1' et de '0', la variable décisionnelle reçue est divisée par le poids du code W de manière à former la variable Y telle que :

$$Y = \frac{Z_d^{(i)}}{W} = u_d^{(i)} + \frac{I_d^{(i)}}{W} \quad (3.78)$$

$u_d^{(i)}$ vaut 0 ou 1. La variable d'IAM jouant ici le rôle du bruit, est divisée par W . Ainsi, la variance et la moyenne de la variable d'IAM normalisée $I' = \frac{I_d^{(i)}}{W}$ sont données par :

$$\sigma_{I'}^2 = \frac{\sigma_I^2}{W^2} \quad (3.79)$$

$$m_{I'} = \frac{m_I}{W} \quad (3.80)$$

La densité de probabilité de la variable d'IAM normalisée à partir de laquelle les LLR vont être déterminés, prend alors l'expression suivante, avec $A = \frac{1}{\phi(m_{I'}/\sigma_{I'})}$:

$$f_{I'}(x) = \begin{cases} \frac{A}{\sqrt{2\pi}\sigma_{I'}} e^{-\frac{(x-m_{I'})^2}{2\sigma_{I'}^2}} & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (3.81)$$

Sachant que la variable d'IAM est strictement positive, si la variable normalisée Y est inférieure à 1, la donnée émise $u_d^{(i)}$ ne peut pas être un 1. Ceci se traduit par : $p(u_d^{(i)} = 1 | Z_d^{(i)}) = 0$ pour $Y < 1$, et le LLR est dans ce cas égal à $+\infty$. Dans les simulations, nous fixerons la valeur du LLR à 100. Pour tous les autres cas, on considère que l'IAM normalisée I' a une répartition décrite par (3.81) et dans ce cas, le LLR s'exprime comme dans le cas classique d'un bruit gaussien (2.61), mais avec une moyenne $m_{I'}$ c'est-à-dire :

$$LLR = \frac{2(Y - m_{I'}) - 1}{2\sigma_{I'}^2} \quad (3.82)$$

3.4 Amélioration des performances avec codes correcteurs d'erreurs

Ainsi pour résumer, les différentes étapes du décodage sont :

- Normalisation de la variable décisionnelle pour former Y selon (3.78)
- Comparaison au seuil $Y_L = 1$ pour calculer les LLR selon :

$$LLR = \begin{cases} \frac{2(Y-m_{I'})-1}{2\sigma_{I'}^2} & \text{pour } Y \geq Y_L = 1 \\ 100 & \text{pour } Y < Y_L = 1 \end{cases} \quad (3.83)$$

Des simulations ont été réalisées dans différents cas (pour différentes familles de codes OOC ou MWOOC, pour différents codes correcteurs d'erreurs) et ont révélé l'efficacité des codes correcteurs dans ce contexte. Parmi les différents résultats de simulations, les TEB obtenus avec la famille MWOOC($67 \times 8, 8$) et le code LDPC(905,725) sont représentés sur la figure 3.19.

On peut remarquer que les TEB obtenus avec le code LDPC(905,725) ayant une redondance de 24%, sont nettement inférieurs à ceux obtenus dans le cas non codé. De plus, la correction est significative même pour un nombre d'utilisateurs élevé. En effet, pour $U = 75$, le TEB initialement à 4×10^{-2} est réduit d'environ deux décades (4×10^{-4}). Pour un nombre d'utilisateurs plus faible, les performances sont encore meilleures dans la mesure où le TEB non codé est plus faible. En effet, le code correcteur est par conséquent plus efficace. Par exemple, pour $U = 60$ utilisateurs, le gain est proche de 3 décades puisque le TEB est réduit de 10^{-2} à 2×10^{-5} .

Ceci permet de valider le modèle statistique utilisé pour décrire l'IAM et calculer les LLR au niveau du décodeur.

3.4.2.3 Evaluation des performances en présence d'IAM et de bruit gaussien

Dans ce paragraphe, on considère qu'une variable de bruit gaussien de moyenne nulle s'ajoute sur chaque temps chip du signal transmis. Dans ce cas, des erreurs pourront être

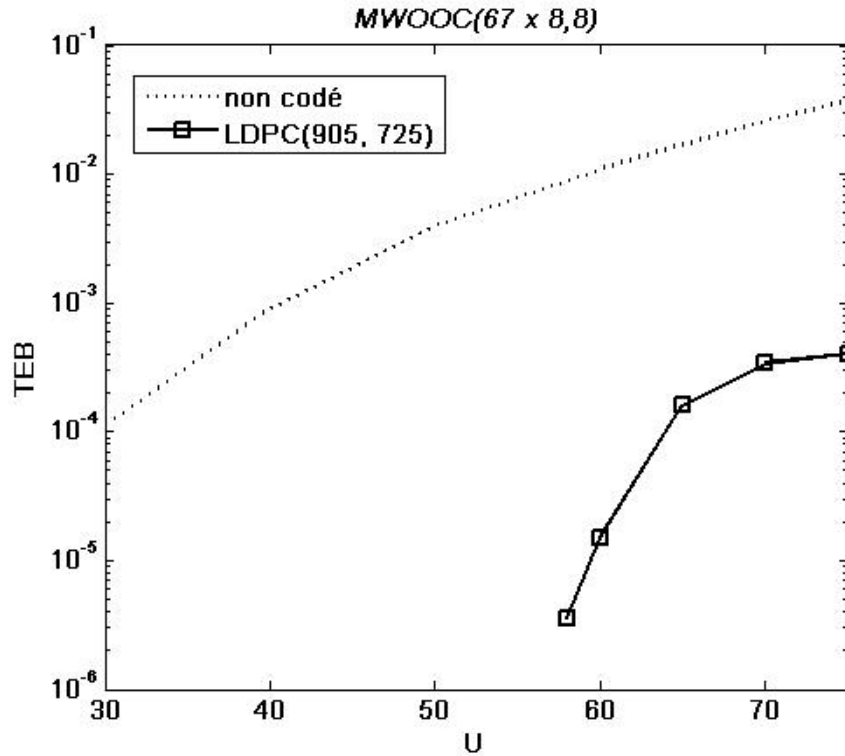


FIG. 3.19 – TEB en fonction du nombre d'utilisateurs U pour la famille MWOOC($67 \times 8, 8$) dans le cas non codé et avec le code LDPC(905,725)

faites sur les '0' et sur les '1', contrairement au cas précédent. La variable de bruit notée n sur le temps bit possède une variance σ^2 et sa densité de probabilité s'exprime pour toute valeur de x par :

$$f_n(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (3.84)$$

Comme dans le cas précédent, on considère la variable d'IAM normalisée par rapport à 1, c'est-à-dire : $I' = \frac{I_d^{(i)}}{W}$.

En considérant les variables d'IAM et de bruit gaussien indépendantes, on peut déterminer la densité de probabilité de la variable de bruit globale (IAM+ bruit gaussien) comme le produit de convolution des expressions (3.84) et (3.81). Ce calcul conduit à l'expression

3.4 Amélioration des performances avec codes correcteurs d'erreurs

suivante (voir annexe G), avec σ_T défini tel que $\sigma_T^2 = \sigma_{I'}^2 + \sigma^2$:

$$f_T(x) = \frac{A}{2\sqrt{2\pi}\sigma_T} e^{-\frac{(x-m_{I'})^2}{2\sigma_T^2}} \operatorname{erfc}\left(-\frac{\sigma_{I'}\sigma}{\sqrt{2}\sigma_T} \left(\frac{x}{\sigma^2} + \frac{m_{I'}}{\sigma_{I'}^2}\right)\right) \quad (3.85)$$

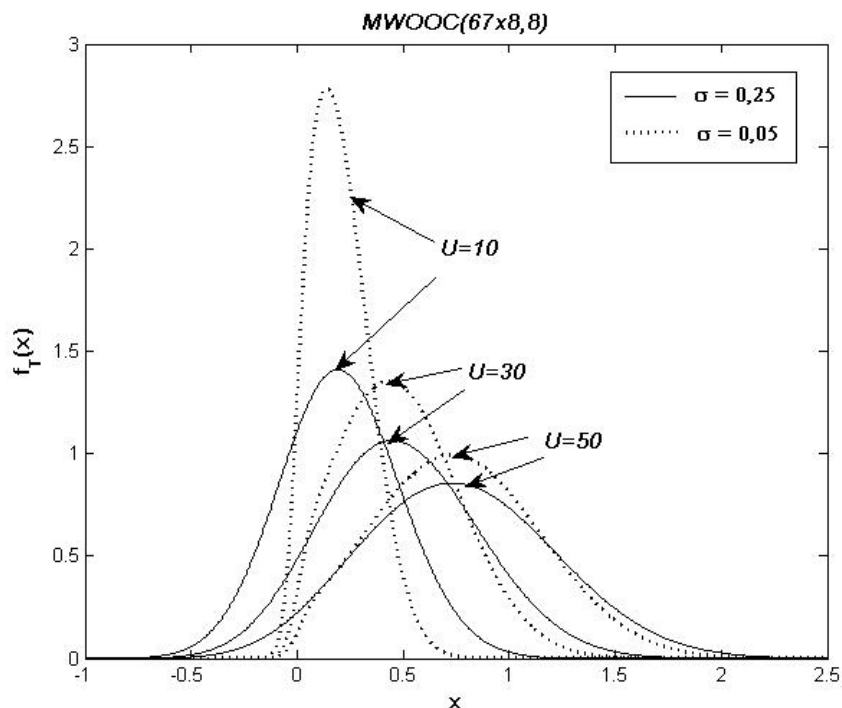


FIG. 3.20 – Densité de probabilité globale $f_T(x)$ pour la famille MWOOC($67 \times 8, 8$) avec différents nombres d'utilisateurs actifs U et deux valeurs de σ différentes

La figure 3.20 représente cette fonction dans le cas d'un code MWOOC($67 \times 8, 8$) pour différents nombres d'utilisateurs et pour deux valeurs différentes de σ : $\sigma = 0,05$ et $\sigma = 0,25$. On peut remarquer que dans les cas où l'IAM est prépondérante par rapport au bruit (cas $\sigma = 0,05$), les densités de probabilités obtenues sont très asymétriques, contrairement au cas où $\sigma = 0,25$ où l'asymétrie est moins prononcée. En effet, lorsque $\sigma = 0,25$, le bruit gaussien est prépondérant par rapport à l'IAM et dans ce cas, la densité de probabilité donnée par (3.85) tend vers la densité de probabilité d'une loi gaussienne

de variance σ^2 , mais centrée en $m_{I'}$ suite à la convolution. A l'inverse, dans l'autre cas, la densité de probabilité ne possède pas la même largeur sur la partie droite ou gauche de la cloche. Ceci est lié à l'opération de convolution à l'issue de laquelle la partie gauche de la courbe représente la contribution due au bruit gaussien et la partie droite celle du bruit global de variance σ_T^2 . Lorsque le bruit gaussien est prépondérant, σ_T^2 tend vers σ^2 et la courbe se symétrise. Lorsque l'IAM est prépondérante, σ_T^2 tend vers $\sigma_{I'}^2$ et la courbe devient asymétrique, proche du cas idéal (IAM sans bruit).

A partir de ces considérations, on définit une valeur limite Y_L représentant un seuil tel que : pour $Y < Y_L$, les erreurs faites sur les '1' sont dues au bruit gaussien et pour $Y > Y_L$, les erreurs faites sur les '0' sont dues au bruit global (IAM + bruit gaussien).

En considérant des modèles basés sur des portions de gaussiennes, cette valeur peut se calculer simplement et correspond alors à la valeur de Y telle que :

$$p(u_d^{(i)} = 1|Y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(Y-1)^2}{2\sigma^2}} = p(u_d^{(i)} = 0|Y) = \frac{1}{\sqrt{2\pi}\sigma_T} e^{-\frac{(Y)^2}{2\sigma_T^2}} \quad (3.86)$$

La résolution de cette équation est détaillée en annexe H et fournit une expression générale du seuil sous la forme :

$$Y_L = 1 + \frac{\sigma^2}{\sigma_{I'}^2} - \frac{\sigma\sigma_T}{\sigma_{I'}^2} \sqrt{1 - 2\sigma_{I'}^2 \ln\left(\frac{\sigma}{\sigma_T}\right)} \quad (3.87)$$

Ainsi, les LLR peuvent être déterminés en fonction de leur valeur par rapport à Y_L selon :

$$LLR = \begin{cases} \frac{2(Y-m_{I'})-1}{2\sigma_T^2} & \text{pour } Y \geq Y_L \\ \frac{2(Y-m_{I'})-1}{2\sigma^2} & \text{pour } Y < Y_L \end{cases} \quad (3.88)$$

Le reste de la simulation reste inchangé par rapport au cas précédent. Les résultats sont alors présentés en fonction du rapport signal à bruit S/B tenant compte à la fois de

3.4 Amélioration des performances avec codes correcteurs d'erreurs

la normalisation en temps chip et du fait que dans le cas codé, la puissance utile ne se trouve que sur les K bits des N bits transmis, c'est-à-dire :

$$S/B = 10 \log \left(\frac{WK}{4N\sigma^2} \right) \quad (3.89)$$

Les courbes sur la figure 3.21 représentent les TEB obtenus avec la famille MWOOC($67 \times 8, 8$) pour $U = 60$ utilisateurs et avec le code LDPC($905, 725$), en fonction du rapport signal à bruit.

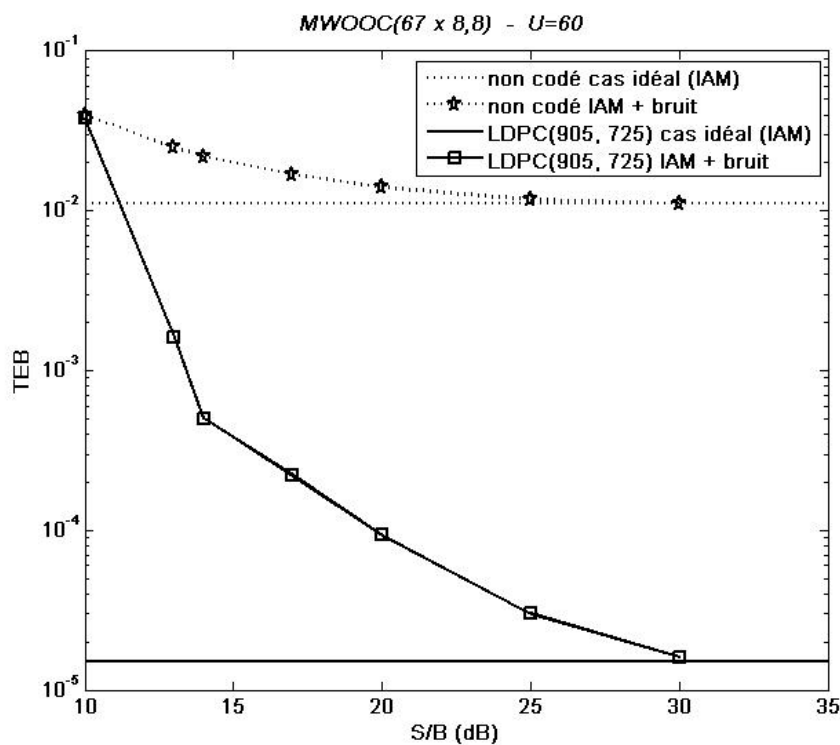


FIG. 3.21 – TEB obtenus pour la famille MWOOC($67 \times 8, 8$) avec $U = 60$ utilisateurs dans le cas non codé et avec le code LDPC($905, 725$) en fonction du rapport signal à bruit

On remarque tout d'abord, dans le cas codé comme dans le cas non codé, que lorsque le S/B augmente, le TEB tend vers le TEB correspondant dans le cas idéal (codé ou non).

Ceci permet de valider le modèle aux limites c'est-à-dire lorsque l'IAM est prépondérante par rapport au bruit gaussien. De plus, on remarque que pour toute valeur du rapport signal à bruit supérieure à 10 dB, le code LDPC apporte une correction, qui est d'autant plus efficace que le S/B est élevé. En effet, comme nous l'avons déjà vu, le pouvoir de correction du code augmente lorsque le TEB est plus faible, donc ici lorsque le S/B augmente.

De cet exemple, on peut extraire que le TEB non codé dans le cas idéal de 10^{-2} est atteint avec le code LDPC(905,725) pour un S/B de 11 dB tandis que dans le cas non codé, un S/B de plus de 25 dB était nécessaire. On a ainsi un gain important en S/B qui est alors de 14 dB.

3.4.2.4 Evaluation des performances en présence d'IAM et de beat noise

Dans le cas où on considère l'impact du beat noise, la densité de probabilité de la variable décisionnelle a été entièrement décrite par le modèle statistique en mixture de χ^2 , qu'il s'agisse d'un '1' ou d'un '0' émis, respectivement par $p^{(1)}(Z_d)$ (3.60) et $p^{(0)}(Z_d)$ (3.61).

Avec ces expressions, il est possible de déterminer les probabilités recherchées dans le calcul des LLR (3.75). En effet, on a $p(u_d^{(i)} = 1 | Z_d^{(i)}) = p^{(1)}(Z_d^{(i)})$, directement donnée par (3.60), et $p(u_d^{(i)} = 0 | Z_d^{(i)}) = p^{(0)}(Z_d^{(i)})$ directement donnée par (3.61).

A partir de l'expression (3.60), on voit que la densité de probabilité $p^{(1)}(Z_d^{(i)})$ est nulle pour $Z_d^{(i)} = 0$, qu'elle a une valeur particulière pour $Z_d^{(i)} = W$ et qu'elle est définie pour toute autre valeur positive par une seule fonction.

Ainsi, on distingue trois cas qui sont :

3.4 Amélioration des performances avec codes correcteurs d'erreurs

– $Z_d^{(i)} = 0$; dans ce cas :

$$p\left(u_d^{(i)} = 1 \mid Z_d^{(i)}\right) = p^{(1)}(0) = 0 \quad (3.90)$$

– $Z_d^{(i)} = W$, on a alors :

$$p\left(u_d^{(i)} = 1 \mid Z_d^{(i)}\right) = p^{(1)}(W) = p(I_d = 0) + \sum_{i=1}^{U-1} p(I_d = i) \cdot p_{i+W, W' \neq 0}(Z_d^{(i)}) \quad (3.91)$$

– $Z_d^{(i)} > 0$ et $Z_d^{(i)} \neq W$: c'est le cas le plus général où seul le terme de somme pondérée de fonctions χ^2 est non nul, d'où :

$$p\left(u_d^{(i)} = 1 \mid Z_d^{(i)}\right) = p^{(1)}(Z_d^{(i)}) = \sum_{i=1}^{U-1} p(I_d = i) \cdot p_{i+W, W' \neq 0}(Z_d^{(i)}) \quad (3.92)$$

En ce qui concerne $p\left(u_d^{(i)} = 0 \mid Z_d^{(i)}\right) = p^{(0)}\left(Z_d^{(i)}\right)$, de par la présence des fonctions de dirac δ , il existe autant de cas particuliers que de valeurs entières prises par $Z_d^{(i)}$ entre 0 et W .

Ainsi, on distingue les différents cas suivants :

– $Z_d^{(i)} = 0$; dans ce cas :

$$p\left(u_d^{(i)} = 0 \mid Z_d^{(i)}\right) = p^{(0)}(0) = p(I_d = 0) \quad (3.93)$$

– $Z_d^{(i)}$ est un entier, $1 \leq Z_d^{(i)} \leq W$: on a alors le terme multiplicatif au dirac

correspondant à la valeur de $Z_d^{(i)}$ plus les sommes pondérées c'est-à-dire :

$$\begin{aligned}
 p\left(u_d^{(i)} = 0 \mid Z_d^{(i)}\right) &= p^{(0)}(Z_d^{(i)}) = p(I_d = Z_d^{(i)}) \cdot \frac{\binom{W}{Z}}{\binom{W+Z-1}{Z}} \\
 &+ \sum_{i=2}^W p(I_d = i) \cdot \left(1 - \frac{\binom{W}{i}}{\binom{W+i-1}{i}}\right) \cdot p_{i,W' \neq 0}(Z_d^{(i)}) \\
 &+ \sum_{i=W+1}^{U-1} p(I_d = i) \cdot p_{i,W' \neq 0}(Z_d^{(i)})
 \end{aligned} \tag{3.94}$$

– $Z_d^{(i)}$ prend n'importe quelle autre valeur, c'est-à-dire $Z_d^{(i)}$ est positif mais n'est pas un entier ou $Z_d^{(i)}$ est un entier strictement supérieur à W : dans ce cas, seuls les termes de sommes pondérées de fonctions χ^2 sont non nuls, d'où :

$$\begin{aligned}
 p\left(u_d^{(i)} = 0 \mid Z_d^{(i)}\right) &= p^{(0)}(Z_d^{(i)}) = \sum_{i=2}^W p(I_d = i) \cdot \left(1 - \frac{\binom{W}{i}}{\binom{W+i-1}{i}}\right) \cdot p_{i,W' \neq 0}(Z_d^{(i)}) \\
 &+ \sum_{i=W+1}^{U-1} p(I_d = i) \cdot p_{i,W' \neq 0}(Z_d^{(i)})
 \end{aligned} \tag{3.95}$$

En confrontant les cas (3.90) à (3.92) aux cas (3.93) à (3.95), on trouve alors quatre cas différents pour le calcul du LLR qui, pour rappel, s'écrit d'une manière générale comme

3.4 Amélioration des performances avec codes correcteurs d'erreurs

(3.75) :

$$LLR = \ln \left(\frac{p(u_d^{(i)} = 0 | Z_d^{(i)})}{p(u_d^{(i)} = 1 | Z_d^{(i)})} \right)$$

Le calcul des LLR s'effectue donc selon les quatre cas suivants :

- $Z_d^{(i)} = 0$: sachant que d'après (3.93) : $p(b_d^i = 1 | Z_d^i) = p^{(1)}(0) = 0$, le rapport logarithmique donne le résultat $LLR = +\infty$: il sera donc fixé à 100 dans les simulations
- $Z_d^{(i)}$ est un entier, $1 \leq Z_d^{(i)} < W$: le LLR est obtenu comme le rapport logarithmique de (3.94) et (3.92)
- $Z_d^{(i)} = W$: le LLR est obtenu comme le rapport logarithmique de (3.94) et (3.91)
- pour toute autre valeur de $Z_d^{(i)}$: le LLR est obtenu comme le rapport logarithmique de (3.95) et (3.92)

Pour rappel, dans toutes les expressions ci-dessus, les termes de la forme $p_{i,W' \neq 0}(Z_d)$ sont des fonctions χ^2 non-centrées définies par (3.52) c'est-à-dire :

$$p_{i,W' \neq 0}(Z_d) = \frac{1}{2var} \left(\frac{Z_d}{i} \right)^{(nW'-2)/4} e^{-(i+Z_d)/(2var)} I_{nW'/2-1} \left(\frac{\sqrt{Z_d i}}{var} \right).$$

var représente la variance des variables gaussiennes, dépendant de la variance de l'IAM σ_I^2 selon (3.45) : $var = \frac{\sigma_I^2}{n}$.

Remarque : l'expression $p_{i,W' \neq 0}(Z_d)$ conduit numériquement à une valeur indéterminée pour de grandes valeurs de n . En effet, si var tend vers 0, le terme en exponentielle tend vers 0 tandis que la fonction de Bessel tend vers l'infini. Ainsi, pour implémenter ce calcul en C, il faut faire une approximation sur l'expression de la fonction de Bessel, valable

pour de grandes valeurs d'arguments et définie par [91] :

$$I_N(Nz) = \frac{1}{\sqrt{2\pi N}} \frac{e^{Nz}}{(1+z^2)^{1/4}} \quad (3.96)$$

Pour de grandes valeurs de n : $nW'/2 - 1 \approx nW'/2$, donc en posant $N = nW'/2$ et $z = \frac{\sqrt{Z_d i}}{\sigma_I^2} \frac{2}{W'}$, on peut alors écrire :

$$I_{nW'/2-1}(\sqrt{Z_d i}/var) \approx I_{nW'/2}(n\sqrt{Z_d i}/\sigma_I^2) = I_N(Nz) \quad (3.97)$$

En remplaçant (3.97) dans l'expression de la fonction χ^2 non centrée (3.52), il vient :

$$p_{i,W' \neq 0}(Z_d) = \frac{1}{2var\sqrt{\pi nW'}} \left(\frac{Z_d}{i}\right)^{\frac{nW'-2}{4}} \frac{1}{\left(1 + \left(\frac{2\sqrt{Z_d i}}{W'\sigma_I^2}\right)^2\right)^{1/4}} e^{\frac{1}{var}\left(\sqrt{Z_d i} - \frac{i+Z_d}{2}\right)} \quad (3.98)$$

Cette formulation (3.98) enlève le problème de la forme indéterminée, et permet l'implémentation du décodeur en C.

Des simulations ont été réalisées en présence de beat noise, utilisant le modèle en mixture de χ^2 , ainsi que le codage correcteur d'erreurs utilisant l'initialisation des *LLR* présentée dans ce paragraphe. Les résultats en termes de TEB sont présentés figure 3.22 pour la famille OOC($F=534$, $W=3$) pour 45 et 50 utilisateurs actifs en fonction de M , avec pour rappel $M = B_o/B_e = n/2$. Les résultats sont représentés dans le cas sans codage correcteur et avec le code LDPC(2715,2535).

