

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Projet de fin d'étude

Pour l'obtention du diplôme de master II

Mention Électronique
Spécialité Réseaux et Télécommunications

présenté par :

Rhissa AG Dolla

Etude et simulation des protocoles ARQ et HARQ dans le cas du SISO SC-FDMA

Proposé par : Mahdi Bersali

Année Universitaire 2015-2016

Dédicaces

C'est avec profonde gratitude que Je dédie ce modeste travail de fin d'étude à mes très chers parents qui ont sacrifié le maximum de leur vie pour mon bonheur, ma réussite et pour mon éducation.

Je dédie également ce travail à mes frères et sœurs, mes familles, mes amis, tous mes professeurs et à tous ce qui me sont chers.

Remerciements

Je tiens à remercier dans un premier temps, toute l'équipe pédagogique du département d'électronique et surtout ceux chargé de la formation en télécommunications et réseaux.

Je tiens surtout à présenter la meilleure forme de mes remerciements pour mon respectueux encadreur monsieur Mahdi BERSALI, enseignant à l'USDB, qui a accepté de m'accueillir et de diriger notre projet en m'apportant tout le soutien technique et moral pour mener à bien mes recherches. Son énergie, sa rigueur, sa disponibilité ,ses remarques pertinentes et son encouragement m'ont permis de réaliser l'ensemble de mes travaux dans un dynamisme de réussite et dans la bonne humeur.

Je remercie aussi l'ensemble des personnes qui ont accepté d'évaluer mes travaux mais aussi l'ensemble des membres du jury de ma thèse.

Mes remerciements vont aussi à tous mes professeurs, enseignants et enfin à toutes les personnes qui m'ont soutenus et à celles qui ont contribué de près ou de loin à l'élaboration de ce travail.

Enfin, j'ai à remercier chaleureusement ma famille, qui m'a toujours encouragée dans mes études et m'a permis de devenir qui je suis.

ملخص:

أحادية SISO في سياق HARQ يتكون هذا البحث في تقنيات دراسة و محاكاة للكشف و الاستنباط تصحيح الأخطاء و أداء هذه التقنيات سوف يتم تقييم نقطة معدل الخطأ النظر بناء على إشارة إلى نسبة الضوضاء و من FDMA الناقل نقل .. خلال قناة متعدد المسارات مع إضافة الضوضاء المضافة الضبابي

كلمات المفاتيح: HARQ ; ARQ ; SC-FDMA ; CC-HARQ ; IR-HARQ

Résumé : ce travail de recherche consiste dans l'étude et la simulation des techniques de détections et de correction d'erreurs ARQ et HARQ dans le contexte d'une transmission SISO SCFDMA. Les performances de telles techniques seront évaluées du point de vue taux d'erreur binaire en fonction du rapport signal sur bruit et ce dans à travers un canal de type multi trajet avec ajout d'un bruit additif gaussien.

Mots clés : HARQ ; ARQ ; SC-FDMA ; CC-HARQ ; IR-HARQ

Abstract :

This research work consists in the study and simulation of the error detection and correction ARQ and HARQ in the context of a SISO SC FDMA transmission .Such technique performance will be evaluated from the perspective of bit error rate depending on the signal to noise ratio and this through a multipath channel type with an additive Gaussian noise.

Keywords : HARQ ; ARQ ; SC-FDMA ; CC-HARQ ; IR-HARQ

Table des matières

Listes des acronymes et abréviation

Liste des figures

Liste des tableaux

INTRODUCTION GENERALE.....1

CHAPITRE1: 1 les techniques de correction directe d'erreurs

1.1 Introduction.....3

1.2 Les codes en bloc.....5

1.2.1 Quelques définitions.....5

1.2.2 Quelques exemples.....7

a) Le contrôle de parité: le bit de parité7

b) Le contrôle de parité: parité longitudinale et transversale.....8

1.2.3 Les codes en blocs linéaires.....8

a) Définition.....8

b) Forme systématique d'un code.....9

c) Caractérisation d'un mot code.....10

d) Décodage par le syndrome10

e) Code de Hamming.....11

1.2.4 Codes polynomiaux.....12

a) Définition.....12

b) Principe du codage.....12

c) Principe du décodage.....13

1.2.5 Codes cycliques.....13

a) Définition.....14

b) Codes cycliques à polynômes irréductibles primitifs.....14

c) Capacités de correction.....14

1.2.6 Codes correcteurs performants.....14

a) Les codes BCH (Bose-Chaudhuri-Hocquenghem).....14

b) Les codes RS (Reed-Salomon).....14

1.3 Codes convolutifs (ou convolutionnels).....15

1.3.1	Généralités.....	15
1.3.2	Présentation du codeur.....	16
1.3.3	Analyse des Codes Convolutionnels.....	17
	a) Diagramme d'état.....	17
	b) Diagramme en Treillis.....	18
1.3.4	Décodage des codes convolutionnels.....	19
1.4	Entrelacement.....	20
1.5	Le turbo-code.....	21
	1.5.1 Codeur.....	22
	1.5.2 Décodeur.....	22
1.6	Les codes LDPC.....	23
	1.6.1 Historique	23
	1.6.2 Principe	23
1.7	Conclusion	25