On note que le code apporte une nette correction, et que notamment, pour de faibles valeurs de M , il est possible de retrouver les performances obtenues dans le cas idéal mais sans codage. En effet, pour 50 utilisateurs, le TEB correspondant au cas sans beat noise et sans codage, est atteint dans le cas codé dès que $M = 2$. Des résultats similaires sont

3.4 Amélioration des performances avec codes correcteurs d'erreurs

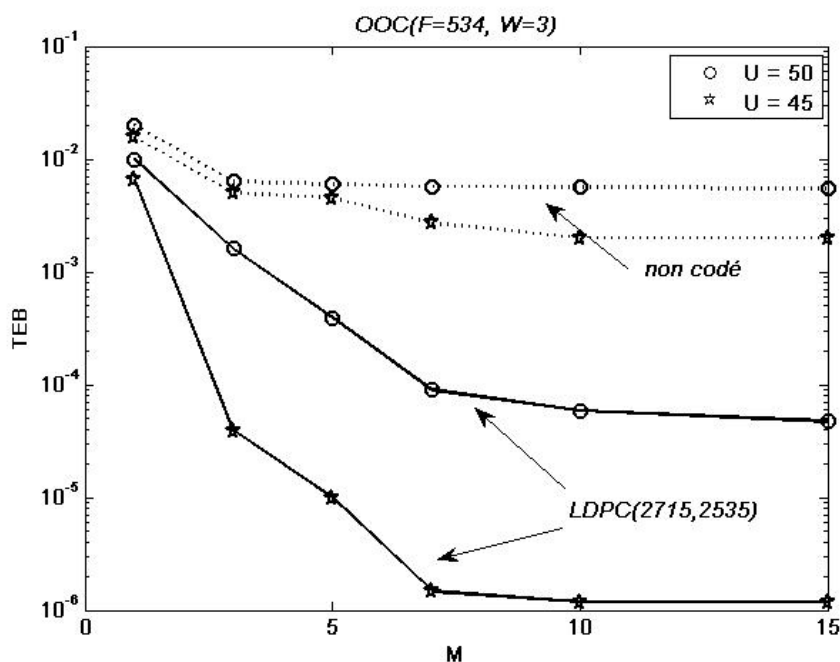


FIG. 3.22 – TEB en fonction du rapport $M = B_o/B_e$ pour la famille $OOC(534, 3)$ dans le cas non codé et avec le code LDPC(2715,2535) pour $U = 45$ et $U = 50$ utilisateurs

obtenus pour 45 utilisateurs. Ceci implique que le codage correcteur d'erreurs peut être utilisé afin de réduire la contrainte imposée sur M et donc sur la cohérence de la source.

Ces résultats prouvent aussi que le codage correcteur d'erreurs permet au réseau de supporter plus d'utilisateurs actifs pour une performance visée et une valeur de M imposée. En effet, dans l'exemple présenté et pour $M = 10$, un TEB de 10^{-4} est atteint avec $U = 50$ utilisateurs dans le cas codé. Cependant, à partir de (3.68) on peut déterminer que le nombre maximal d'utilisateurs permettant de satisfaire ce TEB pour $M = 10$ est limité à 12 utilisateurs. Ainsi, dans cet exemple, le code correcteur a permis de multiplier par 4 le nombre d'utilisateurs actifs.

3.4.2.5 Conclusions

L'étude de codes LDPC a été réalisée dans le cadre d'une transmission optique utilisant la technique d'accès multiple par répartition de codes, dans différents cas, afin de compenser les erreurs introduites par l'IAM, le bruit et le beat noise. Les différents cas traités sont : cas idéal avec IAM uniquement, cas IAM avec bruit gaussien et IAM avec bruit de battements (beat noise). Dans chacun des cas, les performances ont été améliorées par l'utilisation de codes correcteurs LDPC, à condition d'adapter correctement le décodage à la statistique de la variable reçue. Les premiers travaux réalisés avec l'utilisation de codes LDPC sur un canal OCDMA avec IAM et en considérant en plus un bruit gaussien ont fait l'objet d'une publication [92]. Une version améliorée et définissant précisément les modèles utilisés ainsi que les calculs réalisés est en cours de publication [93]. Le travail développant le calcul des LLR dans le cas d'IAM avec beat noise n'a quant à lui jamais été publié, mais est actuellement en cours de rédaction.

Concernant le cas où le signal reçu est perturbé à la fois par le bruit gaussien s'ajoutant avant la détection et par le beat noise lors de la photo-détection, n'ayant pas de description statistique du canal dans ce cas, aucun résultat général n'a pu être établi avec l'utilisation de codes LDPC. Cependant, en fonction du niveau de bruit gaussien et de la valeur de M caractérisant le beat noise, un récepteur basé sur un modèle ou un autre peut s'avérer efficace.

Afin de compléter l'étude réalisée dans le cadre des codes correcteurs appliqués aux systèmes OCDMA, les performances des codes LDPC sont comparées par la suite à celles des codes RS de référence avant d'étudier l'application des codes LPDC à une transmission multimédia.

3.4.3 Comparaison des performances des codes LDPC avec les codes RS

La chaîne de transmission avec codage RS, est identique à celle utilisant les codes LDPC, à la différence que la variable décisionnelle est comparée à un seuil de décision afin de fournir une valeur 0 ou 1 à l'entrée du décodeur.

Le décodage est ensuite réalisé comme décrit dans le chapitre 1, afin d'extraire en sortie les K symboles utiles du mot reçu. Afin de comparer les performances des codes LDPC aux codes RS, des simulations ont été réalisées avec le code RS(255,239). Pour rappel, les symboles sont en fait sur 8 bits et la taille des mots transmis est alors de $255 \times 8 = 2040$ bits. Afin de comparer les performances, le code LDPC(2715,2535) de redondance équivalente a été choisi. La figure 3.23 représente les TEB obtenus avec ces deux codes correcteurs d'erreurs, en fonction du nombre d'utilisateurs actifs U pour la famille OOC($F=534$, $W=3$). De plus, dans le cas présenté, on considère uniquement l'impact de l'IAM.

On constate que les performances obtenues sont meilleures avec le code LDPC qu'avec le code RS de taille et redondance équivalentes. En effet, tout comme dans le cas du canal optique sans CDMA, le TEB sans codage est plus fortement réduit avec l'utilisation du code LDPC que le code RS.

De plus, afin d'évaluer le gain apporté par chacun de ces codes, les performances théoriques dans le cas non codé ont été reportées pour les familles OOC(5340,3) et OOC(16000,3). On peut alors remarquer que pour atteindre un TEB de 10^{-6} avec $U = 30$ utilisateurs, dans le cas non codé, une longueur $F = 5340$ est nécessaire. Avec le code RS(255,239) et les mêmes contraintes, la dimension est réduite à $F = 534$, ce qui permet, sans prendre en compte la redondance, d'avoir un débit 10 fois supérieur. Sachant que les

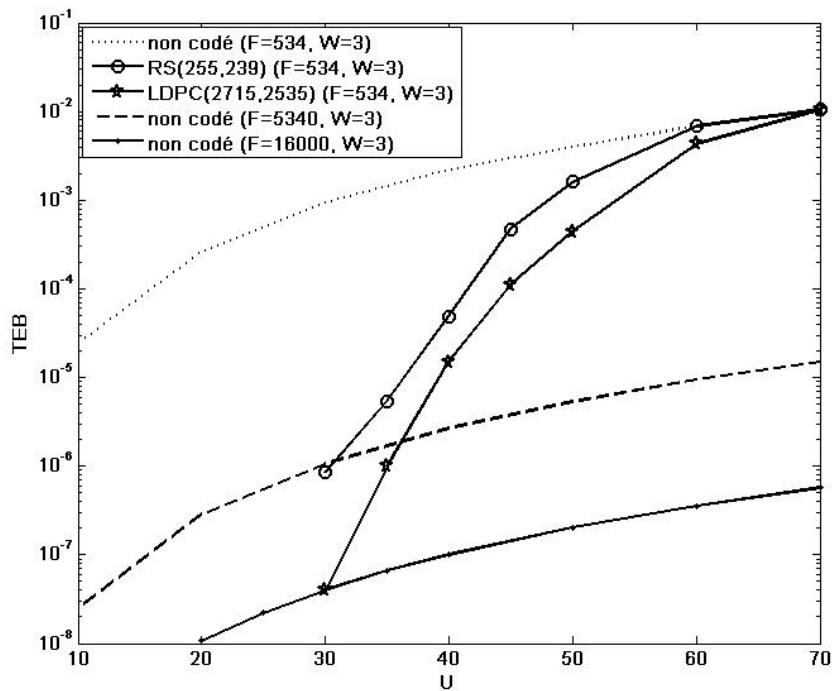


FIG. 3.23 – Comparaison des performances du code RS(255,239) et du code LDPC(2715,2535)

codes utilisés sont peu redondants ($R = 0,93$), le débit utile est en fait 9,3 fois supérieur au lieu de 10 fois. Avec le code LDPC(2715,2535) de même redondance que le code RS étudié, le TEB avec 30 utilisateurs et la famille OOC(534,3) est encore plus faible qu'avec le code RS : 3×10^{-8} . Ce TEB n'est atteint dans le cas sans codage qu'avec une famille OOC de longueur $F = 16000$. Ainsi, dans ce cas, le débit utile est presque 30 fois plus élevé que dans le cas sans codage.

Ainsi, on peut en déduire que, par rapport au code RS, le code LDPC permet non seulement de gagner en TEB, mais encore plus en débit utile, ce qui permet d'envisager l'utilisation du codage correcteur pour améliorer les potentialités de la technique OCDMA pour la transmission multimédia.

3.5 Application des LDPC à un réseau multiservice utilisant la technique OCDMA

3.5.1 Description du système

3.5.1.1 Principe

Un système multimédia utilisant la technique OCDMA, se base sur l'utilisation des codes multi-poids MWOOC présentés au début du chapitre. D'une manière générale, si on considère la transmission de S services différents, chacun caractérisé par un TEB différent, il faut alors prévoir S familles de codes ayant des poids différents. De plus, cette différenciation en TEB est couplée à une différenciation en débit, réalisée par la technique de "parallel mapping" consistant à affecter plusieurs matrices à un même utilisateur.

Pour la différenciation de TEB, si on considère par exemple trois services différents, issus d'une famille de code MWOOC($L \times F, W = L$), on notera $L_1 = L$ et M_1 le poids et le nombre de matrices affectés à la classe de service la plus haute (possédant le plus faible TEB) notée classe 1. Ensuite, si M est le nombre total de matrices pouvant communiquer simultanément, on utilise $M - M_1$ matrices pour effectuer la décomposition en sous-poids L_2 et L_3 tels que $L_2 + L_3 = L$ et $L_2 > L_3$. Les $M_2 = M_3 = M - M_1$ matrices correspondant respectivement aux poids L_2 et L_3 sont affectées aux classes de services moyenne (classe 2) et basse (classe 3).

Pour réaliser en plus la différenciation de débit, il suffit d'affecter par exemple 4 matrices parmi les M_1 à chaque utilisateur de la classe 1, 2 matrices parmi les M_2 à chaque utilisateur de la classe 2 et qu'une seule à la classe 3. Dans ce cas, les débits seront doublés entre chaque classe, c'est-à-dire : $D_1 = 2D_2 = 4D_3$.

L'intercorrélacion entre deux matrices est soit nulle (si aucune longueur d'onde n'est

en commun, comme c'est le cas entre une matrice de la classe 2 et une matrice de la classe 3), soit égale à 1 (si au moins une longueur d'onde et un temps chip sont en commun). Le nombre de longueurs d'onde en commun entre les classes i et j est alors noté $N_\lambda(i, j)$. Comme l'intercorrélacion est non nulle, elle constitue ainsi une source d'IAM, qui limite les performances comme décrit dans le paragraphe suivant.

3.5.1.2 Analyse des performances

En considérant le cas général où S classes de services coexistent, on peut écrire la variable décisionnelle d'un utilisateur de la classe j avec une réception CCR, comme :

$$Z_j = b \cdot L_j + \sum_i I_i^{(j)} \quad (3.99)$$

Où $I_i^{(j)}$ correspond au terme d'IAM dû à l'ensemble des utilisateurs de la classe i sur l'utilisateur désiré de la classe j , et b est la donnée émise par l'utilisateur désiré.

Soit T_j le seuil de décision du CCR correspondant à l'utilisateur désiré, en ne considérant que l'impact de l'IAM, une erreur apparaît alors sur les '0' émis si le terme d'IAM est supérieur au seuil c'est-à-dire : $I_i^{(j)} \geq T_j$. De même que dans le cas simple service, il n'y a pas d'erreur sur les '1' si le seuil T_j est inférieur ou égal au poids.

La probabilité qu'un chip d'un utilisateur non désiré interfère avec un des chips de l'utilisateur désiré est donné de manière similaire au cas simple service (3.9) mais où le poids est remplacé par le nombre de longueurs d'onde en commun entre les deux classes de services $N_\lambda(i, j)$, c'est-à-dire :

$$p_{i,j} = \frac{N_\lambda(i, j)}{2F} \quad (3.100)$$

A partir de (3.100), la probabilité que le terme d'interférence $I_i^{(j)}$ soit égal à une certaine

3.5 Application des LDPC à un réseau multiservice utilisant la technique OCDMA

valeur m_i est obtenue selon la loi binomiale par :

$$p(I_i^{(j)} = m_i) = \binom{M_{i,j}}{m_i} (p_{i,j})^{m_i} (1 - p_{i,j})^{M_{i,j} - 1 - m_i} \quad (3.101)$$

Avec $M_{i,j}$ défini comme le nombre de matrices de la classe i pouvant interférer sur l'utilisateur désiré de la classe j et donné par :

$$M_{i,j} = \begin{cases} M_i & \text{si } i \neq j \\ M_i - 1 & \text{si } i = j \end{cases} \quad (3.102)$$

Ceci permet finalement d'obtenir la probabilité d'erreur pour un utilisateur de la classe j en considérant l'IAM due à toutes les matrices communicantes de chacune des autres classes :

$$P_{e,j} = \frac{1}{2} \sum_{m_1=0}^{M_{1,j}} \cdots \sum_{m_S=0}^{M_{S,j}} p(I_1^{(j)} = m_1) \cdots p(I_S^{(j)} = m_S) \cdot p\left(\sum_i m_i \geq T_j\right) \quad (3.103)$$

Si on considère, en plus de l'IAM, l'impact d'un bruit gaussien de variance σ^2 , la probabilité d'erreur s'obtient de manière similaire au cas simple service [76], c'est-à-dire :

$$P_{e,j} = \frac{1}{4} \sum_{m_1=0}^{M_{1,j}} \cdots \sum_{m_S=0}^{M_{S,j}} p(I_1^{(j)} = m_1) \cdots p(I_S^{(j)} = m_S) \times \left(\operatorname{erfc}\left(\frac{T_j - \sum_i m_i}{\sigma\sqrt{2}}\right) + \operatorname{erfc}\left(\frac{L_j + \sum_i m_i - T_j}{\sigma\sqrt{2}}\right) \right) \quad (3.104)$$

Notons alors que les probabilités d'erreurs (3.103) ou (3.104) ne tiennent compte que de la différenciation en TEB, c'est pourquoi on parle de matrices communicantes et non d'utilisateurs actifs. Ainsi, pour dimensionner une famille MWOOC par rapport

au contexte, on part d'abord du nombre de matrices qui sont ensuite indépendamment affectées aux utilisateurs.

Pour illustrer cette étude, un exemple faisant intervenir trois classes de services est présenté. Afin de chiffrer les débits utiles correspondants, on suppose que la bande passante électrique est donnée par $B_e = 10$ GHz. Tout d'abord, le dimensionnement de la famille MWOOC ainsi que la décomposition en sous-poids sont effectués dans le cas idéal non bruité (prise en compte uniquement de l'IAM). Ensuite, nous évaluerons la dégradation apportée par la prise en compte du bruit.

Dans le cas sans bruit, on fixe tout d'abord le nombre de matrices communicantes en simple service ($TEB=10^{-9}$) à $M = 32$. A partir de l'expression de la probabilité d'erreur (3.28) dans le cas du simple service, on trouve la famille MWOOC($12 \times 163, 12$). Cela correspond à un débit par matrice de $D = 10G/163 \approx 61$ Mb/s.

Pour effectuer la différenciation en TEB, on construit à partir du code MWOOC($12 \times 163, 12$) les familles de sous-poids égaux à 8 et 4. En choisissant $M_1 = 16$, on a $M_2 = M_3 = M - M_1 = 16$, on vérifie d'après (3.104) que : $P_e(1) < 10^{-9}$, $P_e(2) < 10^{-6}$ et $P_e(3) < 10^{-3}$.

Ensuite, pour effectuer la différenciation en débits, on affecte quatre matrices à chaque utilisateur de la classe 1, deux à chaque utilisateur de la classe 2 et une seule pour les utilisateurs de la classe 3. Ainsi, cela permet une transmission simultanée de trois classes de service de TEB et débits différents, selon les caractéristiques reportées tableau 3.1, avec U_i le nombre d'utilisateurs dans la classe de service i .

Afin d'illustrer l'impact du bruit sur les performances, la figure 3.24 représente les TEB obtenus en fonction du S/B pour chacune des classes de service.

On remarque que pour un S/B supérieur à 18 dB, le TEB de 10^{-3} de la classe de service 3 est atteint. Cependant, pour les classes de services 2 et 1, les TEB de 10^{-6}

3.5 Application des LDPC à un réseau multiservice utilisant la technique OCDMA

Classe i	TEB	Débit	M_i	U_i
1	10^{-9}	244 Mb/s	16	4
2	10^{-6}	122 Mb/s	16	8
3	10^{-3}	61 Mb/s	16	16

TAB. 3.1 – TEB, débit, et nombre d'utilisateurs U_i pour chaque classe de service, récepteur CCR, sans codage

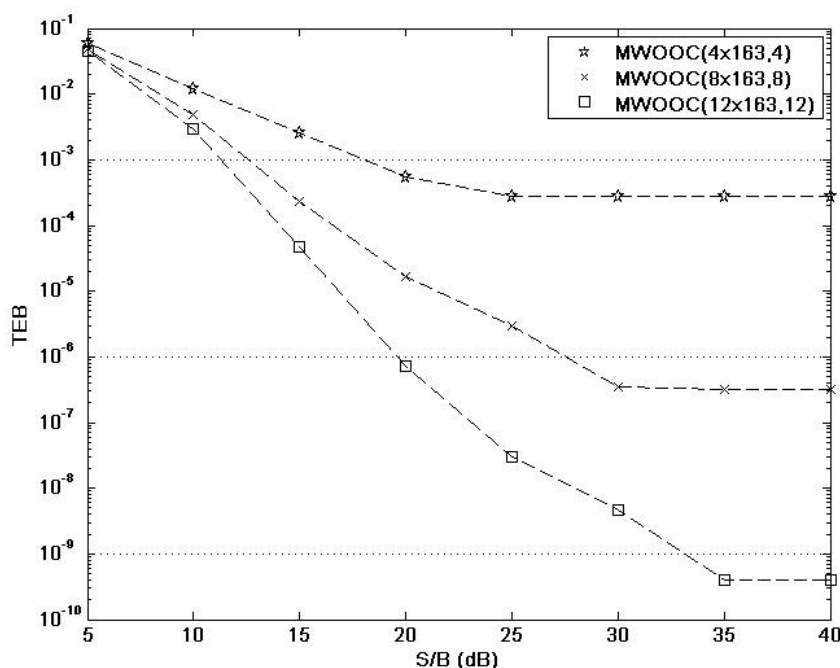


FIG. 3.24 – TEB pour les trois classes de services avec $M_1 = M_2 = M_3 = 16$

et 10^{-9} ne sont atteints respectivement que pour des S/B supérieurs à 27 dB et 33 dB. Ainsi, le système simple n'est pas suffisamment robuste au bruit pour une application multimédia utilisant la technique OCDMA.

Pour remédier à ce problème, une solution utilisant le codage correcteur d'erreurs avec les codes LDPC est envisagée. Avant d'évaluer les gains apportés par une telle technique, l'adaptation réalisée dans le calcul des LLR par rapport au contexte multimédia OCDMA est présentée.

3.5.2 Performances du système multimédia avec codes LDPC

3.5.2.1 Adaptation des LLR au contexte multimédia OCDMA

Afin de déterminer les LLR dans le contexte multimédia utilisant la technique OCDMA, on se base sur le travail réalisé pour le cas simple service, avec le modèle IAM + bruit gaussien. Dans ce cas, il reste à déterminer la variance de l'IAM sur le temps bit par rapport à l'utilisateur désiré de la classe j . De manière similaire au cas simple service, on peut tout d'abord évaluer la variance de l'IAM provenant de l'ensemble des utilisateurs de la classe i par rapport à la classe j de l'utilisateur désiré par :

$$\sigma_{i,j}^2 = M_{i,j} \cdot p_{i,j} \quad (3.105)$$

En considérant le terme d'interférence provenant de toutes les classes de services, on peut alors écrire la variance du terme d'IAM global comme :

$$\sigma_j^2 = \sum_i \sigma_{i,j}^2 = \sum_i M_{i,j} \cdot p_{i,j} \quad (3.106)$$

Ensuite, le calcul des LLR s'effectue comme présenté dans la partie simple service à partir de (3.88) et avec la variance relative à l'IAM donnée par (3.106)

3.5.2.2 Gain en TEB et en S/B

A partir des résultats de simulations et en extrapolant ces résultats, nous avons pu tracer figure 3.25 les performances des trois classes de services étudiées précédemment, pour les mêmes nombres de matrices communicantes ($M_1 = M_2 = M_3 = 16$) avec l'utilisation du code LDPC(2715,2535). Les performances précédemment obtenues sans codage correcteur d'erreurs sont également reportées.

3.5 Application des LDPC à un réseau multiservice utilisant la technique OCDMA

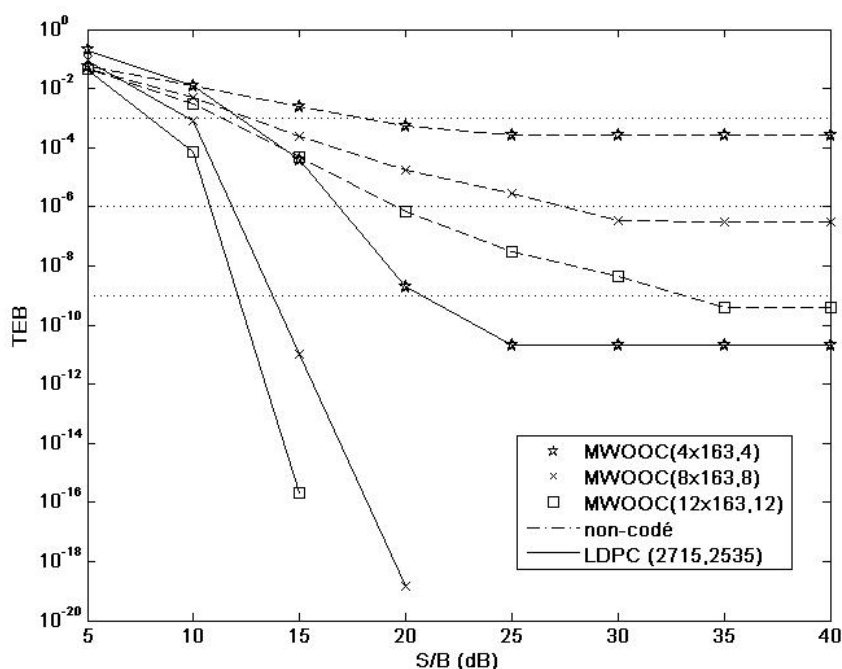


FIG. 3.25 – TEB pour les trois classes de services avec $M_1 = M_2 = M_3 = 16$ avec le code LDPC(2715,2535)

On remarque tout d'abord que les TEB sont fortement réduits avec codage correcteur, pour chacune des classes de services. En effet, le TEB de la classe 3, (initialement à 10^{-3} pour des S/B élevés), est réduit à un TEB en dessous de 10^{-10} dans les mêmes conditions. Concernant les classes 1 et 2, les TEB sont significativement réduits, même pour de faibles valeurs de S/B.

Cela signifie donc que l'utilisation du code LDPC permet de relâcher les contraintes sur le S/B minimal requis pour atteindre un certain TEB. En effet, dans le cas sans bruit, nous avons vu que les TEB de 10^{-9} , 10^{-6} et 10^{-3} relatifs aux classes 1, 2, et 3 sont respectivement atteints pour des S/B de : 33 dB, 27 dB et 18 dB. Avec le code LDPC(2715,2535), ces TEB sont respectivement atteints pour des S/B de : 12 dB, 12 dB et 13 dB. Ceci signifie que le gain en S/B est d'autant plus important que la classe de

service est haute puisqu'il est alors de $33 - 12 = 21$ dB pour la classe 1, de $27 - 12 = 15$ dB pour la classe 2 et de $18 - 13 = 5$ dB pour la classe 3. En effet, plus la classe est haute, plus le TEB correspondant sans codage est faible, et plus le code est performant, d'où un meilleur gain en TEB et en S/B.

Le paragraphe suivant permet de mettre en évidence l'efficacité du codage correcteur d'erreurs à augmenter le nombre d'utilisateurs actifs ainsi que les débits utiles, malgré l'ajout de redondance.

3.5.2.3 Gain en nombre d'utilisateurs actifs et en débits

Etant donné que l'utilisation de codes LDPC permet de diminuer de manière significative les TEB, cela signifie aussi que pour un TEB visé, le nombre de matrices communicantes peut être supérieur au cas non codé.

A titre d'exemple, nous avons tout d'abord cherché avec le même partitionnement que précédemment, pour un S/B de 20 dB, et en gardant le même nombre de matrices affectées par utilisateur dans chaque classe (c'est-à-dire : 4, 2, et 1), le nombre d'utilisateurs pouvant communiquer simultanément. Les résultats sont reportés dans la quatrième colonne du tableau 3.2 (cas 1), qui reporte aussi à titre de comparaison les débits et nombre d'utilisateurs dans chaque classe pour le cas non codé, sans bruit et avec un S/B de 20 dB. Les débits utiles sont légèrement diminués dans le cas codé dû à l'ajout de redondance. Cependant, le nombre d'utilisateurs est beaucoup plus élevé que dans le cas non codé à un S/B de 20 dB. Même par rapport au cas non codé idéal sans bruit, on peut noter une augmentation significative du nombre d'utilisateurs actifs.