CHAPITRE 2 : Techniques de détection et correction d'erreurs ARQ et HARQ

2.1	Introduction	26
2.2	Rappel sur la couche liaison de données	26
	2.2.1 Contrôle d'erreurs	27
	2.2.2 Contrôle du flux	28
2.3	Calcul du CRC	28
2.4	La technique ARQ.....	30
	2.4.1 Principe de l'ARQ	30
	a) Protocole Stop-and-Wait	31
	b) Protocole Go-back-N	33
	c) Protocole Selective-Repeat.....	35
	d) Protocole Stop-and-Wait parallèle.....	36
	2.4.2 Remarques sur la gestion des files d'attente	37
2.5	La technique HARQ	38
	a) HARQ type I.....	39
	b) HARQ type II	39

c) HARQ type III	42
2.6 Conclusion	43

CHAPITRE 3 : Domaines d'applications du protocole HARQ

3.1 Introduction	44
3.2 La technologie HARQ en HSPA.....	44
3.2.1 Mécanisme de retransmission hybride HARQ en HSDPA	46
3.2.2 La technique HARQ en HSUPA	48
3.3 Le HARQ dans la LTE.....	48
3.3.1 Rappel sur la couche MAC du LTE.....	48
a) Ordonnancement.....	49
b) La technique de retransmission HARQ en LTE.....	49
3.3.2 La couche physique du LTE.....	52
3.3.3 Les canaux logiques du LTE avec HARQ.....	53
a) Les canaux physiques	53
b) Les canaux de transport	54
3.4 La technologie HARQ dans le Wimax	55
3.5 La technique d'accès au canal SC-FDMA	57
3.5.1 Rappel sur la technique Single Carrier FDE : SC/FDE.....	57
3.5.2 Principe du SC-FDMA.....	58
3.5.3 Transmission des signaux en SC-FDMA.....	59
3.5.4 Allocations des sous porteuses.....	61
3.5.5 Avantages de la SC-FDMA par rapport à l'OFDMA.....	63
3.6 Conclusion.....	65

CHAPITRE 4 : résultats et interprétations des simulations: applications des techniques ARQ et HARQ en SC-FDMA

4.1 Introduction.....	66
4.1.1 Paramètres de simulation	66
4.2 Les différents cas de simulation.....	67
4.2.1 Simulations sans codage canal	67
4.2.2 Simulations avec codage canal	74

4.3 Conclusion79

CONCLUSION

GENERALE.....80Annexe.....

.....Bibliographie...

.....