La dernière colonne du tableau 3.2 (cas 2) regroupe les débits et nombres d'utilisateurs obtenus en changeant le nombre de matrices par utilisateur dans chaque classe de service par rapport au cas précédent. Cela permet de compenser la perte en débit utile liée à

3.5 Application des LDPC à un réseau multiservice utilisant la technique OCDMA

Classe (TEB)		non-codé		LDPC - S/B = 20dB	
		cas idéal (IAM)	cas bruité S/B=20dB	cas 1	cas 2
1	$D_1(\text{Mb/s})$	244	244	228	457
(10^{-9})	U_1	4	2	10	4
2	$D_2(\text{Mb/s})$	122	122	114	343
(10^{-6})	U_2	8	2	20	8
3	$D_3(\text{Mb/s})$	61	61	57	114
(10^{-3})	U_3	16	4	26	17
Nombre total d'utilisateurs		28	8	56	29

TAB. 3.2 – Améliorations du système avec le code LDPC(2715, 2535), récepteur CCR

l'ajout de redondance. Il y a alors respectivement 8, 6 et 2 matrices affectées pour chaque utilisateur de la classe 1, 2 et 3, ce qui conduit aux débits utiles de 457 Mb/s, 343 Mb/s et 114 Mb/s. Ainsi, la perte en débit est finalement plus que compensée puisqu'on a des débits supérieurs au cas non codé.

De plus, on peut noter que les nombres d'utilisateurs actifs dans chaque classe de service sont plus élevés que dans le cas non codé pour le même S/B, et équivalents à ceux du cas non codé idéal. Ainsi, par rapport au cas non codé, cette dernière solution permet de gagner à la fois en nombre d'utilisateurs actifs et en débits utiles. Des solutions basées sur des compromis entre ces deux solutions peuvent être aussi choisies.

Les performances peuvent être encore améliorées si on considère l'utilisation d'autres récepteurs. Dans un souci de complexité d'implémentation, seul le limiteur avec CCR peut être envisagé en dehors du CCR simple.

3.5.2.4 Impact d'un limiteur sur les performances

De part sa fonction, le limiteur réduit la quantité d'IAM, et améliore d'une manière simple les performances dans le cadre de transmissions OCDMA simple service ou multi-services.

Le dimensionnement des codes multi-poids MWOOC adaptés à la transmission de trois classes de service avec un CCR et un limiteur, a été effectué dans le cas sans bruit au cours d'une thèse précédemment menée au laboratoire [76]. Les résultats obtenus sur la base de 90 matrices communicantes, trois TEB ($TEB_3 = 10^{-3}$, $TEB_2 = 10^{-6}$ et $TEB_1 = 10^{-9}$) et trois débits ($D_3 = D$, $D_2 = 2D$ et $D_1 = 4D$) ont conduit à partitionner le code MWOOC($12 \times 211, 12$), en familles de sous-poids égaux à 8 et 4. Il est ainsi possible d'avoir 60 matrices communicantes dans la classe 1 et 30 matrices communicantes partitionnées en deux dans les classes 2 et 3.

Les débits et nombres d'utilisateurs obtenus sont reportés dans le tableau 3.3.

Classe i	TEB	Débit	M_i	U_i
1	10^{-9}	188 Mb/s	60	15
2	10^{-6}	94 Mb/s	30	15
3	10^{-3}	47 Mb/s	30	30

TAB. 3.3 – TEB, débit, et nombre d'utilisateurs U_i pour chaque classe de service pour le cas avec limiteur, sans bruit et sans codage

La robustesse au bruit du système a également été étudiée et il a été montré que pour atteindre les TEB correspondant aux classes de service haute, moyenne et basse, il faut des S/B respectivement égaux à : 34,5 dB, 29 dB et 20,5 dB.

Il est possible de tenir compte de l'impact du bruit dans le dimensionnement des codes adaptés aux trois classes, mais cela implique une augmentation de la longueur temporelle donc une diminution des débits et/ou une augmentation des longueurs d'onde requises [76].

Afin d'éviter ces conséquences, l'utilisation du codage correcteur d'erreurs a donc été envisagée.

Les résultats suivants illustrent les performances obtenues avec l'utilisation conjointe du limiteur et de codage correcteur d'erreurs.

3.5 Application des LDPC à un réseau multiservice utilisant la technique OCDMA

- Tout d’abord les performances obtenues avec la famille MWOOC(12×211,12) et les sous-familles de poids 8 et 4, ont été évaluées avec le code LDPC(905,725) de rendement $R=0,8$ en présence de bruit. Les TEB étant fortement diminués par rapport au cas sans codage, les valeurs minimales des S/B requis pour atteindre les TEB correspondant aux différentes classes de services sont plus faibles. En effet, pour les classes haute, moyenne et basse, les S/B minimaux sont dans ce cas égaux respectivement à : 15, 1dB, 13, 4 dB et 11, 6 dB. Ainsi, le gain en S/B, d’autant plus grand que le TEB visé est faible, permet de relâcher les contraintes sur les composants du système.
- Par ailleurs, cette amélioration de performances signifie aussi qu’on peut envisager de la même manière que dans le cas du CCR, un plus grand nombre de matrices communicantes donc d’utilisateurs actifs. Toutefois on est limité dans cette augmentation par la cardinalité des codes MWOOC.
- D’autre part, le gain en performance lié au codage correcteur signifie qu’il est possible pour une performance donnée, d’utiliser des familles de codes de longueur plus faible, générant plus d’IAM, mais conduisant à des débits plus élevés. Pour illustrer ce point, nous repartons du cas idéal sans bruit. Nous avons ainsi établi que la famille MWOOC(7×137,7) et les sous-familles de poids 4 et 3, permettent de satisfaire les TEB visés grâce au codage correcteur d’erreurs basé sur le code LDPC(905,725). Donc, grâce à la réduction significative de la durée temporelle du code, les débits sont augmentés malgré l’ajout de redondance. Les résultats sont présentés dans le tableau 3.4.

Les débits ont été améliorés par rapport au cas sans codage, et de plus, nous avons pu remarquer une meilleure résistance au bruit. En effet, pour atteindre en présence de bruit gaussien les trois TEB requis, il faut des S/B minimaux respectivement égaux à :

Classe i	TEB	Débit	M_i	U_i
1	10^{-9}	234 Mb/s	30	15
2	10^{-6}	117 Mb/s	30	15
3	10^{-3}	58 Mb/s	30	30

TAB. 3.4 – TEB, débit, et nombre d'utilisateurs U_i pour chaque classe de service pour le cas avec limiteur et code LDPC(905,725)

26, 3 dB, 22, 5 dB et 15, 8 dB.

Ainsi, par cet exemple, nous avons mis en évidence que l'utilisation de codage correcteur d'erreurs permet non seulement de plus hauts débits mais aussi une meilleure résistance au bruit.

Cependant, du fait de la cardinalité des codes MWOOC, les performances des codes LDPC ne sont pas, dans ce cas, exploitées entièrement. En effet, théoriquement, pour respecter les TEB visés, avec codage correcteur d'erreurs et le limiteur, des familles de longueurs plus courtes pourraient être utilisées. Or, comme le nombre de matrices disponibles dans une famille de code décroît aussi avec la longueur temporelle, il n'est pas possible d'avoir le nombre de matrices désiré. Ainsi, dans ce cas, la cardinalité imposée par la méthode de construction MWOOC, limite la longueur minimale de la famille de code et donc les débits disponibles.

3.5.3 Conclusions

L'étude du codage correcteur d'erreurs dans le cadre de l'OCDMA pour les transmissions multimédia a permis de mettre en évidence l'intérêt d'une telle technique. En effet, avec un récepteur CCR simple et des codes correcteurs LDPC peu redondants, nous avons montré qu'il était possible de fournir des débits intéressants à un nombre significatif d'utilisateurs et de réduire le TEB de la transmission. Ces résultats ont fait l'objet de publications traitant des cas de transmission idéale [94] et en présence de bruit

3.6 Conclusion

gaussien [95]. De plus, l'ajout d'un module limiteur devant le CCR a permis d'illustrer les performances des codes LDPC sur les TEB, débits et nombre d'utilisateurs actifs. Ces études, présentant les performances avec un module limiteur ont également été publiées dans [90].

Même si dans les exemples présentés, les performances des codes correcteurs d'erreurs ne sont donc pas exploitées complètement en présence du limiteur, de part la limitation imposée par la cardinalité des familles MWOOC, l'ensemble de ces résultats montre l'intérêt d'utiliser un code correcteur d'erreurs, que ce soit avec un récepteur CCR, ou avec un limiteur suivi d'un CCR. Ainsi, ces études illustrent la potentialité d'un système de transmission utilisant le codage correcteur d'erreurs ainsi que la technique OCDMA permettant la différenciation des TEB et des débits pour des applications multimédia.

3.6 Conclusion

Dans ce chapitre, nous avons étudié la statistique du signal OCDMA en prenant en compte les différentes limitations du système qui sont l'IAM, le bruit gaussien, et le bruit de battement dû à la photodétection. A partir de la statistique binomiale de l'IAM différents modèles ont été établis afin de caractériser la variable décisionnelle. Notamment, le modèle statistique χ^2 établi au chapitre 2 a servi de base pour définir un modèle en mixture de χ^2 décrivant les battements aux niveau de la photodétection entre les différentes interférences.

Ces modèles ont permis d'évaluer théoriquement et par simulation, les performances du système de transmission OCDMA et de quantifier ainsi les dégradations des performances en fonction de différents paramètres. De plus, afin de compenser les dégradations introduites par ces différentes limitations, nous avons étudié l'amélioration liée à

l'utilisation de codes correcteur d'erreurs de types LDPC. A partir des modèles statistiques établis, nous avons pu déterminer les paramètres nécessaires au décodage optimal des informations. Dans chacun des cas envisagés, l'expression analytique des LLR a été déterminée. Les simulations réalisées avec la chaîne de transmission codée ont mis en évidence l'efficacité des codes LDPC à corriger les erreurs dues à l'IAM mais aussi au bruit gaussien et au beat noise.

Finalement, une illustration de l'application des codes LDPC aux transmissions OCDMA dans le cadre d'un système multimédia (plusieurs TEB et plusieurs débits) a permis de mettre en évidence la potentialité de l'utilisation de ce type de codage pour améliorer la qualité de service proposée.

Conclusion générale et perspectives

Ce mémoire présente l'étude de l'application des codes correcteurs d'erreurs aux communications optiques en vue d'établir un système de transmission capable de fournir de hauts débits à bas coût. Le projet de recherche ANR ECOFRAME (Eléments de CONvergence pour les Futurs Réseaux d'Accès et METropolitains haut débit), qui s'est déroulé en parallèle de cette thèse, illustre le besoin de nouvelles solutions afin d'améliorer la qualité des transmissions optiques. Parmi les différents types de codes correcteurs, les codes en blocs LDPC (Low Density Parity Check codes) ont été envisagés comme une solution pour améliorer les performances de transmissions optiques filaires dans différentes configurations.

Le premier chapitre de cette thèse présente les différents types de codes correcteurs d'erreurs pour dégager les principales caractéristiques du codage et du décodage. De part les contraintes imposées par l'optique (temps de traitement, débits rapides), les codes en blocs apparaissent comme prometteurs et sont donc plus particulièrement étudiés. Parmi eux, les codes à faible densité LDPC sont envisagés ainsi que les codes de Reed-Solomon, qui servent de référence. Ce chapitre permet, par l'illustration des différents schémas de codage, d'aborder les notions essentielles utilisées dans les chapitres suivants.

Le deuxième chapitre se consacre à l'étude du codage correcteur d'erreurs dans le cadre d'une transmission simple où un seul utilisateur occupe une longueur d'onde (cas

mono-utilisateur ou bien multiplexage en longueur d'onde). Dans ce cas, les effets dus à la chaîne de transmission et au photo-récepteur sont pris en compte dans la dégradation des performances. Deux types de canaux ont été étudiés : le canal classique considérant un bruit additif gaussien de moyenne nulle (AWGN : Additive White Gaussian Noise), et un canal à statistique χ^2 traduisant les battements dus à la détection quadratique. En effet, le modèle χ^2 représentant la statistique de la somme de carré de variables gaussiennes traduit l'élévation au carré des composantes du signal reçu par la photodiode. De plus, des simulations physiques réalisées par des partenaires du projet ECOFRAME ont mis en évidence l'adéquation de cette statistique par rapport au contexte d'une transmission optique. Avec chacun des deux modèles de canal, les performances des codes LDPC ont été évaluées et comparées à celles des codes RS de référence. Nous avons montré l'importance de la connaissance de la statistique du canal au niveau du récepteur, qui ne fournit des résultats optimaux que s'il est correctement adapté au canal. La dernière partie de ce chapitre est une application du codage correcteur d'erreurs au réseau ECOFRAME et permet de conclure, par un dimensionnement de codes capables d'améliorer un TEB de 10^{-5} à 10^{-12} , sur la potentialité des codes correcteurs dans un système de communication optique.

Le troisième et dernier chapitre se place dans le cadre d'une transmission optique utilisant la technique d'accès multiple par répartition de codes, ou OCDMA (Optical Code Division Multiple Access). Plusieurs études déjà menées dans le laboratoire ESTE sur l'OCDMA ont montré que les interférences entre utilisateurs constituent une limitation importante dégradant les performances. Dans le but de compenser les erreurs introduites par ces interférences, nous avons envisagé l'application de codes correcteurs, notamment des codes LDPC sur un tel système.

Or, ayant établi au cours du chapitre 2 que la connaissance de la statistique au niveau

du décodeur est importante, nous avons tout d'abord analysé le signal reçu dans une chaîne de transmission OCDMA. Ainsi, de manière similaire au chapitre 2, l'impact de la photodétection sur la statistique du signal transmis a été étudié et a abouti à un modèle statistique basé sur une mixture de χ^2 permettant de traduire les battements dus à la photo-détection. Les performances de la chaîne de transmission OCDMA, en tenant compte non seulement des interférences, mais aussi du bruit de battements ont ainsi pu être établies. L'application des codes LDPC a ensuite été étudiée. L'amélioration qu'ils apportent a permis de montrer l'intérêt d'une telle technique. Afin de conclure ce chapitre, le cas d'une transmission multimédia utilisant la technique OCDMA pour la différenciation de services (TEB et débit) a été envisagé en présence de codes correcteurs de type LDPC. Les taux d'erreurs binaires, mais aussi les débits et nombres d'utilisateurs ont dans ce cas, été fortement améliorés. Ainsi, l'application des codes correcteurs d'erreurs à l'OCDMA permet d'illustrer la potentialité de la technique pour des applications multimédia.

Finalement, l'ensemble du travail réalisé au cours de cette thèse a permis d'illustrer que les codes LDPC fournissent une solution intéressante pour améliorer les performances d'un système de transmission optique, mais que la connaissance de la statistique par le décodeur constitue un point essentiel. De plus, deux cas ont été étudiés pour l'application des codes correcteurs d'erreurs, l'un traitant d'une transmission optique simple mono-utilisateur, l'autre d'une transmission optique utilisant la technique d'accès multiple OCDMA.

Afin de confirmer que l'utilisation de codes LDPC constitue une solution viable pour les communications optiques à haut débit, une des perspectives des travaux réalisés au cours de cette thèse consiste à établir un démonstrateur incluant le codage et le décodage des informations sur circuits, de type FPGA (Field Programmable Gate Array) par exemple.

Ces études, en continuité des travaux réalisés dans le cadre du projet ECOFRAME permettraient de conclure et d'illustrer la faisabilité d'un tel système.

Une autre voie de recherche consiste à étudier d'autres types de canaux optiques. L'optique sans fil par exemple, est actuellement envisagée comme alternative à la radio dans les milieux sensibles d'un point de vue électromagnétique (hôpital par exemple). Des études récemment menées dans notre équipe de recherche concernant le canal optique sans fil en milieu confiné, ont permis de vérifier que les performances sont vite limitées par les trajets multiples et les pertes liées aux différents rebonds. Si on envisage par ailleurs une certaine mobilité, les performances peuvent vite se dégrader. Ainsi, l'insertion de codage correcteur d'erreurs avec un décodage adapté à la statistique particulière du canal optique sans fil pourrait être une solution pour améliorer les performances et donc les potentialités en terme notamment de couverture.

De plus, dans ce cadre, la modulation la plus adaptée n'est pas forcément la modulation simple OOK envisagée dans cette étude. Ainsi, il paraît intéressant d'étudier l'impact de codes correcteurs en présence de différents types de modulations, tels que la PPM (Pulse Position Modulation) et les modulations dérivées qui en découlent.

Par ailleurs, l'étude des codes correcteurs dans le cadre de l'OCDMA a été effectuée en considérant comme limitations, l'interférence et le bruit de battements dus à la détection. Pour cela, la statistique du canal a été modélisée par une mixture de χ^2 , s'inspirant de la statistique χ^2 validée au cours du chapitre 2. Cependant, une étude complémentaire consisterait à valider ce modèle soit par des vérifications expérimentales, soit par des simulations systèmes de la chaîne de transmission complète. Cette étude pourrait également porter sur le cas où on considère non seulement un bruit de battements mais aussi un bruit de type gaussien, afin d'établir dans ce cadre un modèle statistique. Enfin, les recherches peuvent aussi se poursuivre en incluant au dimensionnement des

codes OCDMA pour le multimédia, la présence de beat noise et éventuellement du bruit gaussien.

Finalement, ces différentes voies de recherches peuvent se rejoindre, en envisageant par exemple, le codage correcteur d'erreurs dans le cadre de l'accès multiple OCDMA, utilisant différentes modulations, et pourquoi pas dans le cadre de l'application de la technique OCDMA à une transmission optique en milieu confiné.

Glossaire

AMRC : Accès Multiple par Répartition de Code (=CDMA)

APD : Avalanche Photo Diode

ASE : Amplified Spontaneous Emission

AWGN : Additive White Gaussian Noise

BCH : Bose-Chaudhuri-Hocquenghem

BIBD : Balanced Incomplete Block Design

BP : Belief Propagation (Propagation de croyance)

CCR : Conventional Correlation Receiver

CDMA : Code Division Multiple Access (=AMRC)

CRC : Cyclic Redundancy Check

DCF : Dispersion Compensating Fiber

DPSK : Differential Phase Shift Keying

DSP : Digital Signal Processing

ECOFAME : Eléments de CONvergence pour les Futurs Réseaux d'Accès et
MEtropolitains haut débit

EDFA : Erbium Doped Fiber Amplifier

FEC : Forward Error Correction

FPGA : Field Programmable Gate Array

GF : Galois Field

HL : Hard Limiter

IAM : Interférence d'Accès Multiple

ITU : International Telecommunications Union

LDPC : Low Density Parity Check

LED : Light Emitting Diode

LLR : Logarithmic Likelihood Ratio

MAP : Maximum A Posteriori

ML : Maximum Likelihood

MWOOC : Multi-Wavelength Optical Orthogonal Code

NCG : Net Coding Gain

NSC : Non Systematic Convolutional (code)

OOK : On Off Keying

PIC : Parallel Interference Cancelation (receiver)

PIN : Positive Intrinsic Negative (photodiode jonction PN + couche de matériau intrinsèque)

RIN : Relative Intensity Noise

RS : Reed Solomon

RSC : Recursive Systematic Convolutional (code).

OOC : Optical Orthogonal Code

SIC : Successive Interference Cancelation (receiver)

SOA : Semiconductor Optical Amplifier

TEB : Taux d'Erreur Binaire

VRC : Vertical Redundancy Check

WDM : Wavelength Division Multiplexing

Références

- [1] I.S. Reed and G. Solomon. Polynominal codes over certain finite fields. *SIAM J. Appl. Math*, 8 :300–304, Juin 1960.
- [2] R.G. Gallager. *Low Density Parity Check Codes*. Monograph, M.I.T. Press, 1963.
- [3] S. Haykin. *Communciation Systems - Third Edition*. John Wiley and Sons, 1931. ISBN : 0471571768.
- [4] P. Elias. Error-free coding. *IEEE Transaction on Information Theory*, pages 29–57, Sept. 1954.
- [5] sous la direction de Alain Glavieux. *Codage de canal des bases théoriques aux turbocodes*. Hermes Science Publications, 2005. ISBN : 2746209535.
- [6] R. Hamming. Error-detecting and error-correcting codes. *J. Bell Syst. Tech.*, 29 :147–160, 1950.
- [7] W. Peterson and D.T. Brown. Error-detecting and error-correcting codes. *Proceedings of the IRE*, 49(1) :228–235, Janv. 1961.
- [8] R.C. Bose and D.K. Ray-Chaudhuri. On a class of error correcting binary group codes. *Inf. Control*, 3 :78–79, Mars 1960.
- [9] A. Hocquenghem. Codes correcteurs d’erreurs. *Chiffres*, 2 :147–156, 1959.

- [10] B. Sklar. *Digital Communications - Second Edition*. Prentice Hall, 2001. ISBN : 0130847887.
- [11] W.W. Peterson. Encoding and error-correction procedures for the bose-chaudhuri codes. *IRE Transactions on Information Theory*, 6(Sept.) :459–470, 1960.
- [12] E.R. Berlekamp. *Algebraic Coding Theory*. McGraw-Hill, New York, 1968.
- [13] J.L. Massey. Shift register synthesis and bch decoding. *IEEE Transactions on Information Theory*, 15 :122–127, 1969.
- [14] Y. Sugiyama, Y. Kasahara, S. Hirasawa, and T. Namekawa. A method for solving key equation for goppa codes. *Information and Control*, 27 :87–99, 1975.
- [15] R. Tanner. A recursive approach to low complexity codes. *IEEE Transactions on Information Theory*, 27(5) :533–547, Sept. 1981.
- [16] J.L. Fan. *Constrained Coding and Soft Iterative Decoding*. Kluwer Academic Publishers, 2001.
- [17] D. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.
- [18] P. Moro and D. Candiani. 565 mb/s optical transmission system for repeaterless sections up to 200 km. *IEEE ICC 1991*, 3 :1217–1221, 1991.
- [19] P.M. Gabla et al. 401 km, 622 mb/s and 357 km, 2.488 gb/s im/dd repeaterless transmission experiments using erbium-doped fiber amplifiers and error correcting code. *IEEE Photonic Technology Letters*, 4(10) :1148–1151, Oct. 1992.
- [20] S. Yamamoto, H. Takahira, and M. Tanaka. 5 gbit/s optical transmission terminal equipment using forward error correcting code and optical amplifier. *IEEE Electronics Letters*, 30(3) :254–255, Février 1994.

- [21] J.L. Pamart et al. Forward error correction in a 5 gbit/s 6400 km edfa based system. *IEEE Electronics Letters*, 30(4) :342–343, Février 1994.
- [22] S. Benedetto and G. Bosco. *Channel coding for optical communications*. E. Forestieri Ed., pp.63-78, Springer, New York, NY, USA, 2005.
- [23] International Telecommunication Union. *Forward Error Correction for Submarine Systems*. 2000. Telecommunication Standardization Sector G. 975.
- [24] International Telecommunication Union. *Interfaces for the Optical Transport Network*. 2003. Telecommunication Standardization Sector G. 709.
- [25] O.A. Sab et al. Concatenated forward error correction schemes for long-haul dwdm optical transmission systems. *European Conference on Optical Communication (ECOC) 1999*, 2 :290–291, Sept. 1999.
- [26] O.A. Sab. Fec techniques in submarine transmission systems. *Optical Fiber Communication Conference and Exhibit, 2001. OFC 2001*, 2 :TuF1–1 – TuF1–3, 2001.
- [27] T. Mizuochi et al. Transparent multiplexer featuring super fec for optical transport networking. *SubOptic 2001*, pages 484–487, 2001.
- [28] International Telecommunication Union. *Forward error correction for high bit-rate DWDM submarine systems*. 2004. Telecommunication Standardization Sector G. 975.1.
- [29] T. Mizuochi. Increase of coding gain offered by fec in optical communication systems. *IEEE Journal of Selected Topics in Quantum Electronics*, Mai/Juin 2006.
- [30] S.Y. Chung et al. On the design of low-density parity check codes within 0.0045 db of the shannon limit. *IEEE Communications Letters*, 5(2) :58–60, Fév. 2001.

- [31] I.B. Djordjevic, M. Cvijetic, L. Xu, and T. Wang. Proposal for beyond 100-gb/s optical transmission based on bit-interleaved ldpc-coded modulation. *IEEE Photonics Technology Letters*, 19(12) :874–876, Juin 2007.
- [32] I.B. Djordjevic, S. Sankaranarayanan, S.K. Chilappagari, and Bane Vasic. Low-density parity-check codes for 40-gb/s optical transmission systems. *IEEE J. of Selected Topics in Quantum Electronics* , 12(4) :555–562, Août 2006.
- [33] I.B. Djordjevic and B. Vasic. Iteratively decodable codes from orthogonal arrays for optical communication systems. *IEEE Communications Letters*, 9(10) :924–926, Oct. 2005.
- [34] I.B. Djordjevic, O. Milenkovic, and B. Vasic. Generalized low-density parity-check codes for optical communication systems. *IEEE Journal of Lightwave Technology*, 23(5) :1939–1946, Mai 2005.
- [35] I.B. Djordjevic and B. Vasic. Nonbinary ldpc codes for optical communication systems. *IEEE Photonics Technology Letters*, 17(10) :2224–2226, Oct. 2005.
- [36] I.B. Djordjevic and B. Vasic. Performance of affine geometry low-density parity-check codes in long-haul optical communications. *European Transactions on Telecommunications*, 15(5) :477–483, Sept. 2004.
- [37] O. Milenkovic, I.B. Djordjevic, and B. Vasic. Block-circulant low-density parity-check codes for optical communication systems. *IEEE J. of Selected Topics in Quantum Electronics*, 10(2) :294–299, Mars 2004.
- [38] B. Vasic and I.B. Djordjevic. A forward error correction scheme for ultra long haul optical transmission systems based on low-density parity-check codes. *IEEE International Conference on Communications, 2003. ICC '03*, 2 :1489–1493, Mai 2003.

- [39] I.B. Djordjevic and B. Vasic. Projective geometry ldpc codes for ultralong-haul wdm high-speed transmission. *IEEE Photonics Technology Letters*, 15(5) :784–786, Mai 2003.
- [40] B. Vasic, I.B. Djordjevic, and R. Kostuk. Low-density parity check codes and iterative decoding for long haul optical communication systems. *IEEE Journal of Lightwave Technology*, 21(2) :438–446, Fév. 2003.
- [41] B. Vasic and I.B. Djordjevic. Low density parity check codes for long haul optical communication systems. *IEEE Photononic Technololy Letters*, 14(8) :1208–1210, Août 2002.
- [42] J.L. Dornstetter. On the equivalence between berlekamp’s and euclid’s algorithm. *IEEE Transactions on Information Theory*, IT-33(3) :428–431, 1987.
- [43] D. Chase. Class of algorithms for decoding block codes with channel measurement information. *IEEE Transactions on Information Theory*, 18(1) :170–182, Janv. 1972.
- [44] J. Wolf. Efficient maximum likelihood decoding of linear block codes using a trellis. *IEEE Transactions on Information Theory*, 24(1) :76–80, Janv. 1978.
- [45] David J.C. MacKay. Good error-correcting codes based on very sparse matrices. *IEEE Transactions on Information Theory*, Mars 1999.
- [46] J. Xu et al. Construction of regular and irregular ldpc codes : Geometry decomposition and masking. *IEEE Transactions on Information Theory*, 1(53) :121–134, Janv. 2007.
- [47] M.G. Luby et al. Improved low-density parity-check codes using irregular graphs. *IEEE Transactions on Information Theory*, 47(2) :585–598, Fév. 2001.

- [48] C.Tao Tian, R. Jones, J. Villasenor, and R. Wesel. Selective avoidance of cycles in irregular ldpc code construction. *IEEE Transactions on Communications*, 52(8) :1242–1247, Août 2004.
- [49] Yu Kou, Shu Lin, and Marc P. C. Fossorier. Low density parity check codes based on finite geometries : A rediscovery and new results. *IEEE Transactions on Information Theory*, 47 :2711–2736, 2001.
- [50] B. Amar et al. Construction of low-density parity-check codes based on balanced incomplete block designs. *IEEE Transactions on Information Theory*, 50(6) :1257–1268, 2004.
- [51] I. Djurdjevic, X. Jun, K. Abdel-Ghaffar, and L. Shu. A class of low-density parity-check codes constructed based on reed-solomon codes with two information symbols. *IEEE Communications Letters*, 7(7) :317–319, Juillet 2003.
- [52] H. Zhang and J.M.F. Moura. Large-girth ldpc codes based on graphical models. *4th IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2003*, pages 100–104, Juin 2003.
- [53] M.O’Sullivan. Algebraic construction of sparse matrices with large girth. *IEEE Transactions on Information Theory*, 52(2) :718–727, Fév. 2006.
- [54] David J.C. MacKay and I. Background. Low density parity check codes over $gf(q)$. *IEEE Communications Letters*, 2 :70–71, 1996.
- [55] L. Sun and V. Kumar. Field programmable gate array implementation of a generalized decoder for structured low-density parity check codes. *IEEE International Conference on Field-Programmable Technology*, pages 17–24, 2004.
- [56] M. Arabaci and I. Djordjevic. An alternative fpga implementation of decoders for quasi-cyclic ldpc codes. *16th Telecommunications Forum (TELFOR) 2008*, Nov.

- 2008.
- [57] J. Chen and M.P.C. Fossorier. Near optimum universal belief propagation based decoding of low density parity check codes. *IEEE Transactions on Communications*, 50(3) :406–414, Mars 2002.
- [58] J. Boutros, O. Pothier, and G. Zemor. Generalized low density (tanner) codes. *IEEE International Conference on Communications, 1999. ICC '99*, 1 :441–445, 1999.
- [59] D. Chiaroni, M. Le Pallec, C. Simonneau, S. Borne, D. Bayart, and JP. Hamaide. Cascadability issues in a multi-service optical packet ring network. *Photonics in Switching 2007*, pages 25–26, 2007.
- [60] G.P. Agrawal. *Fiber-Optics Communication Systems*. Wiley-Interscience, 1992. ISBN : 0471542865.
- [61] Y. Zouine. *Contribution par la simulation système à l'étude des contraintes des composants optoelectroniques sur la transmission optique utilisant la technique CDMA*. Thèse de doctorat, Université de Limoges, France, 2005.
- [62] J. Gowar. *Optical Communication Systems - Second Edition*. Prentice Hall International Series In Optoelectronics, 1993. ISBN : 0136387276.
- [63] John G. Proakis. *Digital communications 3rd ed.* McGraw-Hill, 1995. ISBN : 0070517266.
- [64] P.A. Humblet and M. Azizoglu. On the bit error rate of lightwave systems with optical amplifiers. *IEEE Journal of Lightwave Technology*, 9(11) :1576–1582, Nov. 1991.
- [65] S. Sahuguede, D. Fafchamps, A. Julien-Vergonjanne, G. Rodriguez, J.P. Cances, P. Gallion, and J.M. Dumas. Performances du décodage ldpc sur un anneau optique

- fonctionnant en mode paquet. *Journées Nationales d'Optique Guidée 2008, JNOG 2008*, Oct. 2008.
- [66] S. Sahuguede, D. Fafchamps, A. Julien-Vergonjanne, G. Rodriguez, J.P. Cances, and P. Gallion. Ldpc code design and performance analysis on ook chi-square based optical channels. *Accepté en 2009 pour publication dans IEEE Photonics Technology Letters*.
- [67] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Adaptation du décodage itératif des codes ldpc sur canal optique utilisant un modèle à statistique chi-2. *colloque GRETSI 2009*, Sept. 2009.
- [68] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Soft decision ldpc decoding over chi-square based optical channels. *IEEE Journal of Lightwave Technology - Special issue on trends in signal processing for high capacity lightwave systems*, 27(16) :3540 – 3545, Août 2009.
- [69] P.R. Prucnal (Editor). *Optical Code Division Multiple Access : Fundamentals and Applications*. CRC ; Har/Cdr edition, 2006. ISBN : 084933683X.
- [70] J.A. Salehi and C.A. Brackett. Code division multiple-access techniques in optical fiber networks. *IEEE Transactions on Communications*, 37(8) :824–833, Août 1989.
- [71] S.T. Nasaruddin and T. Tetsuo. Design of reconfigurable multiweight wavelength-time optical codes for secure multimedia optical cdma networks. *Proc. of International Conference on Communications (ICC)*, pages 5437–5442, Mai 2008.
- [72] N.G. Tarhuni, T.O. Korhonen, E. Mutafungwa, and M.S. Elmusrati. Multiclass optical orthogonal codes for multiservice optical cdma networks. *IEEE Journal of Lightwave Technology*, 24(2) :694–704, Février. 2006.

- [73] W.C. Kwong and G-C. Yang. Design of a multilength optical orthogonal codes for optical cdma multimedia networks. *IEEE Transactions on Communications*, 50(8) :1258–1265, Août 2002.
- [74] G.C. Yang and W.C. Kwong. Performance comparison of multiwavelength cdma and wdma+cdma for fiber-optic networks. *IEEE Transactions on Communications*, 45(11) :1426–1434, Nov. 1997.
- [75] S.S. Lee and S.W. Seo. New construction of multiwavelength optical orthogonal codes. *IEEE Transactions on Communications*, 50(12) :2003–2008, Déc. 2002.
- [76] M. Morelle. *Codage en 2 Dimensions pour les systèmes de communications Optiques CDMA (OCDMA) - Application aux transmissions multimédia*. Thèse de doctorat, Université de Limoges, France, 2008.
- [77] T.Demeechai and A.B.Sharma. Beat noise in a non-coherent optical cdma system. *IEEE International Conference on Communication Systems (ICCS) 2002*, 2 :899–902, Nov. 2002.
- [78] X.Wang and K. Kitayama. Beat noise in a non-coherent optical cdma system. *Journal of Lightwave Technology*, 22(10) :2226–2235, Oct. 2004.
- [79] T.M. Bazan, D.Harle, and I. Andonovic. Performance analysis of 2-d time-wavelength ocdma systems with coherent light sources : Code design considerations. *Journal of Lightwave Technology*, 24(10), Oct. 2006.
- [80] Tao Pu, Hanyi Zhang, Yili Guo, Ming Xu, and Yuquan Li. Evaluation of beat noise in ocdma system with non-gaussian approximated method. *Journal of Lightwave Technology*, 24(10) :3574–3582, Oct. 2006.
- [81] B. Ni, J.S. Lehnert, and J. Zhang. Impact of beat noise on an incoherent ocdma system with temporal spreading. *IEEE Signals Systems and Computers*, 1 :103–106,

- Nov. 2004.
- [82] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Performance bound of worst beat noise interference scenario in 2-d incoherent ocdma systems. *International Conference on Software, Telecommunications and Computer Networks 2007, SoftCOM 2007*, pages 1–5, Sept. 2007.
- [83] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Error probability bound considering beat noise in 2-d ocdma systems. *Journal of Communications Software and Systems, JCOMSS*, 3(4) :266–274, Déc. 2007.
- [84] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Ocdma channel statistic as a mixture of chi-square distributions. *Advanced International Conference on Telecommunications, AICT 2009*, Mai 2009.
- [85] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Beat noise in optical cdma channels based on a mixture of chi-2 statistics. *International Conference on Photonics in Switching, PS 2009*, Sept. 2009.
- [86] Tancevski and A.Rush. Impact of the beat noise on the performance of 2-d optical cdma systems. *IEEE Communication Letters*, 4(8) :264–266, Août 2000.
- [87] G.C. Yang and W.C. Kwong. *Prime Codes with Applications to CDMA Optical and Wireless Networks*. Artech House, 2002. ISBN : 1580530737.
- [88] S.M. Johnson. A new upper bound for error-correcting codes. *IRE Transaction on Information Theory*, 8(3) :203–207, Avril 1962.
- [89] C. Goursaud-Brugeaud. *Réception Multi-utilisateurs par annulation parallèle dans les systèmes CDMA optiques*. Thèse de doctorat, Université de Limoges, France, 2006.

- [90] M. Morelle, S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Theoretical performance of multi-weight spreading codes for multimedia optical access network. *IEEE International Conference on Communications 2009, ICC 09*, Juin 2009.
- [91] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, ninth dover printing, tenth gpo printing edition, 1964.
- [92] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Performance of optical cdma systems with low density parity check codes. *International Conference on Systems and Networks Communications 2007, ICSNC 2007*, page 57, Août 2007.
- [93] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Beat noise in optical cdma channels based on a mixture of chi-2 statistics. *Accepté en 2009 pour publication dans European Transactions on Telecommunications (ETT)*.
- [94] S. Sahuguede, M. Morelle, A. Julien-Vergonjanne, and J.P. Cances. Multi-rate and qos differentiation in optical code division multiple access networks. *IEEE Consumer Communications and Networking Conference 2008, CCNC 2008*, pages 581 – 585, Janv. 2008.
- [95] S. Sahuguede, A. Julien-Vergonjanne, and J.P. Cances. Adapted ldpc error correction scheme for 2d optical cdma multimedia system. *International Symposium on Spread Spectrum Techniques and Applications 2008, ISSSTA '08*, pages 775–779, Août 2008.
- [96] D. Fafchamps, G. Rodriguez-Guisantes, and P. Gallion. Chi-square statistical models as a good base for the optimisation of optical communication systems. *International Conference on Photonics in Switching, PS 2008*, pages 1–2, Août 2008.

Annexe A

Caractéristiques du code RS(255,239) et du corps GF(2^8)

A.1 Propriétés du code RS(255,239)

Le code RS(255,239) est de longueur $N = 255 = 2^8 - 1$ donc $m = 8$, ce qui signifie que les symboles sont de 8 bits. Il est ainsi construit à partir du corps GF(2^8). De plus, il y a donc $N - K = 16$ symboles de redondance d'où $t = \frac{N-K}{2} = 8$ symboles corrigibles par mots de N bits. Le pouvoir de correction correspond donc à 64 bits au maximum puisque chaque symbole est sur 8 bits. Il peut corriger environ jusqu'à 3% des bits transmis.

Le rendement de ce code est $R = 0,937$, son insertion dans un canal de transmission entraîne donc une légère diminution du débit utile, qui se retrouve multiplié par R .

Le paragraphe suivant, donne les caractéristiques du corps de Galois GF(2^8) à partir duquel le polynôme générateur du code est obtenu, et sur lequel toutes les opérations au niveau du codage et du décodage sont réalisées.

A.2 Propriétés du corps GF(2⁸)

Sur GF(2⁸), il existe 2⁸ = 256 symboles composés de 8 bits comme défini par 1.10 :

$$GF(2^8) = \{0, \alpha^0, \alpha, \dots, \alpha^{254}\}$$

Afin de réaliser les opérations nécessaires au codage / décodage, il faut tout d'abord construire chacun de ces éléments. Les huit premières puissances de α sont facilement obtenues, avec un '1' sur la i^{me} position pour l'élément α^i , $i \in [0, 7]$, c'est-à-dire :

$$\alpha^0 : 10000000$$

$$\alpha^1 : 01000000$$

$$\alpha^2 : 00100000$$

$$\alpha^3 : 00010000$$

$$\alpha^4 : 00001000$$

$$\alpha^5 : 00000100$$

$$\alpha^6 : 00000010$$

$$\alpha^7 : 00000001$$

Pour obtenir les puissances supérieures, on part du polynôme primitif de GF(256) qui est donné par (voir tableau 1.1) :

$$f(X) = 1 + X^2 + X^3 + X^4 + X^8 \tag{A.1}$$

Comme α est par définition une racine du polynôme primitif, c'est-à-dire $f(\alpha) = 0$, on a alors la relation :

$$\alpha^8 = 1 + \alpha^2 + \alpha^3 + \alpha^4 \tag{A.2}$$

A.3 Codage et polynôme générateur du RS(255,239)

Les α^i pour $i \in [0, 8]$ étant alors déterminés, tous les α^i pour $i \in [9, 254]$ peuvent être obtenus à partir de la règle de multiplication :

$$\alpha^{i+1} = \alpha \cdot \alpha^i \tag{A.3}$$

En utilisant également la règle d'addition dans $\text{GF}(2^8)$ de deux α^i différents, qui revient à faire l'addition bit à bit sans retenue.

Par exemple, cela donne pour les premières puissance de α^i , $i \geq 9$:

$$\begin{aligned} \alpha^9 &= \alpha \cdot \alpha^8 = \alpha + \alpha^3 + \alpha^4 + \alpha^5 = && 01011100 \\ \alpha^{10} &= \alpha \cdot \alpha^9 = \alpha^2 + \alpha^4 + \alpha^5 + \alpha^6 = && 00101110 \\ \alpha^{11} &= \alpha \cdot \alpha^{10} = \alpha^3 + \alpha^5 + \alpha^6 + \alpha^7 = && 00010111 \\ \alpha^{12} &= \alpha \cdot \alpha^{11} = \alpha^4 + \alpha^6 + \alpha^7 + \alpha^8 = 00001011 + 10111000 = && 10110011 \\ & \dots && \dots \end{aligned}$$

Avec cette méthode, on obtient alors chacun des 255 α^i qui sont représentés sur les pages suivantes figures A.1 et A.2.

A.3 Codage et polynôme générateur du RS(255,239)

Le polynôme générateur est de degré $N - K = 255 - 239 = 16$. Il possède ainsi 16 racines, qui sont des puissances de α consécutives. Il peut alors prendre l'expression suivante :

$$G(X) = (X + \alpha) \cdot (X + \alpha^2) \cdot (X + \alpha^3) \cdot \dots \cdot (X + \alpha^{16}) \tag{A.4}$$

Une fois le polynôme développé et les α^i additionnés et multipliés selon les règles présentées plus haut, le polynôme générateur s'écrit :

$$\begin{aligned} G(X) = & \alpha^{136} \cdot X^0 + \alpha^{240} \cdot X^1 + \alpha^{208} \cdot X^2 + \alpha^{195} \cdot X^3 + \alpha^{181} \cdot X^4 + \alpha^{158} \cdot X^5 + \alpha^{201} \cdot X^6 \\ & + \alpha^{100} \cdot X^7 + \alpha^{11} \cdot X^8 + \alpha^{83} \cdot X^9 + \alpha^{167} \cdot X^{10} + \alpha^{107} \cdot X^{11} + \alpha^{113} \cdot X^{12} + \alpha^{110} \cdot X^{13} \\ & + \alpha^{106} \cdot X^{14} + \alpha^{121} \cdot X^{15} + X^{16} \end{aligned} \tag{A.5}$$

Le codage d'un bloc de bits s'effectue en multipliant une séquence de 239 symboles par le polynôme $G(X)$. Le mot obtenu est alors de 255 symboles soit $255 \times 8 = 2040$ bits.

A.3 Codage et polynôme générateur du RS(255,239)

Alpha 0 : 1 0 0 0 0 0 0 0	Alpha 64 : 1 1 1 1 1 0 1 0
Alpha 1 : 0 1 0 0 0 0 0 0	Alpha 65 : 0 1 1 1 1 1 0 1
Alpha 2 : 0 0 1 0 0 0 0 0	Alpha 66 : 1 0 0 0 0 1 1 0
Alpha 3 : 0 0 0 1 0 0 0 0	Alpha 67 : 0 1 0 0 0 0 1 1
Alpha 4 : 0 0 0 0 1 0 0 0	Alpha 68 : 1 0 0 1 1 0 0 1
Alpha 5 : 0 0 0 0 0 1 0 0	Alpha 69 : 1 1 1 1 0 1 0 0
Alpha 6 : 0 0 0 0 0 0 1 0	Alpha 70 : 0 1 1 1 1 0 1 0
Alpha 7 : 0 0 0 0 0 0 0 1	Alpha 71 : 0 0 1 1 1 1 0 1
Alpha 8 : 1 0 1 1 1 0 0 0	Alpha 72 : 1 0 1 0 0 1 1 0
Alpha 9 : 0 1 0 1 1 1 0 0	Alpha 73 : 0 1 0 1 0 0 1 1
Alpha 10 : 0 0 1 0 1 1 1 0	Alpha 74 : 1 0 0 1 0 0 0 1
Alpha 11 : 0 0 0 1 0 1 1 1	Alpha 75 : 1 1 1 1 0 0 0 0
Alpha 12 : 1 0 1 1 0 0 1 1	Alpha 76 : 0 1 1 1 1 0 0 0
Alpha 13 : 1 1 1 0 0 0 0 1	Alpha 77 : 0 0 1 1 1 1 0 0
Alpha 14 : 1 1 0 0 1 0 0 0	Alpha 78 : 0 0 0 1 1 1 1 0
Alpha 15 : 0 1 1 0 0 1 0 0	Alpha 79 : 0 0 0 0 1 1 1 1
Alpha 16 : 0 0 1 1 0 0 1 0	Alpha 80 : 1 0 1 1 1 1 1 1
Alpha 17 : 0 0 0 1 1 0 0 1	Alpha 81 : 1 1 1 0 0 1 1 1
Alpha 18 : 1 0 1 1 0 1 0 0	Alpha 82 : 1 1 0 0 1 0 1 1
Alpha 19 : 0 1 0 1 1 0 1 0	Alpha 83 : 1 1 0 1 1 1 0 1
Alpha 20 : 0 0 1 0 1 1 0 1	Alpha 84 : 1 1 0 1 0 1 1 0
Alpha 21 : 1 0 1 0 1 1 1 0	Alpha 85 : 0 1 1 0 1 0 1 1
Alpha 22 : 0 1 0 1 0 1 1 1	Alpha 86 : 1 0 0 0 1 1 0 1
Alpha 23 : 1 0 0 1 0 0 1 1	Alpha 87 : 1 1 1 1 1 1 1 0
Alpha 24 : 1 1 1 1 0 0 0 1	Alpha 88 : 0 1 1 1 1 1 1 1
Alpha 25 : 1 1 0 0 0 0 0 0	Alpha 89 : 1 0 0 0 0 1 1 1
Alpha 26 : 0 1 1 0 0 0 0 0	Alpha 90 : 1 1 1 1 1 0 1 1
Alpha 27 : 0 0 1 1 0 0 0 0	Alpha 91 : 1 1 0 0 0 1 0 1
Alpha 28 : 0 0 0 1 1 0 0 0	Alpha 92 : 1 1 0 1 1 0 1 0
Alpha 29 : 0 0 0 0 1 1 0 0	Alpha 93 : 0 1 1 0 1 1 0 1
Alpha 30 : 0 0 0 0 0 1 1 0	Alpha 94 : 1 0 0 0 1 1 1 0
Alpha 31 : 0 0 0 0 0 0 1 1	Alpha 95 : 0 1 0 0 0 1 1 1
Alpha 32 : 1 0 1 1 1 0 0 1	Alpha 96 : 1 0 0 1 1 0 1 1
Alpha 33 : 1 1 1 0 0 1 0 0	Alpha 97 : 1 1 1 1 0 1 0 1
Alpha 34 : 0 1 1 1 0 0 1 0	Alpha 98 : 1 1 0 0 0 0 1 0
Alpha 35 : 0 0 1 1 1 0 0 1	Alpha 99 : 0 1 1 0 0 0 0 1
Alpha 36 : 1 0 1 0 0 1 0 0	Alpha 100 : 1 0 0 0 1 0 0 0
Alpha 37 : 0 1 0 1 0 0 1 0	Alpha 101 : 0 1 0 0 0 1 0 0
Alpha 38 : 0 0 1 0 1 0 0 1	Alpha 102 : 0 0 1 0 0 0 1 0
Alpha 39 : 1 0 1 0 1 1 0 0	Alpha 103 : 0 0 0 1 0 0 0 1
Alpha 40 : 0 1 0 1 0 1 1 0	Alpha 104 : 1 0 1 1 0 0 0 0
Alpha 41 : 0 0 1 0 1 0 1 1	Alpha 105 : 0 1 0 1 1 0 0 0
Alpha 42 : 1 0 1 0 1 1 0 1	Alpha 106 : 0 0 1 0 1 1 0 0
Alpha 43 : 1 1 1 0 1 1 1 0	Alpha 107 : 0 0 0 1 0 1 1 0
Alpha 44 : 0 1 1 1 0 1 1 1	Alpha 108 : 0 0 0 0 1 0 1 1
Alpha 45 : 1 0 0 0 0 0 1 1	Alpha 109 : 1 0 1 1 1 1 0 1
Alpha 46 : 1 1 1 1 1 0 0 1	Alpha 110 : 1 1 1 0 0 1 1 0
Alpha 47 : 1 1 0 0 0 1 0 0	Alpha 111 : 0 1 1 1 0 0 1 1
Alpha 48 : 0 1 1 0 0 0 1 0	Alpha 112 : 1 0 0 0 0 0 0 1
Alpha 49 : 0 0 1 1 0 0 0 1	Alpha 113 : 1 1 1 1 1 0 0 0
Alpha 50 : 1 0 1 0 0 0 0 0	Alpha 114 : 0 1 1 1 1 1 0 0
Alpha 51 : 0 1 0 1 0 0 0 0	Alpha 115 : 0 0 1 1 1 1 1 0
Alpha 52 : 0 0 1 0 1 0 0 0	Alpha 116 : 0 0 0 1 1 1 1 1
Alpha 53 : 0 0 0 1 0 1 0 0	Alpha 117 : 1 0 1 1 0 1 1 1
Alpha 54 : 0 0 0 0 1 0 1 0	Alpha 118 : 1 1 1 0 0 0 1 1
Alpha 55 : 0 0 0 0 0 1 0 1	Alpha 119 : 1 1 0 0 1 0 0 1
Alpha 56 : 1 0 1 1 1 0 1 0	Alpha 120 : 1 1 0 1 1 1 0 0
Alpha 57 : 0 1 0 1 1 1 0 1	Alpha 121 : 0 1 1 0 1 1 1 0
Alpha 58 : 1 0 0 1 0 1 1 0	Alpha 122 : 0 0 1 1 0 1 1 1
Alpha 59 : 0 1 0 0 1 0 1 1	Alpha 123 : 1 0 1 0 0 0 1 1
Alpha 60 : 1 0 0 1 1 1 0 1	Alpha 124 : 1 1 1 0 1 0 0 1
Alpha 61 : 1 1 1 1 0 1 1 0	Alpha 125 : 1 1 0 0 1 1 0 0
Alpha 62 : 0 1 1 1 1 0 1 1	Alpha 126 : 0 1 1 0 0 1 1 0
Alpha 63 : 1 0 0 0 0 1 0 1	Alpha 127 : 0 0 1 1 0 0 1 1