Listes des acronymes et abréviations

A

ARQ: Automatic Repeat-request

ACK: ACKnowledgement

AMC: Adaptive Modulation and Coding

B

BCH: Bose-Chaudhuri-Hocquenghem

BER: Bit Error Rate

BPSK: Binary Phase Shift Keying

BLER: Block-Error Rate

C

CRC: Cyclic Redundancy Check

CDS: Channel-Dependent Scheduling

BCCH: Broadcast Control Channel

CCCH: Common Control Channel

CDMA: Code Division Multiple Access

CP: Cyclic Prefix

CQI: Channel Quality Indicator

CQICH: CQI Channel

CA: Carrier Aggregation

D

DFT: Discrete Fourier Transform

DL: Down Link

DTCH:Dedicated Traffic CHannel

DCI: DL Control Indicator

DL-SCH: Downlink Shared Channel

DCCH:Dedicated Control CHannel

E

eNB: evolved NodeB

E-HICH: Enhanced Uplink HARQ Indicator Channel

ED: Error Detection

E-UTRAN: Evolved-UTRAN

F

FEC: Forward Error Correction

FDMA: Frequency Division Multiple Access

FFT: Fast Fourier Transform

FDD: Frequency Division Duplex

G

GBN: Go-Back-N

3G: 3rd Generation

4G: 4th Generation

3GPP: 3rd Generation Partnership Project

H

HS-DSCH: High-Speed Downlink Shared Channel

HSDPA: High Speed Downlink Packet Access

HSUPA: High Speed Uplink Packet Access

HSPA: High Speed Packet Access

HARQ: Hybrid Automatic Repeat reQuest

HARQ-IR: HARQ Incremental Redundancy

HARQ -CC:HARQ Chase Combining

HI: HARQ Indicator

I

IMT-2000 International Mobile Telephony

ITU International Telecommunication Union

ISI: Inter Symbol Interference

IDFT: Inverse DFT

IFDMA: Interleaved Frequency Division Multiple Access

IEEE:Institute of Electrical and Electronics Engineers

IES: interférences entre-symboles

L

LTE: Long Term Evolution

LFDMA: Localized Frequency Division Multiple Access

LDPC: Low Density Parity Check

M

MRC: Maximum Ratio Combining

MAC: Media Access Control

MIMO: Multiple Input Multiple Output

MSC: Mobile Switching Center

MCS: Modulation and Coding Schemes

MMS: Multimedia Messaging Service

MS: Mobile Station

MMSE: Minimum Mean Square Error

N

NACK: Negative ACKnowledgement

NT: Nombre de Trames

O

OFDMA: Orthogonal Frequency Division Multiple Access

OFDM: Orthogonal Frequency Division Multiplexing

OVSF: Orthogonal Variable Spreading Factor

P

PDU: Packet Data Unit

PAPR: Peak-to- Average Power Ratio

PDCCH: Physical Downlink Control Channel

PCH: Paging CHannel

PHICH: Physical Hybrid ARQ Indicator Channel

PUSCH: Uplink Shared CHannel

PUCCH: Physical Uplink Control Channel

UL-SCH: Uplink Shared CHannel

PDCCH: Physical Downlink Control Channel

PUCCH: Physical Uplink Control Channel

PMI/RI:Pre-coding Matrix Indicator, and Rank Indicator

Q

QAM: Quadrature Amplitude Modulation

QoS: Quality of Service

QPSK: Quadrature Phase Shift Keying

R

RV: redundancy Version

R99: Release 99

RLC: Radio Link Control

RNC: Radio Network Controller

RLC: Radio Link Control

RNC: Radio Network Controller

RNS: Radio Network Sub-system

RRC: Radio Resource Control

RS: Reed-Solomon

RSC: Recursive Systematic Code

Rx: Receiver

S

SC-FDMA: Single Carrier FDMA

SC/FDE: Single Carrier with Frequency Domain Equalization

SF: Spreading Factor

SGSN: Serving GPRS Support Node

SIR: Signal to Interference Ratio

SNR: Signal to Noise Ratio

SC-HARQ : Soft Combining-HARQ

SPID : Sous-Paquet Identifier

SINR: Signal-to-Interference-and-Noise Ratio

S-to-P: Transformation Série-Parallèle,

SISO : Single Input Single Output

T

TDMA: Time Division Multiple Access

Tx: Transmitter

TTI: Transmission Time Interval

TDD: Time-Division Duplex

U

UL: Uplink

UL-SCH:Uplink Shared CHannel

UCI: UL Control Indicator

UE: User Equipment

UMTS: Universal Mobile Telecommunications System

UTRAN: UMTS Terrestrial Radio

W

Wimax:Worldwide interoperability for Microwave Access

Z

ZF: Zéro Forcing

Liste des figures

Figure 1.1 : schéma de la transmission avec canal bruité.....	4
Figure 1.2: principe du mot de code.....	5
Figure 1.3: exemple de codeur convolutif.....	15
Figure 1.4 : diagramme des états de transition du codeur de la figure 1.3.....	16
Figure 1.5 : codeur (2, 1, 3).....	17
Figure 1.6 :diagramme d'état du codeur (2, 1, 3).....	18
Figure 1.7 : diagramme en treillis du codeur (2, 1, 3).....	19
Figure 1.8 : schéma de principe d'un turbo-codeur.....	22
Figure 1.9 : schéma de principe d'un turbo-décodeur.....	23
Figure 1.10 : Graphe bipartite d'un code LDPC.....	24
Figure 2.1 :Principe de contrôle d'erreur par ARQ.....	27
Figure 2.2 : fonctionnement d'un protocole de liaison.....	31
Figure 2.3 : protocole Send and Wait (V (S) (numéro de séquence du prochain paquet à transmettre. V (R) : numéro de séquence du paquet attendu).....	32
Figure 2.4 : le principe d'acquittement lors du protocole Stop-and-Wait.....	32
Figure 2.5 : protocole Go-Back-N avec une fenêtre d'anticipation de taille 3.....	34
Figure 2.6 : principe des acquittements dans le protocole Go-Back-N.....	35
Figure 2.7 : principe d'acquittement dans le protocole Selective-Repeat.....	36

Figure 2.8 : protocole ARQ Stop-and-Wait parallèle avec un nombre de $w = 3$ processus Stop-and-Wait parallèle.....	37
Figure 2.9 : bloc du diagramme du CC- HARQ	40
Figure 2.10 : HARQ- IR protocole de retransmission pour $N_{max} = 3$. Chaque paquet réémis, la redondance est ajouté aux paquets précédemment erronés.....	42
Figure 3.1 :Les principales mises à jour (release) de la 3GPP.....	46
Figure 3.2 : technique HARQ dans la chaîne de codage du canal HS-DSCH.....	47
Figure 3.3 : principe de fonctionnement de l'HARQ.....	48
Figure 3.4 : le diagramme en blocs de transmission utilisant le HARQ.....	51
Figure 3.5 : le diagramme d'interaction pour HARQ-ARQ.....	52
Figure 3.6 :Schéma de principe d'une transmission SC FDE.....	58
Figure 3.7 : Principe des sous-porteuses orthogonales.....	59
Figure 3.8 : Emetteur SC-FDMA	60
Figure 3.9 : Récepteur SC-FDMA	60
Figure 3.10 :Mappage des sous porteuses avec $N=9$, $M= 3$ et 3 utilisateurs.....	62
Figure 3.11 : Mappage des sous porteuses avec $N=9$ et $M= 3$ et un seul utilisateur.....	62
Figure 3.12 :Similitude entre une chaine OFDMA et SC-FDMA	63
Figure 3.13 :Comparaison entre la détection en SC-FDMA et OFDMA	64
Figure 4.1 : exemple d'application du protocole ARQ en transmission SISO SC- FDMA : Taille de la trame $T=512$ bit, nombre de trames envoyées $NT=16$; $SNR=6$ dB et seuil d'erreurs fixé à seuil=80 bits (erronés) par trame ($BER =0,1564$).....	68
Figure 4.2 : exemple d'application du protocole ARQ en transmission SISO SC- FDMA: $T =512$ bit, $NT=16$; $SNR=6$ dB et seuil= 40 bits (erronés) par trame ($BER =0,07812$).....	70
Figure 4.3 : exemple d'application du protocole ARQ en transmission SISO SC- FDMA : $T =512$ bit, $NT=16$; $SNR=6$ dB et seuil = 80 bits (erronés) par trame ($BER=0,1564$).....	72

Figure 4.4 : exemple d'application du protocole ARQ en transmission SISO SC- FDMA.
T =512 bit, NT=16, SNR=6 dB et seuil= 40 bits (erronés) par trame (BER =0,07812).....73

Figure 4.5 : exemple d'application du protocole ARQ en transmission SISO SC- FDMA : T
=512 bit, NT=16, SNR=6 dB et seuil= 80 bits (erronés) par trame (BER =0,07812).....75

Figure 4.6 : exemple d'application du protocole ARQ en transmission SISO SC- FDMA avec
T =512 bit, NT=16, SNR=6 dB et seuil= 40bits (erronés) par trame (BER
=0,07812).....76

Figure 4.7 : Exemple d'application du protocole ARQ en transmission SISO SC- FDMA avec
T =512 bit, NT=16, SNR=6 dB et seuil= 80 bits (erronés) par trame (BER
=0,07812).....77

Figure 4.8 : Exemple d'application du protocole ARQ en transmission SISO SC- FDMA. T
=512 bit, NT=16, SNR=6 dB et seuil= 40bits (erronés) par trame (BER =0,07812).....78

Liste des tableaux

Tableau 1.1. Exemple de contrôle de parité	7
Tableau 2.1. Liste des polynômes générateurs pour chaque degré de CRC	30

Annexe

Un aperçu sur le programme de nos simulations :

```
clc
clearall
M=16;%taille de la fft
N=64;%taille de la dft
f0=0;% mapping en mode localisé à partir de la première sous porteuse
NbSymbc=496;% taille de la trame
m=4;% concerne la constellation
%codage canal (convolutif)
trellis = poly2trellis(3,[4,5,7]); %R=1/3
tblen=18;% concerne le decodage canal
init_state=0;% idem
%rapport signal sur bruit
SNR = (0:2:20); % SNR en dB dB
SNR_lin = 10.^(SNR/10); % SNR linéaire
sigma2_n = 1./SNR_lin; % Variance du SNR
%paramètre du canal de propagation
Nch = 4; % nombre de trajets
F_att = 2; %facteur de la décroissance exponentielle du profile des retards
% calcul des variancex des trajets
chan_var = exp(-(0:Nch-1)/F_att);
chan_var = chan_var/sum(chan_var); % normalisation des puissance du trajets
max_buffer=1000;
%initialisations
berVec1 = zeros(max_buffer, length(SNR));
Rx_symb=zeros(NbSymbc/4,2);
Buffer=[ ];
Chan=[ ];
p=0;
GberVec1=[ ];
num_err= 0;
Num_err=[ ];
n=1;
```

```

for k=1:max_buffer
data = randi( [0 1],1,NbSymc);
l=length(data);
Buffer=[Buffer data];
Tx_data=Buffer(l*(p+1)-l+1: l*(p+1));% trame à transmettre ou à retransmettre
display('*****')
display(['increment of the buffer is ' num2str( k) ''])
display('*****')
display(['the length of the buffer is ' num2str(length(Buffer)) ''])
display('*****')
display('*****')
display(['numerous of sended frame is ' num2str( p+1) ''])

chan = sqrt(0.5)*( randn(1,Nch) + 1i*randn(1,Nch) ) .* sqrt(chan_var);

% codage de la trame à l'aide du CRC
hGen = comm.CRCGenerator([1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 1], 'ChecksumsPerFrame',1);
codeword = step(hGen, Tx_data) ;
[encoded_data, fin_state] = convenc(codeword,trellis,init_state);
% modulation QAM
QAM_MOD=modem.qammod('M',m,'SymbolOrder','gray','InputType','bit');
symb=(1/sqrt(2))*modulate(QAM_MOD,encoded_data );%384 (16 QAM)
Chan=[Chan; chan ];

%sous programme: test des erreurs
[err_thr]=error_test(n);
[n SNR(n) err_thr ]% indicateur
%sous programme: algorithme SC FDMA
x=[];
Rx_symb=[];
Rx_symb=scfdma(symb,Rx_symb,M,N,f0,chan,Nch,SNR,sigma2_n,n);
%demodulation etdecodage canal
QAM_DEMOD=modem.qamdemod('M',m,'SymbolOrder','gray','OutputType','bit');
Rx_data=demodulate(QAM_DEMOD,Rx_symb.);%% 1536
decoded_data= vitdec(Rx_data,trellis,tblen,'trunc','hard');
%decoded_data=Rx_data;
hDetect = comm.CRCDetector([1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 1], 'ChecksumsPerFrame',1);
[tx1, err1] = step(hDetect, decoded_data);
[num_err berVec1(k,n)] = biterr(tx1',Tx_data);
Num_err=[Num_err num_err]
GberVec1=[GberVec1 berVec1(k,n)]
% sous programme: algorithme ARQ et HARQ
[p,GberVec1,n]=arq_test(p,k,Num_err,err_thr,GberVec1,n);
pause;
end