FIG. A.1 – Puissances de α de 0 à 127

Alpha 128 : 1 0 1 0 0 0 0 1	Alpha 192 : 0 1 0 0 0 0 0 1
Alpha 129 : 1 1 1 0 1 0 0 0	Alpha 193 : 1 0 0 1 1 0 0 0
Alpha 130 : 0 1 1 1 0 1 0 0	Alpha 194 : 0 1 0 0 1 1 0 0
Alpha 131 : 0 0 1 1 1 0 1 0	Alpha 195 : 0 0 1 0 0 1 1 0
Alpha 132 : 0 0 0 1 1 1 0 1	Alpha 196 : 0 0 0 1 0 0 1 1
Alpha 133 : 1 0 1 1 0 1 1 0	Alpha 197 : 1 0 1 1 0 0 0 1
Alpha 134 : 0 1 0 1 1 0 1 1	Alpha 198 : 1 1 1 0 0 0 0 0
Alpha 135 : 1 0 0 1 0 1 0 1	Alpha 199 : 0 1 1 1 0 0 0 0
Alpha 136 : 1 1 1 1 0 0 1 0	Alpha 200 : 0 0 1 1 1 0 0 0
Alpha 137 : 0 1 1 1 1 0 0 1	Alpha 201 : 0 0 0 1 1 1 0 0
Alpha 138 : 1 0 0 0 0 1 0 0	Alpha 202 : 0 0 0 0 1 1 1 0
Alpha 139 : 0 1 0 0 0 0 1 0	Alpha 203 : 0 0 0 0 0 1 1 1
Alpha 140 : 0 0 1 0 0 0 0 1	Alpha 204 : 1 0 1 1 1 0 1 1
Alpha 141 : 1 0 1 0 1 0 0 0	Alpha 205 : 1 1 1 0 0 1 0 1
Alpha 142 : 0 1 0 1 0 1 0 0	Alpha 206 : 1 1 0 0 1 0 1 0
Alpha 143 : 0 0 1 0 1 0 1 0	Alpha 207 : 0 1 1 0 0 1 0 1
Alpha 144 : 0 0 0 1 0 1 0 1	Alpha 208 : 1 0 0 0 1 0 1 0
Alpha 145 : 1 0 1 1 0 0 1 0	Alpha 209 : 0 1 0 0 0 1 0 1
Alpha 146 : 0 1 0 1 1 0 0 1	Alpha 210 : 1 0 0 1 1 0 1 0
Alpha 147 : 1 0 0 1 0 1 0 0	Alpha 211 : 0 1 0 0 1 1 0 1
Alpha 148 : 0 1 0 0 1 0 1 0	Alpha 212 : 1 0 0 1 1 1 1 0
Alpha 149 : 0 0 1 0 0 1 0 1	Alpha 213 : 0 1 0 0 1 1 1 1
Alpha 150 : 1 0 1 0 1 0 1 0	Alpha 214 : 1 0 0 1 1 1 1 1
Alpha 151 : 0 1 0 1 0 1 0 1	Alpha 215 : 1 1 1 1 0 1 1 1
Alpha 152 : 1 0 0 1 0 0 1 0	Alpha 216 : 1 1 0 0 0 0 1 1
Alpha 153 : 0 1 0 0 1 0 0 1	Alpha 217 : 1 1 0 1 1 0 0 1
Alpha 154 : 1 0 0 1 1 1 0 0	Alpha 218 : 1 1 0 1 0 1 0 0
Alpha 155 : 0 1 0 0 1 1 1 0	Alpha 219 : 0 1 1 0 1 0 1 0
Alpha 156 : 0 0 1 0 0 1 1 1	Alpha 220 : 0 0 1 1 0 1 0 1
Alpha 157 : 1 0 1 0 1 0 1 1	Alpha 221 : 1 0 1 0 0 0 1 0
Alpha 158 : 1 1 1 0 1 1 0 1	Alpha 222 : 0 1 0 1 0 0 0 1
Alpha 159 : 1 1 0 0 1 1 1 0	Alpha 223 : 1 0 0 1 0 0 0 0
Alpha 160 : 0 1 1 0 0 1 1 1	Alpha 224 : 0 1 0 0 1 0 0 0
Alpha 161 : 1 0 0 0 1 0 1 1	Alpha 225 : 0 0 1 0 0 1 0 0
Alpha 162 : 1 1 1 1 1 1 0 1	Alpha 226 : 0 0 0 1 0 0 1 0
Alpha 163 : 1 1 0 0 0 1 1 0	Alpha 227 : 0 0 0 0 1 0 0 1
Alpha 164 : 0 1 1 0 0 0 1 1	Alpha 228 : 1 0 1 1 1 1 0 0
Alpha 165 : 1 0 0 0 1 0 0 1	Alpha 229 : 0 1 0 1 1 1 1 0
Alpha 166 : 1 1 1 1 1 1 0 0	Alpha 230 : 0 0 1 0 1 1 1 1
Alpha 167 : 0 1 1 1 1 1 1 0	Alpha 231 : 1 0 1 0 1 1 1 1
Alpha 168 : 0 0 1 1 1 1 1 1	Alpha 232 : 1 1 1 0 1 1 1 1
Alpha 169 : 1 0 1 0 0 1 1 1	Alpha 233 : 1 1 0 0 1 1 1 1
Alpha 170 : 1 1 1 0 1 0 1 1	Alpha 234 : 1 1 0 1 1 1 1 1
Alpha 171 : 1 1 0 0 1 1 0 1	Alpha 235 : 1 1 0 1 0 1 1 1
Alpha 172 : 1 1 0 1 1 1 1 0	Alpha 236 : 1 1 0 1 0 0 1 1
Alpha 173 : 0 1 1 0 1 1 1 1	Alpha 237 : 1 1 0 1 0 0 0 1
Alpha 174 : 1 0 0 0 1 1 1 1	Alpha 238 : 1 1 0 1 0 0 0 0
Alpha 175 : 1 1 1 1 1 1 1 1	Alpha 239 : 0 1 1 0 1 0 0 0
Alpha 176 : 1 1 0 0 0 1 1 1	Alpha 240 : 0 0 1 1 0 1 0 0
Alpha 177 : 1 1 0 1 1 0 1 1	Alpha 241 : 0 0 0 1 1 0 1 0
Alpha 178 : 1 1 0 1 0 1 0 1	Alpha 242 : 0 0 0 0 1 1 0 1
Alpha 179 : 1 1 0 1 0 0 1 0	Alpha 243 : 1 0 1 1 1 1 1 0
Alpha 180 : 0 1 1 0 1 0 0 1	Alpha 244 : 0 1 0 1 1 1 1 1
Alpha 181 : 1 0 0 0 1 1 0 0	Alpha 245 : 1 0 0 1 0 1 1 1
Alpha 182 : 0 1 0 0 0 1 1 0	Alpha 246 : 1 1 1 1 0 0 1 1
Alpha 183 : 0 0 1 0 0 0 1 1	Alpha 247 : 1 1 0 0 0 0 0 1
Alpha 184 : 1 0 1 0 1 0 0 1	Alpha 248 : 1 1 0 1 1 0 0 0
Alpha 185 : 1 1 1 0 1 1 0 0	Alpha 249 : 0 1 1 0 1 1 0 0
Alpha 186 : 0 1 1 1 0 1 1 0	Alpha 250 : 0 0 1 1 0 1 1 0
Alpha 187 : 0 0 1 1 1 0 1 1	Alpha 251 : 0 0 0 1 1 0 1 1
Alpha 188 : 1 0 1 0 0 1 0 1	Alpha 252 : 1 0 1 1 0 1 0 1
Alpha 189 : 1 1 1 0 1 0 1 0	Alpha 253 : 1 1 1 0 0 0 1 0
Alpha 190 : 0 1 1 1 0 1 0 1	Alpha 254 : 0 1 1 1 0 0 0 1
Alpha 191 : 1 0 0 0 0 0 1 0	

FIG. A.2 – Puissances de α de 128 à 254

Annexe B

Algorithmes de décodage des codes RS et BCH

B.1 Algorithme d'Euclide

Le principe de l'algorithme d'Euclide [14] est de réaliser une série de divisions de polynômes et de déterminer à chaque itération les restes et quotients de ces opérations. A chaque itération, on définit Q_i et R_i , quotients et restes de la division, ainsi qu'un polynôme auxiliaire U_i .

La première étape de cet algorithme consiste en l'initialisation des différents polynômes :

$$\left\{ \begin{array}{l} R_{-1}(X) = X^{2t} \\ R_0(X) = S(X) \\ U_{-1}(X) = 0 \\ U_0(X) = 1 \end{array} \right. \quad (\text{B.1})$$

Ensuite à chaque itération on détermine les quantités $Q_i(X)$, $R_{i+1}(X)$ et $U_{i+1}(X)$ en

effectuant la division de $R_{i-1}(X)$ par $R_i(X)$:

$$\frac{R_{i-1}(X)}{R_i(X)} = Q_i(X) + \frac{R_{i+1}(X)}{R_i(X)} \quad (\text{B.2})$$

et pour $U_{i+1}(X)$:

$$U_{i+1}(X) = Q_i(X) \cdot U_i(X) + U_{i-1}(X) \quad (\text{B.3})$$

Lorsque le degré de $U_{i+1}(X)$ est inférieur ou égal à t et le degré de $R_{i+1}(X)$ est strictement inférieur à t on détermine alors les polynômes recherchés par :

$$\begin{cases} \sigma(X) = U_{i+1}(X) \\ \omega(X) = R_{i+1}(X) \end{cases} \quad (\text{B.4})$$

B.2 Algorithme de Berlekamp-Massey

Les polynômes $\sigma(X)$ et $\omega(X)$ sont obtenus de manière itérative avec l'algorithme de Berlekamp-Massey [12, 13]. A l'itération i , ils sont notés $\sigma_i(X)$ et $\omega_i(X)$. De plus, à chaque itération, on se servira de polynômes auxiliaires notés $\widetilde{\sigma}_i(X)$ et $\widetilde{\omega}_i(X)$ ainsi que d'un paramètre noté d_i .

Tout d'abord, d_i ainsi que les différents polynômes sont initialisés comme suit :

$$\begin{cases} \sigma_0(X) = 1 \\ \omega_0(X) = 0 \\ \widetilde{\sigma}_i(X) = 0 \\ \widetilde{\omega}_i(X) = 1 \\ d_0 = 0 \end{cases} \quad (\text{B.5})$$

Les itérations ont alors lieu pour i allant de 0 à $2t$. A chaque tour, cela consiste à :

B.2 Algorithme de Berlekamp-Massey

- Calculer la valeur Δ_i , coefficient de X_i dans le polynôme $\sigma_i(X) \cdot S(X)$.
- Calculer $\sigma_{i+1}(X)$ et $\omega_{i+1}(X)$ avec :

$$\begin{cases} \sigma_{i+1}(X) = \sigma_i(X) + \Delta_i \cdot \widetilde{\sigma_i(X)} \\ \omega_{i+1}(X) = \omega_i(X) + \Delta_i \cdot \widetilde{\omega_i(X)} \end{cases} \quad (\text{B.6})$$

- Faire un test sur Δ_i et d_i :
- si $\Delta_i = 0$ ou $2 \cdot d_i > i$ alors :

$$\begin{cases} \widetilde{\sigma_{i+1}(X)} = X \cdot \widetilde{\sigma_i(X)} \\ \widetilde{\omega_{i+1}(X)} = X \cdot \widetilde{\omega_i(X)} \\ d_{i+1} = d_i \end{cases} \quad (\text{B.7})$$

- dans les autres cas :

$$\begin{cases} \widetilde{\sigma_{i+1}(X)} = X \cdot \Delta_i^{-1} \cdot \sigma_i(X) \\ \widetilde{\omega_{i+1}(X)} = X \cdot \Delta_i^{-1} \cdot \omega_i(X) \\ d_{i+1} = i + 1 - d_i \end{cases} \quad (\text{B.8})$$

Quand $i = 2t$, les polynômes recherchés sont donnés par :

$$\begin{cases} \sigma(X) = \sigma_{i+1}(X) \\ \omega(X) = \omega_{i+1}(X) \end{cases} \quad (\text{B.9})$$

Annexe C

Méthodes de construction des codes LDPC aléatoires

C.1 Construction de Gallager : régulier aléatoire

La méthode énoncée par Gallager [2] nécessite une contrainte définissant une relation entre le nombre de lignes de H ($N - K$) et le poids des colonnes w_c ainsi qu'entre le nombre de colonnes (N) et le poids des lignes w_l :

$$\begin{cases} N - K = \alpha_1 w_c \\ N = \alpha_2 w_l \end{cases} \quad (\text{C.1})$$

Avec $\alpha_1, \alpha_2 \in \mathbb{N}$.

Pour contruire H il faut alors :

- Déterminer un couple (N, w_l) qui satisfait (C.1) et de sorte que N soit le plus proche de N_s . Pour cela on fait varier w_l (entre 3 et 10 par exemple), et on détermine α_2

et N avec :

$$\begin{cases} \alpha_2 = \left\lceil \frac{N_s}{w_l} \right\rceil \\ N = \alpha_2 w_l = \left\lceil \frac{N_s}{w_l} \right\rceil w_l \end{cases} \quad (\text{C.2})$$

On choisit ensuite le couple de valeurs qui permet de minimiser $|N - N_s|$.

- Déterminer K et w_c de manière à satisfaire (C.1) et tels que $|K - K_s|$ soit minimal. Pour cela, comme précédemment on fait varier w_c dans un certain intervalle et on détermine α_1 et K avec :

$$\begin{cases} \alpha_1 = \left\lceil \frac{N - K_s}{w_c} \right\rceil \\ K = N - \alpha_1 w_c = N - \left\lceil \frac{N - K_s}{w_c} \right\rceil w_c \end{cases} \quad (\text{C.3})$$

- Une matrice vide de dimensions $(N - K) \times N$ est alors créée. Elle est divisée en w_c sous-matrices de dimension $\alpha_1 \times N$ qui posséderont à la fin de la construction chacune un élément non nul par colonne.
- La première de ces sous-matrices est contruite en plaçant des '1' sur les positions (i, j) avec $i \in [0, \alpha_1 - 1]$ et pour chaque $i, j \in [i w_l, (i + 1) w_l - 1]$
- Chacune des autres sous-matrices est obtenue par permutation aléatoire des colonnes de la première sous-matrice.

C.2 Construction de MacKay : régulier aléatoire

Contrairement à la méthode de Gallager, la méthode proposée par Mackay [45] permet de s'affranchir de la condition (C.1) et d'avoir ainsi plus de souplesse sur les dimensions du code.

Cependant, pour satisfaire (1.33) et pour pouvoir choisir un poids w_c quelconque, on

impose la condition suivante :

$$\frac{N}{N-K} \in \mathbb{N} \quad (\text{C.4})$$

Pour construire H il faut alors :

- Déterminer la dimension K du code la plus proche de K_s respectant (C.4). Pour cela on définit β tel que :

$$\beta = \left\lceil \frac{N_s}{N_s - K} \right\rceil \quad (\text{C.5})$$

Puis on détermine K tel que $\beta = \frac{N}{N-K}$ c'est-à-dire :

$$K = N \times \frac{\beta - 1}{\beta} = N \times \frac{\left\lceil \frac{N_s}{N_s - K} \right\rceil - 1}{\left\lceil \frac{N_s}{N_s - K} \right\rceil} \quad (\text{C.6})$$

- Une fois les paramètres (N, K) ainsi que le poids des colonnes w_c fixés :
- Fixer le poids des lignes w_l avec

$$w_l = w_c \times \frac{N}{N-K} = w_c \times \beta \quad (\text{C.7})$$

- Construire une matrice vide de dimensions $(N-K) \times N$
- Pour chaque colonne j de 0 à $N-1$, choisir aléatoirement w_c positions différentes pour y placer un '1'
- Parcourir toutes les lignes i de 0 à $N-K-1$ pour uniformiser le poids des lignes w_l . Si une ligne i possède plus de w_l '1', on choisit un '1' aléatoirement ($\text{position}(i, j)$) et on le déplace en position (i', j) où la ligne i' contient moins de w_l '1'. Si aucune ligne i' ne correspond à ce critère, on choisit un '1' sur une autre colonne j' et on réitère l'opération jusqu'à ce que la ligne i possède exactement w_l '1'.

C.3 Construction de Luby : non-régulier aléatoire

La méthode générale de construction des codes non-réguliers aléatoires selon Luby [47] est décrite comme suit :

- Définir tout d'abord le nombre total de liaisons entre les noeuds de variable et les noeuds de parité N_L . Pour les codes réguliers, $N_L = N \cdot w_c = (N - K) \cdot w_l$.
- Affecter à chaque noeud de variable noté v_i un certain nombre $w_{c,i}$ (constant ou non) de liaisons vers les noeuds de parité de sorte que $\sum_{i=1}^N w_{c,i} = N_L$.
- De la même manière, affecter à chaque noeud de parité noté c_i un certain nombre $w_{l,i}$ (constant ou non) de liaisons vers les noeuds de variable de sorte que $\sum_{i=1}^m w_{l,i} = N_L$.
- Choisir une permutation aléatoire du vecteur $P = [1 \ 2 \dots N_L]$ pour former le vecteur permuté π . Le vecteur P correspond aux liaisons associées aux noeuds de variable, et le vecteur π aux noeuds de parité.
- Identifier chaque $P(i)$, $i \in [1 \dots N_L]$ à la valeur $\pi(i)$ pour créer la liaison entre le noeud de variable associé à $P(i)$ et le noeud de parité associé à $\pi(i)$.
- Si plusieurs liaisons se retrouvent ainsi formées entre un même noeud de parité et un noeud de variable, les supprimer de sorte qu'il n'en reste qu'une.
- Construire la matrice de parité à partir du graphe de Tanner ainsi constitué.

Annexe D

Algorithmes de décodage à propagation de croyance

D.1 Notations générales

Pour chacune des deux versions présentées ici de l'algorithme de décodage à propagation de croyance (ou BP : Belief Propagation), les paramètres suivants doivent être définis :

$v = [v_0 v_1 \dots v_j \dots v_{N-1}]$: vecteur de N bits correspondant aux valeurs présentes sur les N noeuds de variable

$c = [c_0 c_1 \dots c_i \dots c_{m-1}]$: vecteur de m bits correspondant aux valeurs présentes sur les m noeuds de parité

$\hat{v} = [\hat{v}_0 \hat{v}_1 \dots \hat{v}_j \dots \hat{v}_{N-1}]$: vecteur de N bits correspondant aux valeurs décidées à chaque itération pour les N noeuds de variable

$\mathcal{N}(i)$: ensemble des noeuds de variable v_j qui participent au i^{me} contrôle de parité

$$\mathcal{N}(j) = \{i : H_{ji} = 1\}$$

$\mathcal{M}(j)$: ensemble des noeuds de contrôles c_i constituant une équation de parité à laquelle le noeud de variable v_j participe

$$\mathcal{M}(j) = \{i : H_{ij} = 1\}$$

$\mathcal{N}(i)\setminus j$: sous-ensemble obtenu à partir de $\mathcal{N}(i)$ privé du j^{me} noeud de variable v_j

$\mathcal{M}(j)\setminus i$: sous-ensemble obtenu à partir de $\mathcal{M}(j)$ privé du i^{me} noeud de contrôle c_i

Dans chacune des deux versions de l'algorithme, on notera qu'il y a, comme pour le décodage hard, 4 étapes importantes :

1. Initialisation
2. Etape "horizontale" : calcul des messages au niveau des noeuds de contrôle et émission vers les noeuds de variable
3. Etape "verticale" : calcul des messages au niveau des noeuds de variable et émission vers les noeuds de contrôle.
4. Prise de décision et test : si $\hat{v} \times H^T = 0$, arrêt de l'algorithme, sinon, retour à l'étape 2

D.2 Le décodeur somme-produit ou à propagation de croyance

Dans cette version de l'algorithme, on considère les deux quantités suivantes :

q_{ij}^x : probabilité que le bit de variable v_j ait la valeur x (0 ou 1) en fonction des informations obtenues par tous les noeuds de contrôle sauf le i^{me}

r_{ij}^x : probabilité que le bit de contrôle c_i ait sa valeur actuelle si le bit de variable v_j a la valeur x sachant que les autres bits de variables ont une distribution décrite par les

$$q_{ij'}^x, j' \in \mathcal{N}(i)\setminus j$$

Les étapes se déroulent selon la procédure suivante.

1. Initialisation

On définit à partir de la statistique du canal p_j^0 la probabilité a priori que le bit v_j soit un 0 et p_j^1 la probabilité que ce soit un 1 :

$$p_j^0 = P(v_j = 0), \quad p_j^1 = P(v_j = 1) = 1 - p_j^0$$

On initialise alors, pour chaque noeud de variable v_j et chaque noeud de contrôle c_i les quantités q_{ij}^0 et q_{ij}^1 avec ces probabilités a priori :

$$q_{ij}^0 = p_j^0$$

$$q_{ij}^1 = p_j^1$$

2. Etape horizontale : calcul des messages au niveau des noeuds de contrôle et émission vers les noeuds de variable

A l'aide des équations de parité définissant les ensembles $\mathcal{N}(i)$ et $\mathcal{M}(j)$ on calcule, au niveau de chaque noeud de contrôle c_i , les probabilités r_{ij}^0 et r_{ij}^1 pour chaque j , de la manière suivante :

$$r_{ij}^0 = \sum_{v_{j'} : j' \in \mathcal{N}(i) \setminus j} P(c_i | v_j = 0, \{v_{j'} : j' \in \mathcal{N}(i) \setminus j\}) \prod_{j' \in \mathcal{N}(i) \setminus j} q_{ij'}^{v_{j'}}$$

$$r_{ij}^1 = \sum_{v_{j'} : j' \in \mathcal{N}(i) \setminus j} P(c_i | v_j = 1, \{v_{j'} : j' \in \mathcal{N}(i) \setminus j\}) \prod_{j' \in \mathcal{N}(i) \setminus j} q_{ij'}^{v_{j'}}$$

Les probabilités conditionnelles dans la somme ne peuvent prendre en réalité que les valeurs 0 ou 1 en fonction des valeurs de c_i , v_j et des $v_{j'}$. Les expressions des probabilités r_{ij}^0 et r_{ij}^1 peuvent donc se simplifier en posant [2] :

$$\delta q_{ij} = q_{ij}^0 - q_{ij}^1 \quad \text{et} \quad \delta r_{ij} = r_{ij}^0 - r_{ij}^1$$

il vient :

$$\delta r_{ij} = (-1)^{c_i} \prod_{j' \in \mathcal{N}(i) \setminus j} \delta q_{ij'}$$

$$r_{ij}^x = \frac{(1 - (-1)^x \delta q_{ij})}{2}, \quad x \in \{0, 1\}$$

Ces valeurs de probabilités sont ensuite envoyées vers chaque noeud de variable.

3. Etape verticale : calcul des messages au niveau des noeuds de variable et émission vers les noeuds de contrôle

A partir des valeurs r_{ij}^x reçues sur chaque noeud de variable, les probabilités q_{ij}^x sont mises à jour. Ainsi, au niveau de chaque noeud de variable v_j , on calcule pour chaque i les quantités :

$$q_{ij}^0 = \alpha_{ij} p_j^0 \prod_{i' \in \mathcal{M}(j) \setminus i} r_{i'j}^0$$

$$q_{ij}^1 = \alpha_{ij} p_j^1 \prod_{i' \in \mathcal{M}(j) \setminus i} r_{i'j}^1$$

Où la valeur de α_{ij} est fixée de sorte que

$$q_{ij}^0 + q_{ij}^1 = 1$$

Ces valeurs de probabilités sont ensuite envoyées vers chaque noeud de contrôle.

On détermine aussi au niveau des noeuds de variable v_j pour chaque j les probabilités a posteriori avec :

$$q_j^0 = \alpha_{ij} p_j^0 \prod_{i \in \mathcal{M}(j)} r_{ij}^0$$

$$q_j^1 = \alpha_{ij} p_j^1 \prod_{i \in \mathcal{M}(j)} r_{ij}^1$$

4. Prise de décision et test

A partir des valeurs q_j^0 et q_j^1 , une décision hard est effectuée au niveau de chaque noeud de variable v_j pour former le mot décidé \hat{v} .

On teste alors si $\hat{v} \times H^T = 0$. En cas de succès le décodage s'arrête, sinon on revient à l'étape 2.

D.3 Le décodeur LLR somme-produit ou à propagation de croyance de croyance

Dans cette version, les messages échangés sont sous forme de rapports logarithmiques de vraisemblance (LLR : Logarithmic Likelihood Ratio). Les quantités suivantes doivent alors être définies :

F_j : LLR a priori déterminé à partir du bit reçu sur le noeud de variable v_j

L_{ij} : LLR envoyé par le noeud de contrôle c_i au noeud de variable v_j en fonction des informations de tous les noeuds de variable $v_{j'}, j' \neq j$

z_{ij} : LLR envoyé par le noeud de variable v_j au noeud de contrôle c_i à partir des informations F_j et $L_{i'j}, i' \neq i$

z_j : LLR a posteriori de chaque noeud de variable v_j à chaque itération.

Les différentes étapes se déroulent alors selon la procédure suivante.

1. Initialisation

La statistique du canal permet de déterminer p_j^0 la probabilité a priori que le bit reçu v_j soit un 0 et p_j^1 la probabilité que ce soit un 1 sachant que le bit émis est u_j .

$$p_j^0 = P(v_j = 0|u_j), p_j^1 = P(v_j = 1|u_j) = 1 - p_j^0$$

Comme on suppose que les variables v_j et u_j sont aléatoires et équiprobables, la règle de Bayes s'applique et pour $x = 0$ ou 1 on a :

$$p_j^x = P(v_j = x|u_j) = P(u_j = x|v_j).$$

Le LLR a priori étant défini par $LLR = \ln \left(\frac{P(u_j=1|v_j)}{P(u_j=0|v_j)} \right)$, il est alors égal à $LLR = \ln \left(\frac{p_j^1}{p_j^0} \right)$.

Ainsi, on initialise $F_j = \ln \left(\frac{p_j^1}{p_j^0} \right)$.

Ensuite, pour chaque noeud de variable v_j , et chaque noeud de contrôle c_i , on initialise :

$$z_{ij} = F_j$$

2. Etape horizontale : calcul des messages au niveau des noeuds de contrôle et émission vers les noeuds de variable

Les LLR émis par chaque noeud de contrôle c_i ont ici l'expression suivante :

$$L_{ij} = \ln \left(\frac{1-T_{ij}}{1+T_{ij}} \right)$$

où T_{ij} est calculé à chaque noeud de contrôle c_i par :

$$T_{ij} = \prod_{j' \in \mathcal{N}(i) \setminus j} \frac{1-e^{z_{ij'}}}{1+e^{z_{ij'}}$$

3. Etape verticale : calcul des messages au niveau des noeuds de variable et émission vers les noeuds de contrôle

Les LLR émis par chaque noeud de variable v_j s'expriment alors par :

$$z_{ij} = F_j + \sum_{i' \in \mathcal{M}(j) \setminus i} L_{i'j}$$

On détermine aussi au niveau des noeuds de variable v_j pour chaque j les LLR a posteriori avec :

$$z_j = F_j + \sum_{i \in \mathcal{M}(j)} L_{ij}$$

4. Prise de décision et test

A partir des valeurs z_j on détermine le mot décidé $\hat{v} = [\hat{v}_j]$ avec la règle de décision :

$$\hat{v}_j = 1 \text{ si } z_j > 0 \text{ et } \hat{v}_j = 0 \text{ si } z_j < 0$$

On teste alors si $\hat{v} \times H^T = 0$. En cas de succès le décodage s'arrête, sinon on revient à l'étape 2.