```

Bibliographie

- [1] Yoann Morel : communication numérique, Codes correcteurs d'erreurs
- [2] Claude Carlet : Cours de Codes Correcteurs d'Erreurs (et fonctions booléennes), D.E.A de mathématiques et d'informatique de Bamako, 2007.
- [3] Laurent MURA : Introduction aux CCE Application VHDL du code de Hamming, IUT de Colmar – Département Réseaux et Télécommunications.
- [4] Yoann Morel: Codes correcteurs d'erreurs, communication numérique.
- [5] Jean-Yves Antoine : Codage : codes polynomiaux et cycliques (Sécurité des Réseaux), LI-Université François Rabelais de Tour.
- [6] K. Lahèche : Les codes en informatique : codes détecteurs et correcteurs, Hermès, 1995.
- [7] BERROU C : « Les turbo codes convolutifs », Université de Marne-la-vallée, 14/04/99.
- [8] Hadj-Saïd Naima, Ali-Pacha Adda, Belgoraf A, Belmekki B: Codage canal : codes correcteurs d'erreurs, Université des Sciences et de la technologie d'Oran – USTO-Mohammed BOUDIAF, 2004.
- [9] Irina Adjudeanu: Codes correcteurs d'erreurs LDPC structurés, UNIVERSITÉ LAVAL, page 7, 2010.

- [10] I. DIOP, S. M. FARSSI, M. CHAUMONT, O. KHOUMA, H. B. DIOUF : Utilisation des codes LDPC en sténographie, page 2.
- [11] Moustapha El Aoun : Optimisation des techniques de codage pour les transmissions radio avec voie de retour, page 17-18-44, Sep 2012.
- [12] Nabih ALAOUI : coopérative Communications In Mobile Ad hoc Networks, le 05 Décembre 2013.
- [13] Claude Duvallet : TD 3 - Automatic Repeat reQuest (ARQ), 2008-2009 (<http://litis.univlehavre.fr/~duvallet/enseignements/Cours/LPRODA2I/UF9/LPRODA2I-TD3-UF9.pdf>).
- [14] Marceau Coupechoux, Philippe Godlewski, Philippe Martins : algorithmes de reprises sur erreurs (ARQ), Télécom Paris Tech, page : 9-81-82, 10/03/2015.
- [15] Esen : chapitre III : les protocoles de liaisons de données, université de la Manouba, 01/02/15.
- [16] Nabih ALAOUI: Cooperative Communications in Mobile Ad hoc Networks, UNIVERSITE DE LIMOGES, page: 49-51, 05 December 2013.
- [17] Tumula V. K. Chaitanya: HARQ Systems: Resource allocation, feedback error protection, and bits-to-symbol mappings, page 11, 2013.
- [18] Redouane Sassioui : analyse et amélioration de la combinaison AMC-HARQ ,2015.
- [19] : Imene BEN SALEM : adaptation des taux et des puissances de transmission pour les schémas Incremental Redundancy HARQ tronqués, page : 6.
- [20] Aata El Hamss, Adaptation de la stratégie de retransmission dans le protocole HARQ, page : 18-19, 2014.
- [21] Mohamad MAAZ : Allocation de ressource et analyse des critères de performance dans les réseaux cellulaires coopératifs, INSA Rennes, page : 30, 03.12.2013.
- [22] Yvon Sosthène Yameogo : études de nouvelles techniques d'estimation et d'égalisation de canal adaptées au système SC-FDMA, université de Rennes 1, 2011.