Annexe E

Application du codage correcteur d'erreurs à l'anneau optique ECOFRAME

E.1 Présentation

Le réseau proposé dans le projet ECOFRAME (Eléments de CONvergence pour les Futurs Réseaux d'Accès et METropolitains haut débit) [59] répond aux besoins de nouveaux concepts en matière d'architecture du réseau d'accès de part le développement de nouvelles applications en temps réel et à haut débit. Dans cet objectif, une solution de réseau métro-accès à bas coût, utilisant une topologie anneau et fonctionnant en mode paquet avec un multiplexage en longueur d'onde (WDM : Wavelength Division Multiplexing) est envisagée. L'anneau optique doit supporter un débit global proche de 400 Gb/s (40 longueurs d'ondes à 10 Gb/s chacune). La configuration du réseau et des noeuds associés inclut plusieurs composants actifs amplificateurs, nécessaires à la transmission

mais aussi générateurs de bruit et de non-linéarités altérant les performances.

Le but de notre contribution est de pouvoir relâcher les contraintes sur les composants optiques et de compenser les dégradations qu'ils engendrent à l'aide d'un système de codage correcteur d'erreurs. A partir des codes correcteurs d'erreurs étudiés précédemment, une solution est recherchée afin de garantir un TEB de 10^{-12} tandis que le TEB est initialement de 10^{-5} .

Pour dimensionner les paramètres du code correcteur, l'étude est réalisée tout d'abord sur un canal AWGN avant d'être transposée sur un canal χ^2 . En effet, des simulations physiques réalisées dans ce contexte [65,96] ont mis en évidence que la statistique du canal réel s'apparente plus à une distribution de type χ^2 non centrée qu'à une gaussienne. De plus, l'étude réalisée précédemment dans ce chapitre a montré que les résultats en termes de correction d'erreurs sont identiques sur chacun des deux canaux et pouvaient donc se transposer d'un canal à un autre.

E.2 Contraintes et caractéristiques liées au contexte

Contraintes

Le réseau ECOFRAME fonctionne en mode paquet, c'est-à-dire que les données issues de différents utilisateurs et /ou services sont groupées dans des paquets appelés "containers", transportés sur le réseau, qu'ils soient pleins, entièrement, partiellement ou même vides. Chaque container est de taille fixe, ce qui impose des limites (remplissage non optimal, perte de paquets...) mais facilite la gestion des paquets dans les noeuds du réseau. De plus, chaque container possède un en-tête qui comporte une bande de garde de $200ns$, et un préambule de $20ns$. Les paquets clients à l'intérieur du container étant de flux divers, ont chacun un en-tête particulier pour les informations de service.

Les données à protéger sont au niveau du container. Le préambule et l'en-tête ne sont pas protégés par le code correcteur car les informations doivent être directement accessibles lors du routage. Dans l'étude réalisée, la durée de la partie utile du container est fixée à $10\mu s$, mais elle peut être modifiée a posteriori. De plus, le débit en ligne maximal correspondant au débit bit (d'informations, de redondance ou de bourrage) est fixé.

La figure E.1 représente un container tel que décrit dans le projet ECOFRAME.

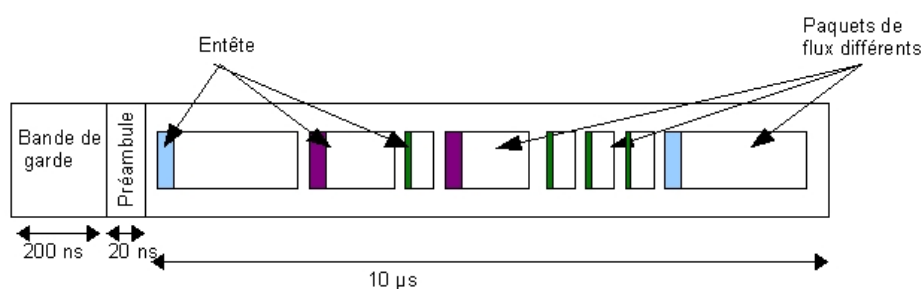


FIG. E.1 – Structure du container Ecoframe

Pour dimensionner le code correcteur, il faut évaluer le nombre de bits dans le container. En fonction du débit en ligne maximal noté D_{Lmax} et de la durée du container d , le nombre maximal de bits par container est fixé par $N_{bitmax} = D_{Lmax} \cdot d$. Pour un débit en ligne maximal de 10 Gb/s , une durée de $10\mu s$ correspond à $N_{bitmax} = 100\,000$ bits. Etant donnée la taille importante du paquet, il est impossible d'appliquer un seul code correcteur sur l'ensemble des bits.

La solution envisagée représentée figure E.2, consiste à découper le container indépendamment de la nature des paquets, en un nombre entier de "trames FEC (Forward Error Correction)" α tel que :

$$\alpha = \left\lceil \frac{N_{bitmax}}{N_{trame}} \right\rceil \quad (\text{E.1})$$

Avec N_{trame} la taille d'une trame FEC en bit et N_{bitmax} la taille du container (en bits).

Dans ce cas, suivant les dimensions du code utilisé, le nombre de bits de bourrage

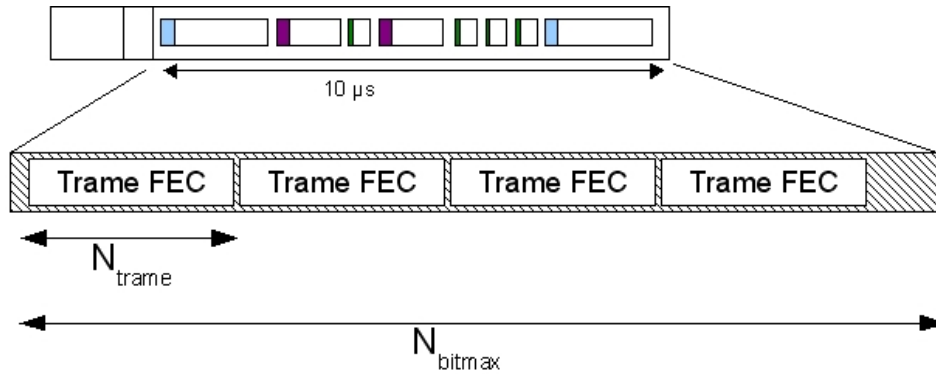


FIG. E.2 – Découpage du container en trame FEC

donné par $N_{bitmax} - N_{trame}$ est plus ou moins important. Le but sera de le minimiser de manière à maximiser le débit utile réel.

Caractéristiques du canal

Des simulations physiques du réseau ont été réalisées dans le cadre du projet [65, 96] à partir d'un simulateur tenant compte des bruits cumulés engendrés par les amplificateurs optiques en ligne, de la dispersion de la fibre et des phénomènes de non-linéarité. Les simulations sont réalisées dans le cas où les effets non linéaires ne sont pas importants. De plus, le réseau est modélisé par des noeuds (dont le nombre varie de 10 à 20), comportant des amplificateurs optiques engendrant un bruit de type ASE gaussien.

Les diagrammes de l'oeil obtenus permettent d'identifier la statistique réelle des signaux optiques se propageant sur le réseau, en effectuant une analyse de données par des méthodes de régression.

Des calculs d'erreur quadratique montrent qu'une statistique χ^2 non centrée et d'ordre $n = 4$ décrit plus fidèlement la statistique réelle du photocourant qu'une statistique gaussienne. Un exemple de cette statistique est représenté figure E.3.

Ainsi, le canal est considéré par la suite comme un canal χ^2 d'ordre $n = 4$.

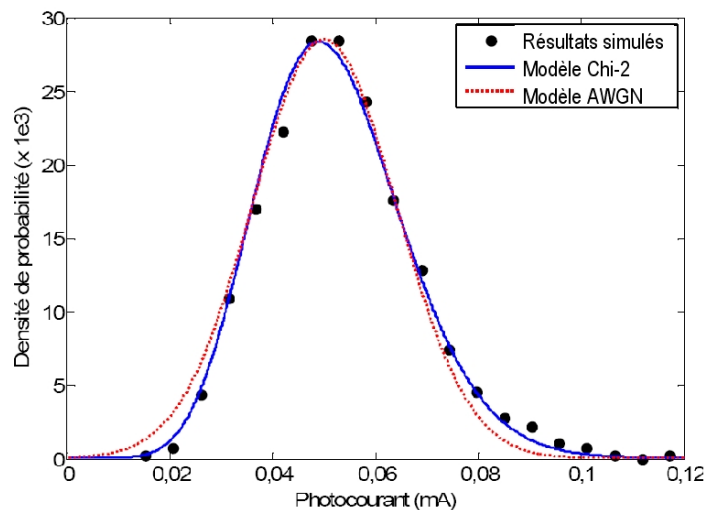


FIG. E.3 – Régression gaussienne et χ^2

Dans le cas particulier $n = 4$, la densité de probabilité de la variable χ^2 s'exprime de la manière suivante :

$$p_Y(y) = \frac{1}{2\sigma^2} \sqrt{\frac{y}{s^2}} e^{-(s^2+y)/(2\sigma^2)} I_1 \left(\sqrt{y} \frac{s}{\sigma^2} \right) \quad (\text{E.2})$$

pour $y \geq 0$, avec $s^2 = \sum_{i=1}^4 m_i^2$ et m_i les moyennes des variables gaussiennes élevées au carré.

On considère de plus que les moyennes m_i sont toutes identiques, égales à m , on a alors $s^2 = 4m^2$.

E.3 Dimensionnement des codes RS

Paramètres du dimensionnement

Pour rappel, les codes RS sont caractérisés par le paramètre m , définissant la classe du code, c'est-à-dire le nombre de bits par symbole. Ce paramètre influe sur le nombre de

m (classe du code)	$N = 2^m - 1$ (symboles)	N_{trame} (bits)	α	Débit en ligne réel D_L (Gb/s)
6	63	378	264	9,98
7	127	889	112	9,96
8	255	2040	49	10
9	511	4599	21	9,66
10	1023	10230	9	9,21
11	2047	22517	4	9,01

TAB. E.1 – Nombre de bits par trame FEC et débit en ligne réel pour différentes classes de codes RS

bits par trame N_{trame} qui est alors donné pour un code RS(N, K) par :

$$N_{trame} = N \cdot m = (2^m - 1) \cdot m \quad (\text{E.3})$$

Le nombre de bits utiles par trame FEC est donné par :

$$N_{utile} = K \cdot m \quad (\text{E.4})$$

Ainsi, pour m variant entre 6 et 11, on obtient 6 tailles possibles N_{trame} pour les trames FEC (tableau E.1). Le nombre α de trames FEC par container est alors déterminé à partir de (E.1). Le nombre de bits (utiles+redondants hors bits de bourrage) par container est ensuite déterminé de sorte à obtenir le débit en ligne réel D_L , défini comme le nombre de bits (utiles+redondants) transportés par seconde. Ces différents débits sont reportés dans le tableau E.1.

A partir du tableau E.1, la classe de code maximisant le débit en ligne réel correspond à $m=8$. De plus, pour un code de paramètres (N, K), le rendement est $R = K/N$ et le débit utile est obtenu par :

$$D_u = R \cdot D_L \quad (\text{E.5})$$

Influence des paramètres sur les performances

Comme vu dans la partie 2.4.3.2, la redondance et la longueur du code RS sont des paramètres influant sur les performances. Pour une valeur de m fixée, plus la redondance augmente plus le TEB s'améliore. De plus, pour une redondance fixée, les codes RS sont d'autant plus efficaces que leur longueur est grande, c'est-à-dire que la classe du code m est élevée. L'influence de chacun de ces deux paramètres est analysée plus particulièrement dans les paragraphes suivants.

Influence de la redondance

m étant fixé (donc la longueur du code $N = 2^m - 1$ fixée), les TEB obtenus par simulation pour différentes redondances de code sont représentés en fonction du rapport signal à bruit sur les figures E.4 à E.7.

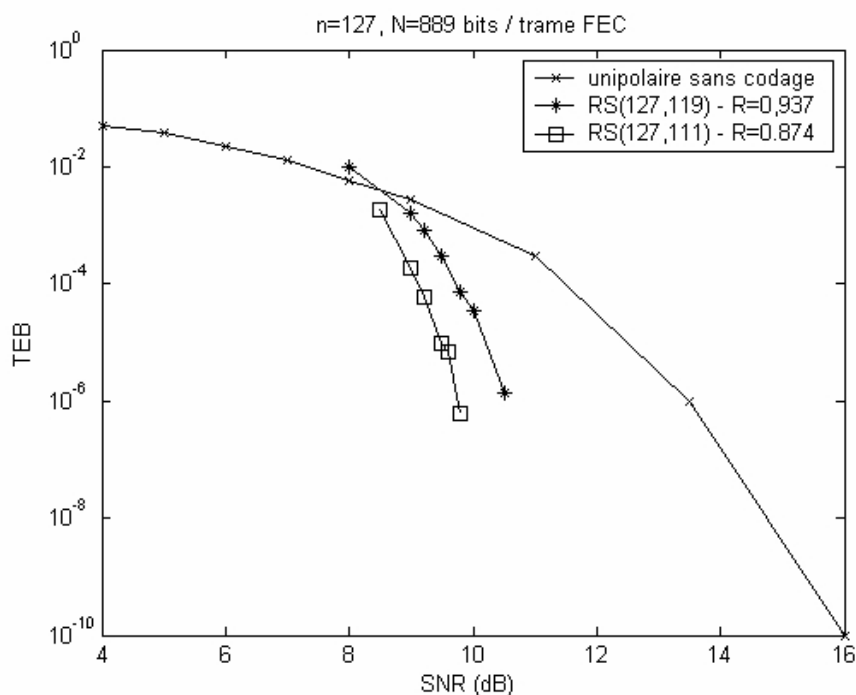


FIG. E.4 – TEB obtenus par simulation pour les codes RS de classe $m = 7$

Ces courbes confirment que pour une valeur de m fixée, plus la redondance augmente

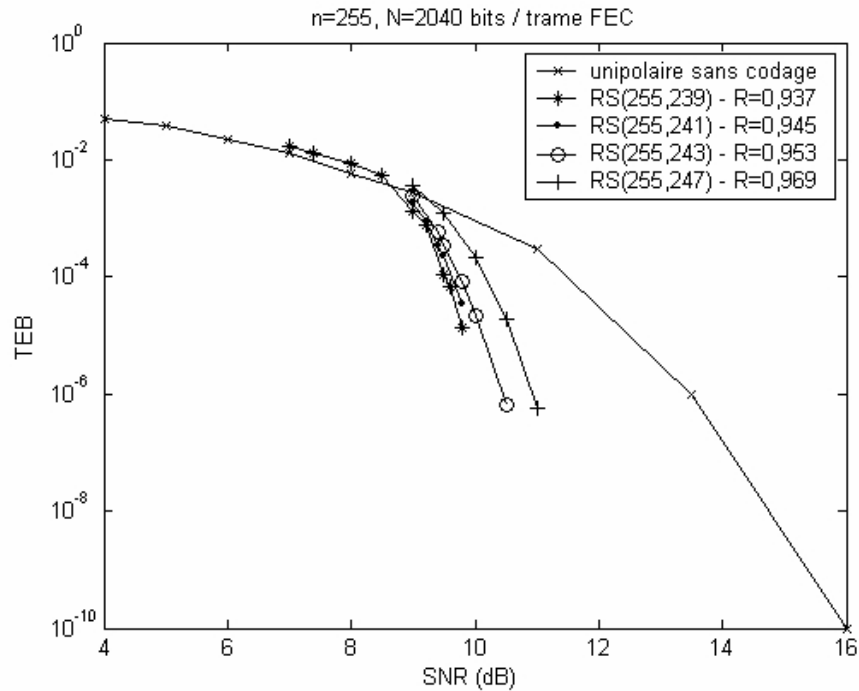


FIG. E.5 – TEB obtenus par simulation pour les codes RS de classe $m = 8$

plus le TEB s'améliore. De plus, les résultats peuvent être extrapolés pour obtenir les gains de codage, comme nous le verrons par la suite.

Influence de la longueur

Pour comparer les différents codes, les performances obtenues pour une même redondance ($\approx 6,3\%$) sont représentées sur la figure E.8. Le rendement des codes est dans ce cas $R = K/N = 0,937$.

A partir de ces courbes, on retrouve le fait que le pouvoir de correction des codes RS est d'autant plus élevé que la classe du code (ou sa longueur) est grande.

Gains de codage

Les gains de codage (en dB) à 10^{-12} ont pu être extrapolés à partir des résultats obtenus par simulation et présentés précédemment. Ils sont comparés dans le tableau E.2 à ceux obtenus avec les formules du standard ITU développées au début du chapitre. Pour rappel,

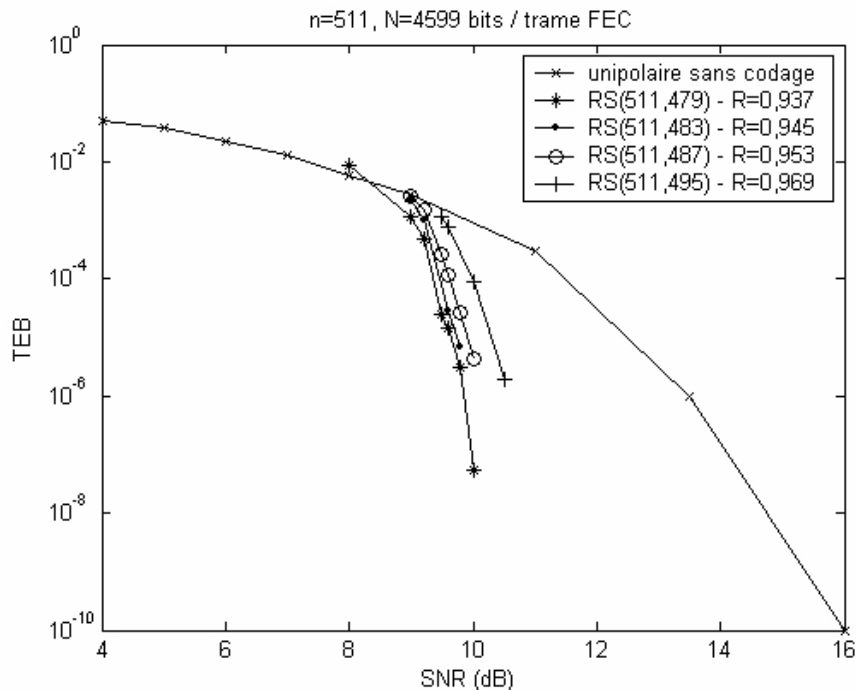


FIG. E.6 – TEB obtenus par simulation pour les codes RS de classe $m = 9$

le gain de codage correspond à la différence de rapport signal à bruit entre le cas non codé et le cas codé à un TEB fixé. Le rapport signal à bruit requis pour atteindre 10^{-12} dans le cas non codé est de 17 dB.

A partir de ces valeurs, on peut remarquer que les gains de codage obtenus à partir de la formule de standardisation ITU sont dans la plupart des cas, proches de ceux obtenus par simulation. Cependant, dans certains cas, la formule théorique surestime jusqu'à 0,7dB le gain de codage.

Dimensionnement des codes RS

La figure E.9 représente le gain de codage en fonction de la longueur de la trame FEC, et de la redondance ($\approx 1 - R$).

Pour un gain de codage minimum fixé, on peut déduire de la figure E.9 les paramètres des codes qui respectent certaines conditions. Si la condition souhaitée est de passer d'un

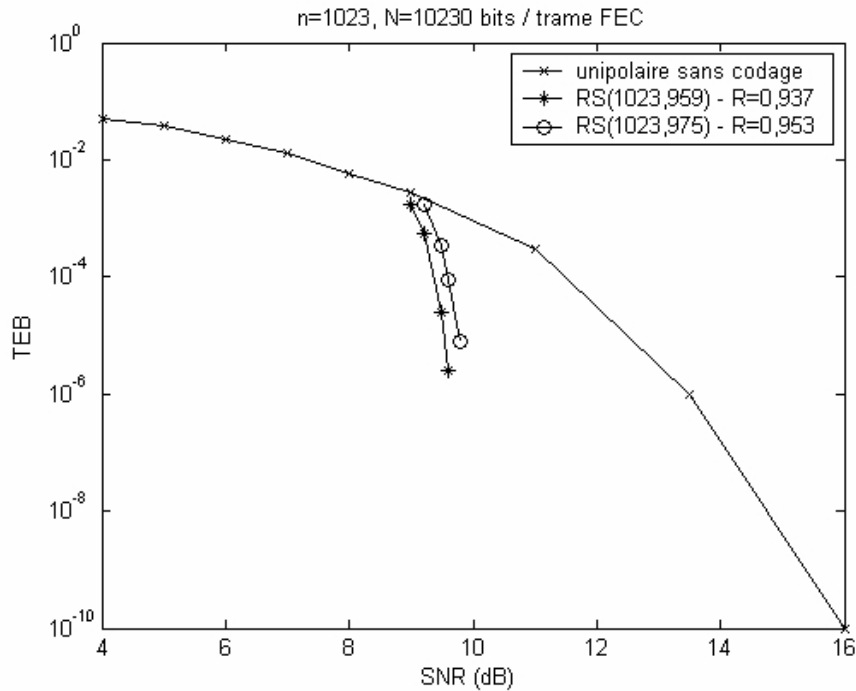


FIG. E.7 – TEB obtenus par simulation pour les codes RS de classe $m = 10$

TEB de 10^{-5} à un TEB de 10^{-12} , sachant qu'il faut un S/B sans codage de 12,5 dB pour atteindre 10^{-5} , le gain de codage minimum doit être de $17 - 12,5 = 4,5$ dB. Ainsi, à partir de la courbe du gain de codage, on ne garde que les codes satisfaisant cette condition, donc au-dessus de la droite tracée. Afin de maximiser le débit utile on relève pour chaque classe de code celui de plus faible redondance qui a un gain de codage suffisant. Les résultats sont reportés dans le tableau E.3. Le débit utile maximal est obtenu pour la classe de code $m = 8$, qui maximisait par ailleurs le débit en ligne réel avec les valeurs de D_{Lmax} et de d imposées. Cependant, les valeurs de débits en lignes précédemment obtenues pour une durée de $10\mu s$ peuvent être optimisées en rendant plus souple la valeur de la durée du container d (entre 9 et $11\mu s$ par exemple). La figure E.10 représente les variations pour chacun des codes pour m entre 7 et 10, du débit en ligne en fonction de d entre 9 et $11\mu s$. Les variations obtenues sont en dents de scie vu qu'on considère des nombres entiers de

E.3 Dimensionnement des codes RS

Code N,K	$R = K/N$	S/B à 10^{-12} (dB)	Gain de codage à 10^{-12} (dB)	Gain de codage à 10^{-12} Formules ITU (dB)
127,111	0,874	11,2	5,8	5,8
127,119	0,937	12,7	5	4,7
255,239		11,5	5,5	5,6
511,479		10,8	6,3	6,3
1023,959		10,1	6,9	6,7
255,241		0,945	12,2	4,8
511,483	0,953	11,2	5,8	6,1
255,243		12,5	4,5	5,2
511,487		11,7	5,3	5,9
1023,975		10,6	6,4	6,4
255,247		0,969	13	4
511,495	0,969	12,3	4,7	5,3

TAB. E.2 – Gains de codage obtenus pour différents codes RS

	$m=7$ $N_{trame} = 889$	$m=8$ $N_{trame} = 2040$	$m=9$ $N_{trame} = 4599$	$m=10$ $N_{trame} = 10230$
Redondance minimale	6,3%	5,5%	3,1%	3,1%
Rendement	0,937	0,945	0,969	0,969
D_L (Gbit/s)	9,96	10	9,66	9,21
Débit utile D_u (Gbit/s)	9,33	9,45	9,36	8,92

TAB. E.3 – Comparaison des débits utiles des différents codes respectant le gain de codage à 10^{-12}

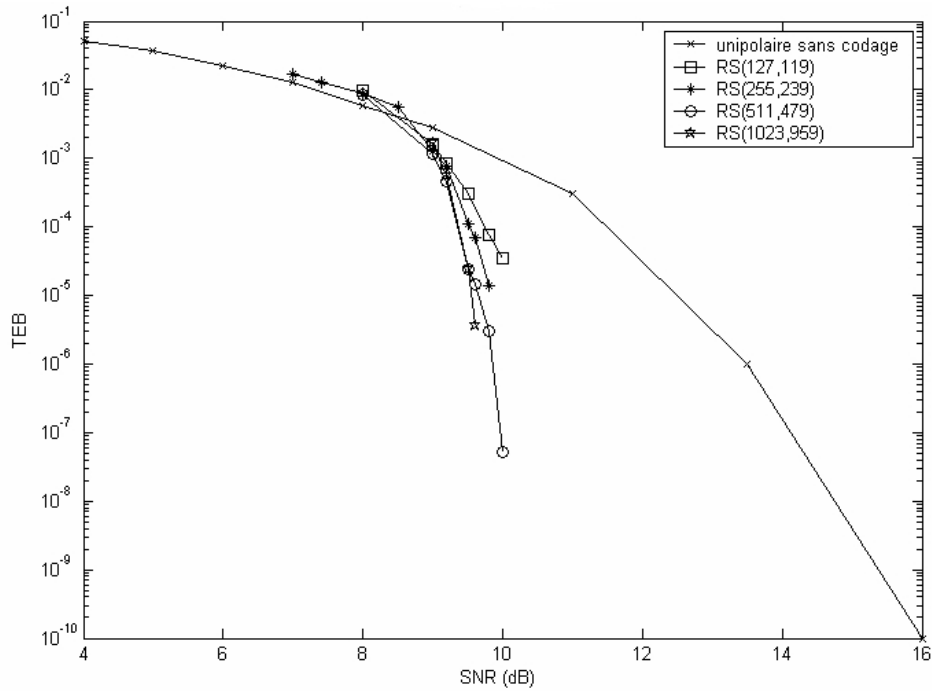


FIG. E.8 – TEB en fonction du S/B pour une redondance de 6,3% - $R = 0,937$

trames FEC dans les containers.