- [23] BRAHIMI Zouheira, ZELLOUMA Hadjra :Effet d'HPA sur le système SC-FDMA, 2014-2015.
- [24] Omar Fariss: rapport_sc-fdma,academia.edu, 2014
(http://www.academia.edu/7210579/Rapport_sc-fdma).
- [25] Didace ,Iskill KETA-WAPOUTOU et MOUSSE : 3GPP LTE interface radio long termevolution: single carrier FDMA, ENST Paris,2007.
- [26] d1n7iqsz6ob2ad.cloudfront.net/document/pdf/5385b8fdde38c.pdf
- [27] Abdelhakim Youcef : amélioration des performances du traitement d'interférences dans le domaine fréquentiel pour les communications acoustiques sous-marines, télécom Bretagne,12 décembre 2013.
- [28] <https://fr.scribd.com/doc/30356227/3G-LTE>
- [29] Melle BOUCHENTOUF Hadjer ,Mr BOUDGHENE STAMBOULI Riyad : ETUDE DES PERFORMANCES DES RESEAUX 4G (LTE), 2012 – 2013.
- [30] HibaMOUACHI ,Idriss KNADEL, Kawtar ZERHOUNI : étude et simulation de la norme LTE par 3GPP,ENSA Marrakeck,19/06/2012
(<https://fr.scribd.com/doc/131820454/Simulation-OFDM-sous-simulink-et-etude-de-la-norme-LTE>).
- [31] OuldMedeni Mohamed Bouye : application des codes correcteurs d'erreurs en sténographie,page 21-22 , 11 juillet 2012.
- [32] Raphaël Le Bidan, Xavier Lagrange : Optimisation des techniques de codage pour les transmissions radio avec voie de retour.
- [33] J-C ROLIN G Eiffel Dijon: transmission de l'information sous forme numérique ,02/2008
- [34] C. Pham : liaison logique, RESO-LIP/INRIA Université Lyon 1(<http://www.ens-lyon.fr/~cpham>)
- [35] EL HAJJ Paul, DAHBI Nabil : étude de la technologie Wimax mobile, télécom sud Paris, 20/01/2010.

- [36] AMAZIT ABDELGHANI : IMPACT DES INTERFÉRENCES DE LA COUCHE PHYSIQUE SUR LA COUCHE MAC DANS LA TECHNOLOGIE LTE, JUIN 2011.
- [37] Sachin K.R , Nikhil SrinathBetgov , Nisarga C , Apeksha S.K , Smt. Usha M.S: A Review of Hybrid ARQ in 4G LTE, Department of Computer Science and Engineering, 2015
- [38] Mark Stambaugh : Agilent Technologies | *Mobile Dev Conception, 10 septembre 2008*
(<http://mobiledevdesign.com/standards/harq-process-boosts-lte-communications>)
- [39] AMAZIT ABDELGHANI : IMPACT DES INTERFÉRENCES DE LA COUCHE PHYSIQUE SUR LA COUCHE MAC DANS LA TECHNOLOGIE LTE, UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES, JUIN 2011.
- [40] Yannick Bouguen, Eric Hardouin, François-Xavier Wolf : LTE et les réseaux 4G, groupe Eyrolles, 2012.
- [41] Roger Piqueras Jover : LTE PHY Fundamentals
(http://www.ee.columbia.edu/~roger/LTE_PHY_fundamentals.pdf)
- [42] Marceau Coupechoux : Long Term Evolution PHY, MAC et RLC, Telecom ParisTech, Décembre 2014
- [43] OMAR RACHID HADJAR : ANALYSE, IMPLÉMENTATION ET ÉVALUATION DE PERFORMANCE DE LA FUTURE MÉTHODE D'ACCÈS HSDPA, UNIVERSITÉ LAVAL QUÉBEC, 2006.
- [44] Erik Dahlman, Stefan Parkvall, Johan Skold et Per Beming. 3G Evolution: HSPA and LTE for Mobile Broadband. Academic Press Inc, 2007.
- [45] ZdenekBecvar ,Pavel Mach: On Efficiency of ARQ and HARQ Entities Interaction in Wimax Networks, Czech Technical University in Prague.
- [46]. Wireless and Mobile Communication (WMC) Group Faculty of Electrical Engineering, Mathematics and Computer Science Delft University of Technology.
- [47] GUY PUJOLLE : les réseaux, EYROLLES, page 109, édition 2014
- [48] Dr.Joern Krause: future 3GPP RAN standardization activities for LTE, 24.10.2012.

Introduction générale

Depuis quelques années, les moyens de communications sans fil sont devenus très populaires puisqu'ils permettent aux utilisateurs de communiquer les uns avec les autres indépendamment de l'endroit où ils se situent. De plus, la demande pour des hauts débits ne cesse d'augmenter. Dans les systèmes de transmission radio actuels, qu'ils soient fixes ou mobiles, la transmission des données se fait généralement en mode paquet.

Au cours de l'échange de données dans des réseaux sans fil, des erreurs de transmission se produisent. Les données corrompues ne peuvent pas être traitées ultérieurement sans correction. Les techniques basées sur la demande automatique de répétition (ARQ) ou correction directe des erreurs (FEC) sont classiquement utilisées pour réparer des données erronées dans les systèmes de transmission. Lorsque l'on en dispose d'une voie de retour (*feedback*), nécessaire pour informer l'émetteur de la qualité de l'information reçue, on peut envisager de coupler les codes correcteurs d'erreur à un mécanisme de retransmission des paquets pour améliorer la robustesse du système vis à vis des fluctuations de puissance sur le canal afin d'éviter le gaspillage des ressources lors des retransmissions. En fait, la protection des données des utilisateurs par un code FEC permet de minimiser le nombre des retransmissions, tandis qu'une stratégie ARQ permet de limiter la redondance.

Une technique, appelée Hybrid ARQ, rencontre aujourd'hui un succès notable car elle permet d'adapter le codage (voire la modulation) et donc le débit utile de la transmission aux conditions de propagation [32]. Cette technique combine les techniques ARQ, qui consistent en la retransmission des paquets de données sur détection d'une erreur, avec un codage canal ou FEC réalisant ainsi le meilleur compromis entre les deux procédés. Ce schéma puissant permet donc de tirer profit, en fonction du SNR, des propriétés de l'ARQ et celles de FEC et ce pour fiabiliser le lien entre l'émetteur et le récepteur par la retransmission de paquets ou de

redondance supplémentaire si l'information n'a pas été correctement reçue au niveau du récepteur. Ces schémas HARQ font aujourd'hui partie des nouvelles normes des transmissions sans fil tels que le Wimax (IEEE 802.16), le HSPA (HSDPA et HSUPA) et la 3GPP-LTE. Ainsi il est d'un grand intérêt d'étudier les systèmes basés sur les HARQ.

Le présent mémoire est constitué de six parties : la première partie comprend une introduction qui décrit d'une façon générale la technologie HARQ et son intérêt dans les systèmes sans fil. La deuxième partie est une description des techniques de détection et correction d'erreurs directes de transmission FEC ou codage canal. Durant la troisième partie, nous détaillerons le fonctionnement des protocoles d'acquiescement ARQ et spécialement des mécanismes HARQ. La quatrième partie est consacrée à une brève étude des domaines d'implémentation de la technologie HARQ tel que le Wimax, HSPA, et la LTE, ainsi qu'un aperçu sur la technique d'accès au canal SC-FDMA utilisé dans la voie montante (*uplink*) de la norme de téléphonie mobile 4G LTE. Quant à la cinquième partie les résultats et interprétations seront abordés. Finalement, le mémoire se termine par une conclusion qui résumera notre travail.

Chapitre 1 les techniques de correction directe d'erreurs

1.1 Introduction

Par définition un code correcteur d'erreurs est une technique de codage basée sur la redondance de l'information, destinée à détecter, voire corriger, les erreurs qui peuvent-être générées par un canal de communication peu fiable [1].

Le codage correcteur d'erreurs, dont l'origine remonte à la fin des années 40 [2], permet de transmettre de façon fiable l'information, codée au moyen de mots binaires d'une longueur donnée sur un canal de communication plus ou moins bruité. Dans ces schémas FEC l'émetteur ajoute des informations redondantes au message utile et le récepteur exploite cette redondance pour décider quel message a été envoyé. On les classe en trois grandes familles de codes :

- les codes en blocs : le codage/décodage d'un bloc dépend uniquement des informations de ce bloc.
- les codes convolutifs (ou récurrents) : le codage/décodage d'un bloc dépend des informations d'autres blocs (généralement de blocs précédemment transmis).
- les codes approchant la capacité de Shannon : le turbo codes et les codes LDPC.

Voyons schématiquement le problème posé par la transmission sur un canal bruité.

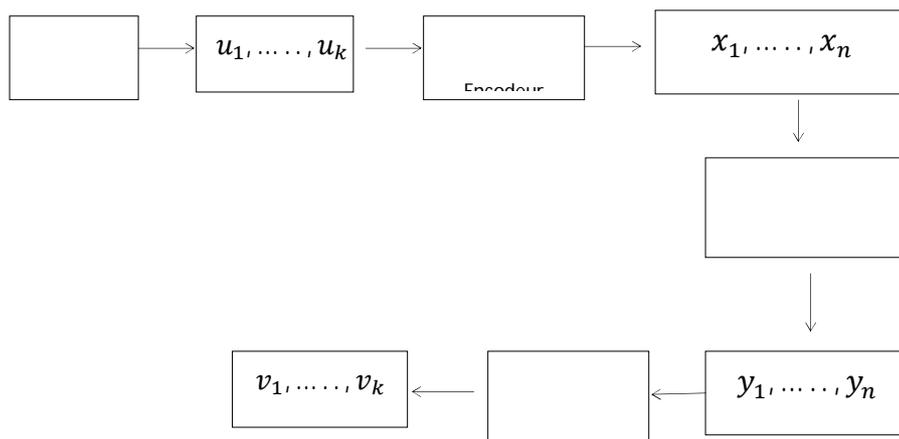


Figure 1.1. Schéma de la transmission avec canal bruité [2].

L'encodeur transforme le mot transmis en un mot de code. Nous appelons taux de transmission le rapport k/n . La valeur r telle que

$$n = k + r \tag{1.1}$$

est appelée redondance.

Pouvoir détecter une erreur, c'est pouvoir répondre à la question suivante [2] :

Dans le cas du canal bruité de la figure 1.1, certains symboles peuvent être modifiés. Les vecteurs (y_1, \dots, y_n) et (x_1, \dots, x_n) sont-ils égaux ?