On retrouve sur cette courbe les débits présentés dans le tableau E.3 pour $d = 10\mu s$. La classe de code $m = 8$ permet alors d'obtenir le plus haut débit de 10 Gb/s. Ensuite, pour chaque code, il est possible de trouver entre 9 et $11\mu s$, une valeur de durée du container d maximisant le débit. Le débit utile optimal peut alors être déterminé pour chacun des codes précédemment sélectionnés (voir tableau E.4).

	m=7 $N_{trame} = 889$	m=8 $N_{trame} = 2040$	m=9 $N_{trame} = 4599$	m=10 $N_{trame} = 10230$
$d_{opt} (\mu s)$	9,7	10	10,6	10,3
D_L (Gbit/s)	9,99	10	9,98	9,91
Débit utile (Gbit/s) D_u	9,36	9,45	9,67	9,6

TAB. E.4 – Comparaison des débits utiles avec la durée du container adaptée à chacun des codes

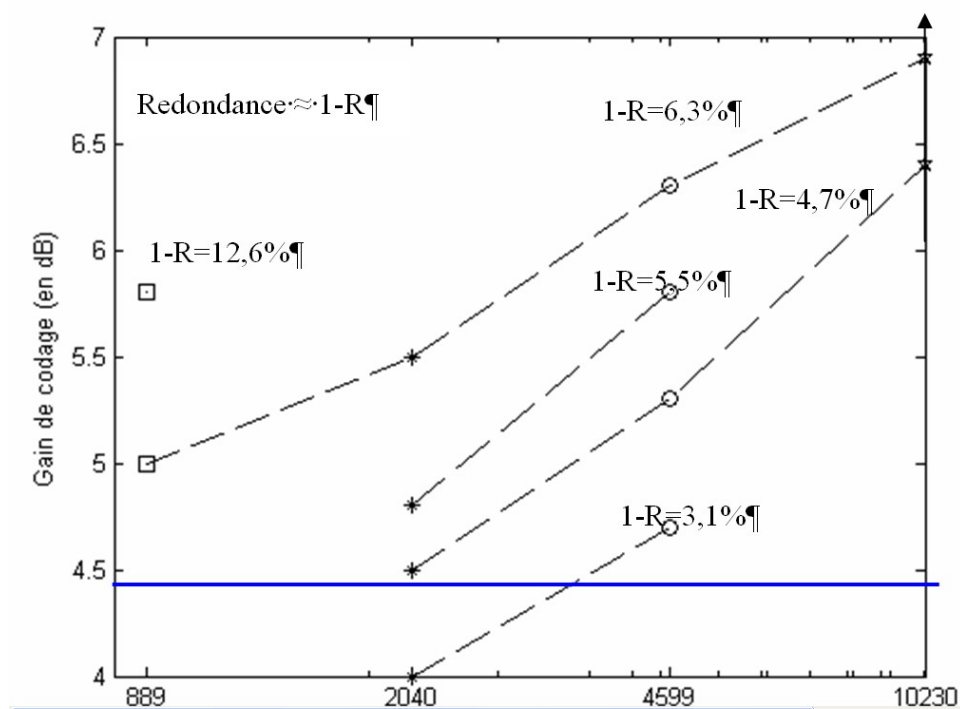


FIG. E.9 – Gain de codage en fonction de la taille de la trame FEC (en bits) N_{trame} et de la redondance

Dans ce cas, le code permettant le plus haut débit et respectant le gain de codage minimal est obtenu dans la classe $m=9$ avec un rendement de 0,969 (c'est le RS (511,495)). C'est donc celui-ci le plus adapté pour maximiser le débit utile.

Dans toute l'étude de dimensionnement réalisée, on a considéré que le débit en ligne maximal était limité par des contraintes physiques et par conséquent, le débit réel est inférieur au débit maximal, sa valeur dépendant des paramètres du code. Le raisonnement aurait pu se faire en fixant le débit en ligne utile réel (à 10 Gb/s par exemple) et en considérant ainsi que le débit en ligne n'est pas limité à une certaine valeur maximale. Dans ce cas, l'insertion de codes correcteurs d'erreurs introduit un sur-débit, puisque le débit réel des informations doit être plus élevé que le débit utile afin de compenser l'ajout de redondance. Le raisonnement, légèrement différent aurait conduit à des conclusions

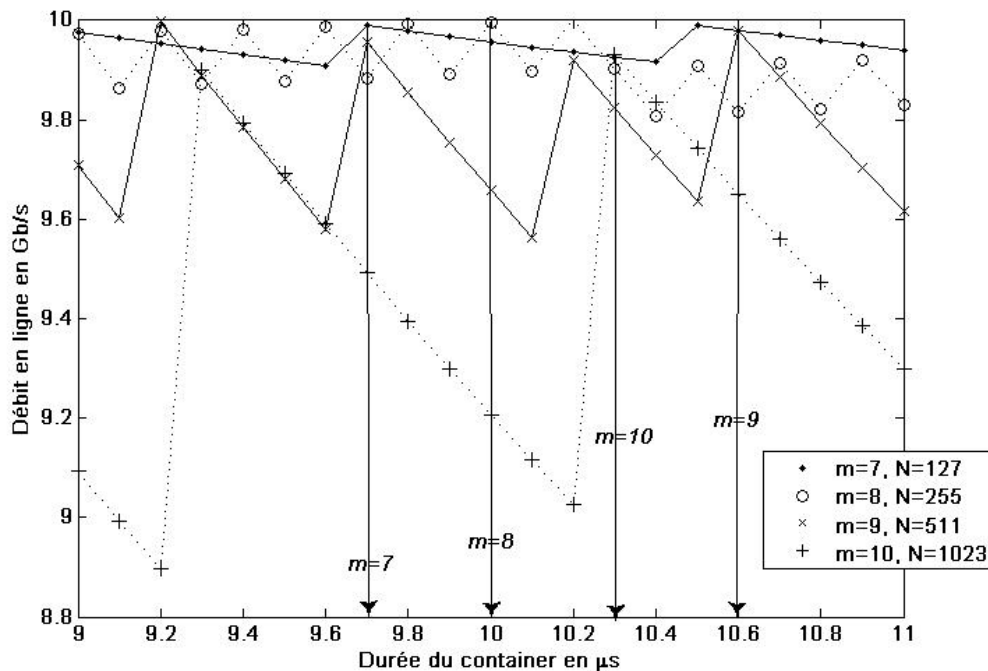


FIG. E.10 – Variation du débit en ligne en fonction de la durée du container autour de $10\mu s$

similaires.

Conclusion

L'étude réalisée sur les codes RS a permis de déterminer les gains de codage des codes RS en fonction de leur classe et de leur redondance. Le code optimal satisfaisant le gain de codage minimum et maximisant le débit utile réel pour une durée de container autour de $10\mu s$ a ainsi pu être déterminé. Pour d'autres valeurs de durée de container une étude similaire peut être effectuée. Sachant que les performances en termes de TEB donc en termes de gain de codage ne dépendent pas de la durée du container, l'étude du compromis porte alors uniquement sur la maximisation des débits en ligne.

Le paragraphe suivant détaille l'analyse paramétrique des codes LDPC sur un canal AWGN, avec les mêmes contraintes et caractéristiques liées au réseau que pour les codes

RS.

E.4 Dimensionnement des codes LDPC

Paramètres du dimensionnement

Nous considérons dans cette partie les codes LDPC construits de manière déterministe avec la méthode BIBD [50], qui comme nous l'avons précédemment vu, constituent un compromis complexité / pouvoir de correction. Contrairement aux RS qui ne proposent pas de taille (en bits) de trame FEC entre 2040 et 4599, puis entre 4599 et 10230, on pourra générer des codes LDPC de tailles intermédiaires.

L'étude des performances des codes RS a montré que la classe de code la plus adaptée est $m=9$ (c'est-à-dire 4599 bits par trame FEC). Nous nous concentrerons donc pour comparer les deux types de codes correcteurs sur des tailles de trames FEC entre 2000 et 20000 bits, ce qui correspond à des tailles de codes RS entre les classes 8 et 10.

Comme nous avons pu le voir dans la partie 1.4.2.3, les paramètres, permettant de définir la taille de la trame FEC, notée N_{trame} sont :

- q : nombre premier donnant la taille des vecteurs colonnes de la matrice de parité,
- w : le poids des colonnes de la matrice de parité (souvent pris égal à 3 ou 4),
- ns : nombre entier inférieur à $\left\lfloor \frac{q-1}{w(w-1)} \right\rfloor$, et correspondant au nombre de vecteurs colonnes permutés.

Les caractéristiques du code et de la trame FEC sont alors les suivantes :

- $N_{trame} = q \cdot ns$: nombre de bits par trame FEC
- $N_{utile} = q \cdot (ns - 1)$: nombre de bits utiles par trame FEC
- Rendement $R = \frac{ns-1}{ns}$

Le rendement du code R dépend uniquement du paramètre ns . Un tracé du rendement

est représenté figure E.11 en fonction de ns .

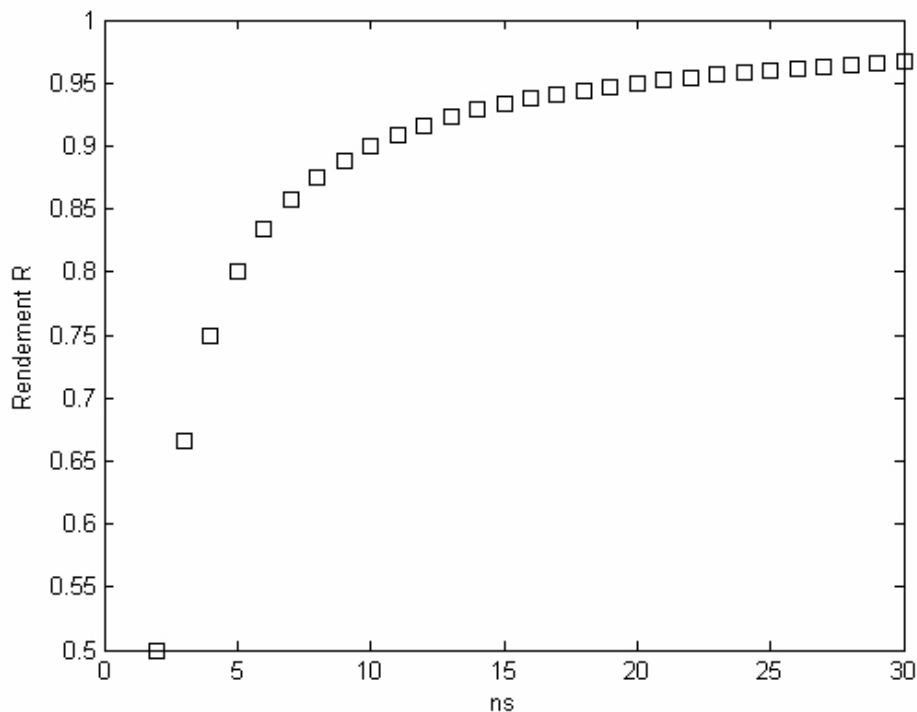


FIG. E.11 – Rendement du code LDPC en fonction de ns

Cette courbe permet de déterminer la valeur de ns nécessaire pour obtenir une redondance donnée. Au vu de l'évolution, on se rend compte qu'une forte précision sur la valeur du rendement est possible pour des valeurs proches de 1.

En ce qui concerne la taille des codes LDPC, elle dépend des deux paramètres ns et q , comme représenté sur la figure E.12.

A partir de cette courbe, on peut obtenir pour une valeur de ns fixée (donc pour une redondance fixée) les différentes tailles possibles de la trame FEC. Des exemples sont reportés pour $ns=25$ dans le tableau E.5.

Cet exemple montre qu'à redondance donnée (ici $ns=25$ correspondant à $R=0,96$ soit une redondance de 4%), on peut avoir des intermédiaires dans les valeurs des tailles de

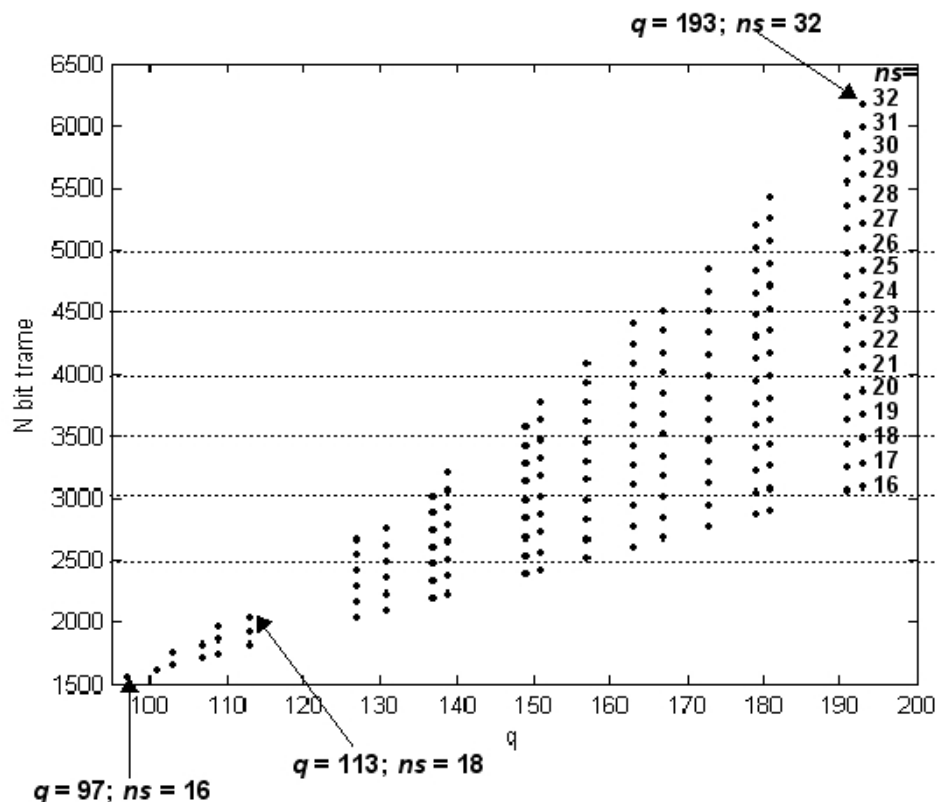


FIG. E.12 – Nombre de bits par trame FEC N_{trame} en fonction de q et de ns

q	151	157	163	167	173	179	181	...
N_{trame}	3775	3925	4075	4175	4325	4475	4525	...

TAB. E.5 – Tailles de trame FEC pour une redondance de 4% ($ns=25$)

trame FEC N_{trame} , contrairement aux codes RS.

La figure E.12 permet aussi d'obtenir pour une taille N_{trame} fixée, différents couples (q, ns) . Le tableau E.6 illustre certains exemples.

Construction de codes LDPC de redondance et taille équivalentes aux codes RS étudiés

Nous nous intéresserons dans un premier temps à des codes LDPC de rendements équivalents à ceux des codes RS étudiés précédemment. Les codes RS étudiés appartenant

N_{trame}	2500	3000	3500	4000	4500	5000
Couples	(131,19)	(137,22)	(151,23)	(157,25)	(167,27)	(179,28)
(q, ns)	(137,18)	(149,20)	(157,22)	(167,24)	(173,26)	(191,26)
	(149,17)	(157,19)	(167,21)	(173,23)	(181,25)	
	(157,16)	(167,18)	(173,20)	(181,22)		
		(179,17)				
		(191,16)				

TAB. E.6 – Couples (q, ns) correspondant à une taille de trame FEC recherchée

aux classes $m=7, 8, 9, 10$ ont des rendements respectivement égaux à $R=0,937; 0,9454; 0,969; 0,969$. Cela fixe donc la plage de valeurs pouvant être prises par le paramètre ns comme illustré tableau E.7 :

ns	R
16	0,938
18	0,944
32	0,969

TAB. E.7 – Rendements obtenus avec différentes valeurs de ns

Le second paramètre à déterminer est q , nombre premier fixant les valeurs N_{trame} et N_{utile} .

q est donné par le nombre premier le plus proche de $\left\lfloor \frac{N_{trame(souhaite)}}{ns} \right\rfloor$ où $N_{trame(souhaite)}$ correspond à une valeur approximative de la taille (en bits) envisagée et dont la plage de valeur [889-10230] est donnée par les classes de codes 7 à 10 des codes RS.

De plus, q doit satisfaire l'équation $ns \leq \left\lfloor \frac{q-1}{w(w-1)} \right\rfloor$ d'où la valeur minimale de q :

$$q_{min} = ns \cdot w \cdot (w - 1) + 1 \quad (E.6)$$

Les valeurs des paramètres q et ns correspondant aux codes RS étudiés précédemment sont regroupées dans le tableau E.8 pour un poids $w=3$.

Résultats de simulation

E.4 Dimensionnement des codes LDPC

R	$N_{\text{bittramesouhaite}}$ (RS)	ns	$\lfloor \frac{N_{\text{bittramesouhaite}}}{ns} \rfloor$	q_{min}	q (premier)	N_{bittrame}
0,938	889	16	56	97	97	1552
0,944	2040	18	113	109	113	2034
0,969	4599	32	144	193	193	6176
0,969	10230	32	320	193	317	10240

TAB. E.8 – Paramètres des codes LDPC proches des codes RS de classe $m = 7, 8, 9$ et 10

Les simulations réalisées avec les codes présentés tableau E.8 ont montré, que dans chacun des cas, le gain de codage à 10^{-12} avec des LDPC est supérieur de 2dB par rapport à celui obtenu avec les codes RS de redondance et taille équivalentes. Ce résultat est en accord avec les conclusions précédemment établies dans la partie 2.4.4.3.

L'utilisation des codes LDPC permet donc, à redondance donnée d'avoir une marge supplémentaire de 2 dB sur le rapport signal à bruit pour atteindre un TEB visé.

De plus, sachant que le gain de codage est, dans tous les cas envisagés, supérieur au gain de codage minimal (de 4,5 dB), il est possible de diminuer le taux de redondance en augmentant la valeur de ns . Ainsi, les débits utiles pourront être augmentés par rapport à ceux obtenus avec les codes RS, pour un gain de codage visé. Pour cela, une étude paramétrique complète des codes pouvant satisfaire un gain de codage de 4,5dB à 10^{-12} est nécessaire, et présentée dans le paragraphe suivant.

Dimensionnement des codes LDPC

Afin de dimensionner les codes LDPC permettant de maximiser le débit utile tout en respectant le gain de codage minimal fixé, les deux paramètres étudiés sont le débit utile réel et le gain de codage.

Débit utile réel

Pour un débit en ligne maximal de 10 Gb/s et une durée de container autour de 10 μs , on a cherché la valeur optimale de la durée du container entre 9 et 11 μs (permettant de

**Application du codage correcteur d'erreurs à l'anneau optique
ECOFRAME**

se rapprocher au maximum du nombre entier de trames FEC), pour chacun des codes. Les durées optimales et les débits utiles ainsi obtenus avec différents codes (caractérisés par ns) sont reportés dans le tableau E.9.

ns	N_{trame}	N_{utile}	d_{opt} (μs)	Nombre de trames FEC / container	Débit en ligne réel (Gbit/s)	Débit utile (Gbit/s)
18	1962	1853	10,4	53	10	9,44
19	2413	2286	9,9	41	9,99	9,47
20	2540	2413	9,4	37	10	9,5
21	2667	2540	10,7	40	9,97	9,5
22	3014	2877	10,9	36	9,95	9,5
23	3197	3058	9,6	30	9,99	9,56
24	3576	3427	9,3	26	10	9,58
25	3775	3624	10,2	27	9,99	9,59
26	4082	3925	9,8	24	10	9,61
27	4401	4238	9,7	22	9,98	9,61
28	4844	4671	9,7	20	9,99	9,63
29	5191	5012	10,4	20	9,98	9,64
30	5430	5249	9,8	18	9,97	9,64
31	5921	5730	9,5	16	9,97	9,65
32	6176	5983	10,5	17	10	9,69
33	6567	6368	9,2	14	9,99	9,69
35	7385	7174	10,4	14	9,94	9,66
40	9640	9399	9,7	10	9,94	9,69
45	12195	11924	11	9	9,98	9,76
46	12742	12465	10,2	8	9,99	9,78
50	15350	15043	10,8	7	9,95	9,75
55	18205	17874	11	6	9,93	9,75
60	22020	21653	9	4	9,79	9,62

TAB. E.9 – Débits utiles et durées optimales pour différentes valeurs de ns

Ces résultats montrent que quand ns devient trop important ($ns > 46$), les débits diminuent à cause de la taille du container qui limite le nombre de trames FEC par container et du fait que les trames FEC sont d'autant plus longues. Par conséquent, le débit maximal (9,78 Gb/s) est obtenu pour $ns=46$. Ce code correspond au LDPC(12742,12465), qui possède une taille importante. Si on se fixe une limite sur la

E.4 Dimensionnement des codes LDPC

taille du code N inférieure à 10000 bits, le code obtenu pour $ns = 32$: LDPC(6176,5983), est le code le plus court maximisant le débit utile qui vaut cette fois 9,69 Gb/s.

Gain de codage

Les résultats de simulation présentés pour les codes LDPC de taille et de redondance proches des codes RS envisagés ont montré que pour les codes LDPC tels que $ns \leq 32$, le gain de codage minimum était respecté. Ainsi, le code LDPC(6176,5983) obtenu pour $ns=32$ (soit 3,1% de redondance) correspond bien aux spécifications requises. Des simulations ont été réalisées de manière à vérifier que le code LDPC(12742,12465) obtenu avec $ns=46$ (soit 2,2 % de redondance), respecte aussi le gain de codage minimum de 4,5 dB.

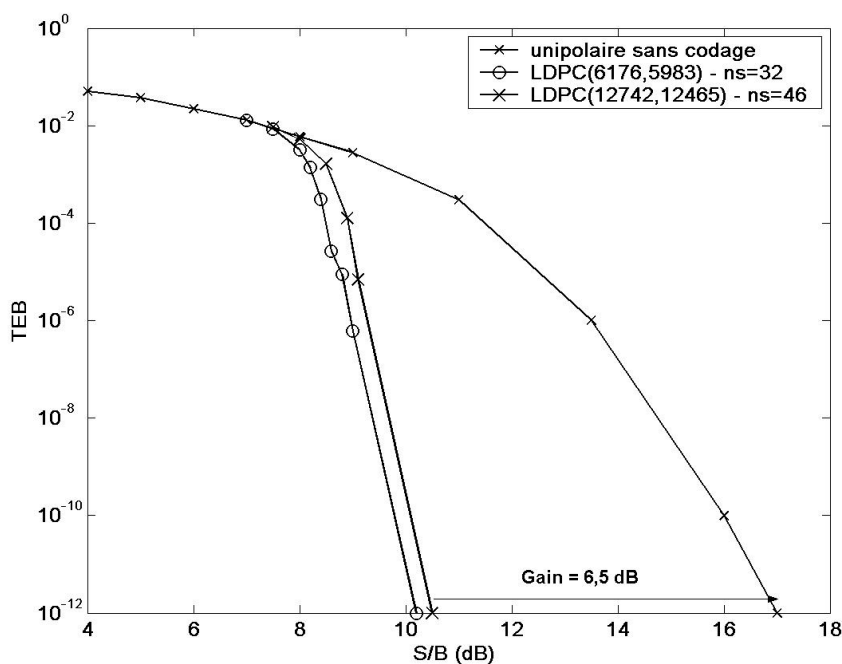


FIG. E.13 – Performances des codes LDPC(6176,5983) et LDPC(12742,12465)

Les courbes sur la figure E.13 représentent les TEB obtenus par simulation et en extrapolant pour les codes LDPC(6176,5983) et LDPC(12742,12465) obtenus

respectivement avec $ns=32$ et 46 . On peut en déduire que pour $ns=46$, les performances sont comme prévu, moins bonnes que pour $ns=32$ (car le code est moins redondant). Cependant, le gain de codage à 10^{-12} est de $6,5$ dB, ce qui respecte toujours la limite de $4,5$ dB.

Ainsi, pour maximiser le débit utile, il convient de choisir le code obtenu pour $ns=46$, c'est-à-dire le LDPC(12742,12465).

Conclusions

Les simulations réalisées ont montré qu'à redondance et taille équivalentes par rapport aux codes RS, les codes LDPC ont de meilleures performances. En effet, le gain de codage à 10^{-12} est supérieur en moyenne de 2 dB par rapport aux codes RS. Une étude prenant en considération le nombre de trames par container, en ajustant la taille du container autour de $10 \mu s + / - 1 \mu s$, a permis de déterminer le code LDPC maximisant le débit utile, tout en respectant le gain de codage. Si on se fixe une limite sur la taille du code inférieure à 10000 bits, afin d'avoir un codeur de complexité comparable au code RS, le code LDPC(6176,5983) obtenu avec $q = 193$ et $ns = 32$, (comparable au code RS(511,495) de longueur égale à $511 \times 9 = 4599$) apparaît comme celui maximisant le débit utile. En effet, de redondance égale à $3,1\%$, il conduit à un gain de codage de $6,7$ dB, et le débit utile est alors de $9,69$ Gb/s pour une durée optimale du container de $10,6 \mu s$.

Si aucune limite n'est fixée sur la taille de la trame, l'étude de dimensionnement aboutit au code LDPC(12742,12465) obtenu avec $q = 227$ et $ns = 46$, qui conduit à un débit utile de $9,78$ Gb/s. Ce code possède une redondance de $2,2\%$ et conduit à un gain de codage légèrement inférieur au code LDPC(6176,5983), c'est-à-dire égal à $6,5$ dB.

E.5 Adaptation au canal χ^2

Les simulations réalisées dans les parties 2.5.2.2 et 2.5.3.2 ont montré que les performances sur canal χ^2 sont conformes à celles obtenues sur canal AWGN à condition de choisir correctement le seuil de décision en ce qui concerne les codes RS et de calculer les LLR adéquats pour le décodage des codes LDPC.

En ce qui concerne les codes RS, les simulations réalisées parties 2.5.2.2 et 2.5.2.3 ont permis de vérifier tout d'abord que les TEB obtenus sont conformes à la formule théorique. De plus, la formule théorique ne dépendant pas de la statistique du canal, nous pouvons en conclure que le dimensionnement réalisé pour les codes RS sur canal AWGN est adaptable au cas du canal χ^2 d'ordre $n = 4$ caractérisant le canal ECOFRAME. Ainsi, le code RS(511,495) est le code également retenu pour le dimensionnement sur canal ECOFRAME.

Concernant les codes LDPC, dans le cas où $n = 4$, les LLR se résument alors, dans le cas réel ($A_0 \neq 0$) par :

$$LLR = \frac{A_1^2 - A_0^2}{2\sigma^2} + \ln \frac{A_1}{A_0} + \ln \left(\frac{I_1 \left(\sqrt{Y} \frac{A_0}{\sigma^2} \right)}{I_1 \left(\sqrt{Y} \frac{A_1}{\sigma^2} \right)} \right) \quad (\text{E.7})$$

Dans le cas idéal ($A_0 = 0$, et $ER_{in} = \infty$), on a :

$$LLR = \frac{A_1^2}{2\sigma^2} + \ln \left(\frac{\sqrt{Y} A_1}{2\sigma^2 I_1 \left(\sqrt{Y} \frac{A_1}{\sigma^2} \right)} \right) \quad (\text{E.8})$$

Les simulations précédemment réalisées (partie 2.5.3.2) ont été effectuées en considérant un canal χ^2 pour $n = 4$ avec le code LDPC(6176,5983) finalement retenu dans l'étude du dimensionnement ECOFRAME. Le gain de codage étant identique sur

chacun des deux canaux, ceci signifie que le dimensionnement réalisé sur canal AWGN est également valable sur un canal χ^2 . Ainsi, le code LDPC(6176,5963), sélectionné pour ses performances en termes de gain de codage et de débit utile disponible est retenu également pour le dimensionnement sur canal χ^2 d'ordre 4 représentant le canal ECOFRAME.

Nous avons également vu, à travers les simulations réalisées parties 2.5.2.3 et 2.5.3.2, qu'il est essentiel de prendre en compte la spécificité du canal pour que le décodage soit efficace. Pour les codes RS, cela revient à choisir un seuil optimal de détection, qui dépend des paramètres caractéristiques de la distribution χ^2 . Pour les codes LDPC, il faut adapter correctement l'algorithme de décodage en effectuant le calcul du LLR en fonction des caractéristiques du canal χ^2 . En effet, si aucune modification n'est apportée, et si on approxime le canal par un canal de type AWGN, le FEC n'est plus performant. Pour les codes LDPC, les modifications apportées au niveau de l'algorithme de décodage dans le cas d'un canal χ^2 ne portent que sur le calcul des valeurs initiales des LLR en fonction de la sortie du canal. Une fois les LLR initialisés, les calculs de LLR a posteriori suivent la même règle qu'il s'agisse d'un canal χ^2 ou AWGN. Ainsi, il suffira d'avoir en début de chaîne de réception un module capable de calculer le LLR initial correctement.

Annexe F

Obtention de la densité de probabilité de la variable décisionnelle avec le modèle statistique du beat noise

On part de l'expression générale de la densité de probabilité (3.55) :

$$p(Z_d) = \sum_i p(s_Z^2 = i) p_{s_Z^2}(Z_d)$$

A partir des différentes considérations, il est possible d'obtenir une expression complète de cette densité de probabilité. En effet, en remplaçant tout d'abord le terme $p_{s_Z^2}(Z_d)$ par son expression en fonction de W' (3.54), il vient :

$$p(Z_d) = \sum_i p(s_Z^2 = i) \cdot \left(p(W' = 0 | s_Z^2) \cdot p_{s_Z^2, W'=0}(Z_d) + p(W' \neq 0 | s_Z^2) \cdot p_{s_Z^2, W' \neq 0}(Z_d) \right) \quad (\text{F.1})$$

Pour remplacer les différents termes, on distingue maintenant les cas $b_d = 0$ et $b_d = 1$.

**Obtention de la densité de probabilité de la variable décisionnelle
avec le modèle statistique du beat noise**

Lorsque $b_d = 1$, on a d'après (3.51) : $s_Z^2 = W + I_d \geq W$.

Ainsi, on a : $p(s_Z^2 = i) = p(I_d + W = i) = p(I_d = i - W)$. En remplaçant cette relation dans (F.1) on obtient la densité de probabilité pour $b_d = 1$:

$$p^{(1)}(Z_d) = \sum_{i=W}^{W+U-1} p(I_d = i - W) \cdot (p(W' = 0 | s_Z^2) \cdot p_{i,W'=0}(Z_d) + p(W' \neq 0 | s_Z^2) \cdot p_{i,W' \neq 0}(Z_d)) \quad (\text{F.2})$$

En effectuant le changement de variable $i' = i - W$ on peut réécrire (F.2) en :

$$p^{(1)}(Z_d) = \sum_{i'=0}^{U-1} p(I_d = i') \cdot (p(W' = 0 | s_Z^2) \cdot p_{i'+W,W'=0}(Z_d) + p(W' \neq 0 | s_Z^2) \cdot p_{i'+W,W' \neq 0}(Z_d)) \quad (\text{F.3})$$

Sachant d'après (3.56) que pour $b_d = 1$ on a :

$$p(W' = 0) = \begin{cases} 1 & \text{si } I_d = 0 \\ 0 & \text{si } I_d \geq 1 \end{cases}$$

L'expression (F.3) peut se simplifier en :

$$p^{(1)}(Z_d) = p(I_d = 0) \cdot p_{W,W'=0}(Z_d) + \sum_{i=1}^{U-1} p(I_d = i) \cdot p_{i+W,W' \neq 0}(Z_d) \quad (\text{F.4})$$

Soit en remplaçant $p_{W,W'=0}(Z_d)$ par son expression sous forme de dirac (3.53), l'obtention de l'expression finale de la densité de probabilité pour un '1' émis comme (3.60) :

$$p^{(1)}(Z_d) = p(I_d = 0) \cdot \delta(Z_d - W) + \sum_{i=1}^{U-1} p(I_d = i) \cdot p_{i+W,W' \neq 0}(Z_d)$$

Avec pour rappel les expressions de $p(I_d = i)$ et de $p_{i+W,W' \neq 0}(Z_d)$ fournies respectivement

par (3.20) et (3.52) avec $s_Z^2 = i + W$:

$$p(I_d = i) = \binom{U-1}{i} p^i (1-p)^{U-1-i}$$

et

$$p_{i+W, W' \neq 0}(Z_d) = \frac{1}{2var} \left(\frac{Z_d}{i+W} \right)^{(nW'-2)/4} e^{-(i+W+Z_d)/(2var)} I_{nW'/2-1} \left(\frac{\sqrt{Z(i+W)}}{var} \right)$$

On considère maintenant le cas où $b_d = 0$, qui implique alors que $s_Z^2 = I_d$ et $p(s_Z^2 = i) = p(I_d = i)$. Ainsi, $p^{(0)}(Z_d)$ s'écrit :

$$p^{(0)}(Z_d) = \sum_i p(I_d = i) \cdot (p(W' = 0 | s_Z^2) \cdot p_{i, W'=0}(Z_d) + p(W' \neq 0 | s_Z^2) \cdot p_{i, W' \neq 0}(Z_d)) \quad (\text{F.5})$$

Sachant que pour $b_d = 0$, on a d'après (3.57), (3.58) et (3.59) :

$$p(W' = 0) = \begin{cases} 0 & \text{si } I_d > W \\ 1 & \text{si } I_d \in \{0, 1\} \\ \frac{\binom{W}{I_d}}{\binom{W+I_d-1}{I_d}} & \text{si } 2 \leq I_d \leq W \end{cases}$$

**Obtention de la densité de probabilité de la variable décisionnelle
avec le modèle statistique du beat noise**

On obtient alors en remplaçant dans (F.5) :

$$\begin{aligned}
 p^{(0)}(Z_d) &= p(I_d = 0) \cdot p_{0,W'=0}(Z_d) + p(I_d = 1) \cdot p_{1,W'=0}(Z_d) \\
 &+ \sum_{i=2}^W p(I_d = i) \cdot \left(\frac{\binom{W}{i}}{\binom{W+i-1}{i}} \cdot p_{i,W'=0}(Z_d) + \left(1 - \frac{\binom{W}{i}}{\binom{W+i-1}{i}} \right) \cdot p_{i,W' \neq 0}(Z_d) \right) \\
 &+ \sum_{i=W+1}^{U-1} p(I_d = i) \cdot p_{i,W' \neq 0}(Z_d) \tag{F.6}
 \end{aligned}$$

En remplaçant par les fonctions de dirac, on obtient l'expression finale de la densité de probabilité pour un '0' émis (3.61) :

$$\begin{aligned}
 p^{(0)}(Z_d) &= p(I_d = 0) \cdot \delta(Z_d) + p(I_d = 1) \cdot \delta(Z_d - 1) \\
 &+ \sum_{i=2}^W p(I_d = i) \cdot \left(\frac{\binom{W}{i}}{\binom{W+i-1}{i}} \cdot \delta(Z_d - i) + \left(1 - \frac{\binom{W}{i}}{\binom{W+i-1}{i}} \right) \cdot p_{i,W' \neq 0}(Z_d) \right) \\
 &+ \sum_{i=W+1}^{U-1} p(I_d = i) \cdot p_{i,W' \neq 0}(Z_d)
 \end{aligned}$$

Avec pour rappel les expressions de $p(I_d = i)$ et de $p_{i,W' \neq 0}(Z_d)$ fournies respectivement par (3.20) et (3.52) avec $s_Z^2 = i$.

Annexe G

Calcul de la densité de probabilité de la variable de bruit globale (IAM+AWGN)

La densité de probabilité de la variable de bruit globale est obtenue à partir de la densité de probabilité de chacune des variables : IAM et bruit gaussien. En effet, en supposant ces deux phénomènes indépendants, la densité de la somme de ces variables est obtenue comme le produit de convolution des densités de probabilités données par (3.84) et (3.81), ce qui s'exprime par :

$$f_T(x) = \int_{-\infty}^{+\infty} f_{I'}(t) \cdot f_n(x-t) dt \quad (\text{G.1})$$

En utilisant (3.84) et (3.81), il vient :

$$f_T(x) = \int_0^{+\infty} \frac{A}{\sqrt{2\pi\sigma_{I'}^2}} e^{-\frac{(t-m_{I'})^2}{2\sigma_{I'}^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-t)^2}{2\sigma^2}} dt \quad (\text{G.2})$$

En développant les termes en exponentielle et en les triant par rapport aux puissances de t il vient :

$$\begin{aligned} f_T(x) &= \frac{A \cdot e^{-\left(\frac{m_{I'}^2}{2\sigma_{I'}^2} + \frac{x^2}{2\sigma^2}\right)}}{2\pi\sigma_{I'}\sigma} \int_0^{+\infty} e^{-t^2\left(\frac{\sigma_{I'}^2+\sigma^2}{2\sigma_{I'}^2\sigma^2}\right) + t\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)} dt \\ &= \frac{A \cdot e^{-\left(\frac{m_{I'}^2}{2\sigma_{I'}^2} + \frac{x^2}{2\sigma^2}\right)}}{2\pi\sigma_{I'}\sigma} \int_0^{+\infty} e^{-\left(\frac{\sigma_{I'}^2+\sigma^2}{2\sigma_{I'}^2\sigma^2}\right)\left(t^2 - 2t\frac{\sigma_{I'}^2\sigma^2}{\sigma_{I'}^2+\sigma^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)\right)} dt \end{aligned} \quad (G.3)$$

On peut reconnaître dans l'exponentielle le début d'un carré de sorte que (G.3) puisse s'écrire :

$$f_T(x) = \frac{A \cdot e^{-\left(\frac{m_{I'}^2}{2\sigma_{I'}^2} + \frac{x^2}{2\sigma^2}\right)}}{2\pi\sigma_{I'}\sigma} \int_0^{+\infty} e^{-\left(\frac{\sigma_{I'}^2+\sigma^2}{2\sigma_{I'}^2\sigma^2}\right)\left(\left(t - \frac{\sigma_{I'}^2\sigma^2}{\sigma_{I'}^2+\sigma^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)\right)^2 - \left(\frac{\sigma_{I'}^2\sigma^2}{\sigma_{I'}^2+\sigma^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)\right)^2\right)} dt \quad (G.4)$$

En sortant de l'intégrale les termes indépendants de t , il vient :

$$\begin{aligned} f_T(x) &= \frac{A \cdot e^{-\left(\frac{m_{I'}^2}{2\sigma_{I'}^2} + \frac{x^2}{2\sigma^2}\right) + \left(\frac{\sigma_{I'}^2\sigma^2}{\sigma_{I'}^2+\sigma^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)\right)^2}}{2\pi\sigma_{I'}\sigma} \int_0^{+\infty} e^{-\left(\frac{\sigma_{I'}^2+\sigma^2}{2\sigma_{I'}^2\sigma^2}\right)\left(t - \frac{\sigma_{I'}^2\sigma^2}{\sigma_{I'}^2+\sigma^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)\right)^2} dt \\ &= \frac{A \cdot e^{\frac{(x-m_{I'})^2}{2(\sigma_{I'}^2+\sigma^2)}}}{2\pi\sigma_{I'}\sigma} \int_0^{+\infty} e^{-\left(\frac{\sigma_{I'}^2+\sigma^2}{2\sigma_{I'}^2\sigma^2}\right)\left(t - \frac{\sigma_{I'}^2\sigma^2}{\sigma_{I'}^2+\sigma^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)\right)^2} dt \end{aligned} \quad (G.5)$$

En définissant la valeur $\sigma_T^2 = \sigma_{I'}^2 + \sigma^2$ comme la variance globale de bruit, et en faisant le changement de variable :

$$u = t - \frac{\sigma_{I'}^2\sigma^2}{\sigma_T^2} \left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2} \right) \quad (G.6)$$

Il vient :

$$f_T(x) = \frac{A \cdot e^{\frac{(x-m_{I'})^2}{2\sigma_T^2}}}{2\pi\sigma_{I'}\sigma} \int_{-\frac{\sigma_{I'}^2\sigma^2}{\sigma_T^2}\left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2}\right)}^{+\infty} e^{-\left(\frac{\sigma_T^2}{2\sigma_{I'}^2\sigma^2}\right)u^2} du \quad (G.7)$$

Avec le changement de variables :

$$v = \frac{\sigma_T}{\sqrt{2}\sigma_{I'}\sigma} u \quad (\text{G.8})$$

L'équation (G.7) devient :

$$f_T(x) = \frac{A \cdot e^{-\frac{(x-m_{I'})^2}{2\sigma_T^2}}}{2\pi\sigma_{I'}\sigma} \frac{\sqrt{2}\sigma_{I'}\sigma}{\sigma_T} \int_{-\frac{\sigma_{I'}\sigma}{\sqrt{2}\sigma_T} \left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2} \right)}^{+\infty} e^{-v^2} dv \quad (\text{G.9})$$

$$= \frac{A \cdot e^{-\frac{(x-m_{I'})^2}{2\sigma_T^2}}}{\sqrt{2}\pi\sigma_T} \int_{-\frac{\sigma_{I'}\sigma}{\sqrt{2}\sigma_T} \left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2} \right)}^{+\infty} e^{-v^2} dv \quad (\text{G.10})$$

On reconnaît alors la fonction d'erreur complémentaire définie par :

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^{+\infty} e^{-v^2} dv \quad (\text{G.11})$$

Donc la densité de probabilité s'écrit finalement comme :

$$f_T(x) = \frac{A}{2\sqrt{2}\pi\sigma_T} e^{-\frac{(x-m_{I'})^2}{2\sigma_T^2}} erfc \left(-\frac{\sigma_{I'}\sigma}{\sqrt{2}\sigma_T} \left(\frac{m_{I'}}{\sigma_{I'}^2} + \frac{x}{\sigma^2} \right) \right) \quad (\text{G.12})$$

Annexe H

Calcul de Y_L

On considère une transmission OCDMA en présence de bruit gaussien, les erreurs sont effectuées à la fois sur les '0' qui sont bruités par le bruit global (bruit gaussien et IAM) mais aussi sur les '1' bruités par le bruit gaussien.

D'une manière générale, la valeur reçue par le décodeur peut s'écrire : $Y = x + n$ où n représente la contribution globale de bruit (AWGN+IAM) et x est le bit émis (0 ou 1).

Il existe alors une valeur Y_L correspondant à une valeur limite de Y pour laquelle la probabilité p_0 qu'il s'agisse d'un '0' bruité par l'IAM + le bruit gaussien, est égale à la probabilité p_1 qu'il s'agisse d'un '1' bruité par du bruit gaussien uniquement.

Avec la modélisation de la distribution de la variable globale de bruit que nous avons établi, on considère alors que :

- pour toute valeur Y supérieure à Y_L , les erreurs ont lieu sur les '0' émis à cause de la variable de bruit IAM + bruit gaussien (de variance $\sigma_T^2 = \sigma^2 + \sigma_I^2$, où σ^2 est la variance du bruit gaussien et σ_I^2 la variance équivalente de l'IAM)
- pour toute valeur Y inférieure à Y_L , les erreurs ont lieu sur les 1 émis à cause de la variable de bruit gaussien uniquement

On peut alors exprimer p_0 :

$$p_0 = p(x = 0|Y) = p(Y - n = 0|Y) = p(n = Y) \quad (\text{H.1})$$

On considère dans ce cas que les erreurs sont dues au bruit global (IAM + bruit gaussien), d'où :

$$p_0 = \frac{1}{\sigma_T \sqrt{2\pi}} e^{-\frac{1}{2\sigma_Y^2} y^2} \quad (\text{H.2})$$

De même pour p_1 :

$$p_1 = p(x = 1|Y) = p(Y - n = 1|Y) = p(n = Y - 1) \quad (\text{H.3})$$

On considère dans ce cas que les erreurs sont dues uniquement au bruit gaussien d'où :

$$p_1 = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2\sigma^2} (Y-1)^2} \quad (\text{H.4})$$

A partir de l'égalité des deux expressions précédentes, on détermine la valeur de Y_L correspondant à $p_0 = p_1$:

$$\frac{\sigma}{\sigma_T} = e^{-\frac{(Y_L-1)^2}{2\sigma^2}} \cdot e^{\frac{Y_L^2}{2\sigma_T^2}} \quad (\text{H.5})$$

D'où :

$$\frac{\sigma}{\sigma_T} = e^{-\frac{1}{2\sigma^2} (Y_L-1)^2} e^{\frac{1}{2\sigma_T^2} (Y_L+1)^2} \quad (\text{H.6})$$

En passant par le logarithme et en mettant au même dénominateur on obtient :

$$\ln\left(\frac{\sigma}{\sigma_T}\right) = \frac{\sigma^2 Y_L^2 - \sigma_T^2 Y_L^2 - \sigma_T^2 + 2\sigma_T^2 Y_L}{2\sigma^2 \sigma_T^2} \quad (\text{H.7})$$

En développant les termes et en remplaçant σ_T^2 par $\sigma^2 + \sigma_I^2$, il vient :

$$\sigma_I^2 Y_L^2 - 2Y_L \sigma_T^2 + \sigma_T^2 \left(1 + 2\sigma^2 \ln \left(\frac{\sigma}{\sigma_T}\right)\right) = 0 \quad (\text{H.8})$$

Cette équation se résoud comme une équation du second degré, par le calcul du discriminant :

$$\begin{aligned} \Delta &= 4\sigma_T^4 - 4\sigma_I^2 \sigma_T^2 \left(1 + 2\sigma^2 \ln \left(\frac{\sigma}{\sigma_T}\right)\right) \\ &= 4\sigma_T^2 \left(\sigma_T^2 - \sigma_I^2 \left(1 + 2\sigma^2 \ln \left(\frac{\sigma}{\sigma_T}\right)\right)\right) \end{aligned} \quad (\text{H.9})$$

Puis on détermine la racine qui est comprise entre 0 et 1 c'est-à-dire :

$$Y_L = \frac{2\sigma_T^2 - \sqrt{\Delta}}{2\sigma_I^2} \quad (\text{H.10})$$

Soit encore :

$$Y_L = \frac{2\sigma_T^2 - 2\sigma_T \sqrt{\sigma_T^2 - \sigma_I^2 \left(1 + 2\sigma^2 \ln \left(\frac{\sigma}{\sigma_T}\right)\right)}}{2\sigma_I^2} \quad (\text{H.11})$$

Cette expression se simplifie finalement en :

$$Y_L = 1 + \frac{\sigma^2}{\sigma_I^2} - \frac{\sigma \sigma_T}{\sigma_I^2} \sqrt{1 - 2\sigma_I^2 \ln \left(\frac{\sigma}{\sigma_T}\right)} \quad (\text{H.12})$$

Liste des Publications

Publications internationales dans des revues à comité de lecture

1. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Error Probability Upper Bound Considering Beat Noise in 2-D OCDMA Systems" dans *Journal of Communication Software and Systems (JCOMSS)*, vol.3, n° 4, Dec.2007, pp.266-274.
2. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Soft Decision LDPC Decoding over Chi-Square Based Optical Channels", *IEEE Journal of Lightwave Technology - Special issue on trends in signal processing for high capacity lightwave systems*, vol.27, n° 16, Août 2009, pp.3540-3545.
3. S.Sahuguède, D.Fafchamps, A.Julien-Vergonjanne, G.Rodriguez, J.P.Cances, P.Gallion, "LDPC Code Design and Performance Analysis on OOK Chi-square Based Optical channels", *IEEE Photonics Technology Letters*, vol.21, n° 17, Sept. 2009, pp.1190-1192.
4. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Impact of Optical CDMA channel statistic on soft decision FEC implementation", accepté en 2009 pour publication à *European Transactions on Telecommunications (ETT)*.

Communications internationales avec actes et comité de lecture

1. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Performance bound of worst beat noise interference scenario in 2-D incoherent OCDMA systems", présenté à *International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2007*, 27-29 Sept. 2007, pp.1-5

2. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Performance of Optical CDMA Systems with Low Density Parity Check Codes", présenté à *International Conference on Systems and Networks Communications, ICSNC 2007*, 25-31 Août 2007, pp.57-57
3. S.Sahuguède, M.Morelle, A.Julien-Vergonjanne, "Multi-Rate and QoS Differentiation in Optical Code Division Multiple Access Networks" présenté à *5th IEEE Consumer Communications and Networking Conference, CCNC 2008*, 10-12 Jan. 2008, pp.581-585
4. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Adapted LDPC Error Correction Scheme for 2D Optical CDMA Multimedia System", présenté à *IEEE 10th International Symposium on Spread Spectrum Techniques and Applications, ISSSTA '08*, 25-28 Août 2008, pp.775-779.
5. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "OCDMA channel statistic as a mixture of chi-square distributions", présenté à *The Fifth Advanced International Conference on Telecommunications, AICT 2009*, 24-28 Mai 2009. Best paper.
6. M.Morelle, S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Theoretical performance of Multi-Weight spreading codes for multimedia optical access network", présenté à *IEEE International Conference on Communications, ICC 2009*, 14-18 Juin 2009, pp.1-5.
7. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Beat Noise corrupted Optical CDMA channel statistic based on a mixture of chi-square model ", *Photonics in Switching, PS 2009*, Septembre 2009.

Communications nationales avec actes et comité de lecture

1. D.Fafchamps, S.Sahuguède, A.Julien-Vergonjanne, G.Rodriguez, J.P.Cances,

P.Gallion, J.M.Dumas, "Performances du décodage LDPC sur un anneau optique fonctionnant en mode paquet", présenté aux 27èmes Journées Nationales d'Optique Guidée, JNOG 2008, 20-22 Octobre 2008, pp.272-274.

2. S.Sahuguède, A.Julien-Vergonjanne, J.P.Cances, "Adaptation du décodage itératif des codes LDPC sur canal optique utilisant un modèle à statistique chi-2", *GRETSI 2009*, Septembre 2009.

Codage de canal pour les communications optiques

L'étude réalisée dans ce mémoire porte sur le codage correcteur d'erreurs utilisé pour compenser les différentes limitations d'un système de transmission optique sur fibre, notamment celles liées au processus quadratique de photo détection.

Dans le cas simple utilisateur, ce processus modifie la statistique du signal détecté qui s'apparente à une fonction chi-2. En tenant compte de cette statistique, le décodage itératif des codes LDPC a été adapté pour optimiser les performances.

Dans le cas d'un système incluant l'accès multiple par répartition de codes, principalement limité par l'interférence d'accès multiple, la photo détection engendre un bruit de battements, ou beat noise. L'étude de la statistique du signal détecté a permis d'établir un modèle théorique en mixture de chi-2. Les performances du décodage LDPC ont été optimisées en se basant sur ce modèle, permettant ainsi une application au cas d'une transmission multimédia.

Mots clés : Codes LDPC, décodage itératif, communications optiques filaires, photodétection, accès multiple par répartition de codes, beat noise

Forward Error Correction for Optical Communications

The work done in this study focuses on forward error correction used to compensate for performance degradations in fiber optical links, mainly corrupted by the quadratic photo detection process.

When we consider a single user transmission, this process has an impact on the photo detected signal statistic, which can be represented as a chi-square function. Taking into account this statistic, the iterative decoding used for LDPC codes is modified in order to optimise the performance.

In the case including Optical Code Division Multiple Access, mainly disrupted by Multiple Access Interference, photo detection induces beatings called beat noise. Studying the photo detected signal statistic leads us to establish a theoretical model based on a mixture of chi-square distribution. The error correction performance has been optimised by taking into account this model in the decoding process, which permits considering an application to multimedia transmissions.

Keywords : LDPC codes, iterative decoding, fiber optical communications, photodetection, code division multiple access, beat noise