Pouvoir corriger cette erreur c'est pouvoir, après détection, obtenir par décodage:

$$(v_1, \dots, v_k) = (u_1, \dots, u_k) \tag{1.2}$$

1.2 Les codes en bloc

Des codages par blocs sont représentés par des nombreux codages détecteurs/correcteurs d'erreurs. L'idée consiste à coder non pas les bits individuellement, mais à les coder par blocs d'une certaine longueur fixée. L'information est ainsi découpée en blocs tous de même longueur, notons cette information k , ensuite chacun des blocs ainsi obtenus est codé séparément. Dans le but de pouvoir détecter et corriger des erreurs de transmission, les techniques de codages augmentent toujours la longueur des mots en ajoutant des bits qui introduisent de la redondance.

Le codeur $C(n, k)$ introduit de la redondance :

$$r = n - k \quad (1.3)$$

Dont n représentent les bits à transmettre et k la redondance. Les r bits rajoutés ne dépendent que des k bits d'information.

1.2.1 Quelques définitions [3]

- Un code (k, n) transforme (code) tout bloc initial de k bits d'information en un bloc codé de n bits. Le code introduit une redondance puisque $n \geq k$. Le code est systématique si les k premiers bits du bloc codé sont égaux aux bits du bloc initial. Alors les $r = n - k$ derniers bits forment un champ de contrôle d'erreur.
- On appelle mot du code, la suite de n bits obtenue après un codage $C(k, n)$. Le nombre n de bits qui composent un mot du code est appelé la longueur du code. La dimension k étant la longueur initiale des mots. La figure ci-après illustre le principe.

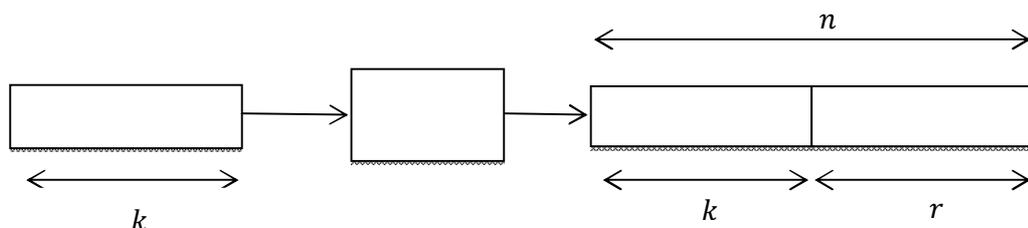


Figure 1.2 : principe du mot de code [3].

Exemple: mot initial = [1 0 1 1 1], mot codé = [1 0 1 1 1 1 0] avec $R = 5/7$.

- Le rendement d'un code (k, n) est défini par :

$$R = k/n(1.4)$$

La protection est dite forte si R tend vers 0, et faible si R tend vers 1.

- Le poids de Hamming noté p d'un mot est le nombre de bits à 1 qu'il contient: exemple:
 $p(0110000) = 2$.
- La distance de Hamming notée d entre deux mots de même longueur est définie par le nombre de positions binaires qui diffèrent entre ces deux mots. On l'obtient par le poids de Hamming de l'opération OU exclusif binaire des 2 mots.

Exemple : $d(1100110, 1010110) = p(1100110 \text{ XOR } 1010110) = p(0110000) = 2$

La distance de Hamming notée dH d'un code est la distance minimum entre tous les mots du code.

La capacité de détection (de correction) d'un code est définie par les configurations erronées qu'il est capable de détecter (corriger). Une erreur simple (respectivement double, ou d'ordre p) affecte une seule (respectivement 2, ou p) position(s) binaire(s) d'un mot. Pour qu'un code ait une capacité de détection (respectivement correction) des erreurs d'ordre e , il faut que sa distance de Hamming soit supérieure à $1 + e$ (respectivement $1 + 2e$), ou encore:

- o Nombre d'erreurs détectées est
 $(dH - 1)$ (1.5)

- o Nombre d'erreurs corrigées est
 $((dH - 1) / 2)$ (1.6)

Exemple : distance $dH = 3$ implique capacité de détection ≤ 2 , capacité de correction ≤ 1 .

1.2.2 Quelques exemples

a. *Le contrôle de parité: le bit de parité*

Il existe deux types de contrôle de parité (pair et impair) et il est indispensable que l'émetteur et le récepteur s'entendent sur le type à utiliser pour l'ensemble de la transmission.

Parité paire : Lorsque le nombre de '1' dans les données envoyées est impair, le bit de parité (bit de contrôle) est placé à '1' de manière à ce que le nombre total de '1' soit pair y compris le bit de parité. Dans le cas contraire, le bit de parité est placé à 0 [33]

Parité impaire : Elle correspond au système inverse.

Quelle que soit la parité choisie, si un bit est modifié au cours de la transmission, les calculs de parité effectués par l'émetteur et par le récepteur différeront.

Le poids de Hamming des mots du code est paire (impaire). C'est un code systématique $(k, k + 1)$ dans lequel un bit (le bit de parité) est ajouté au mot initial pour assurer la parité. Son rendement est faible lorsque k est petit. Exemple : transmission de caractères utilisant le code ASCII (voir tableau 1.1)

Lettre	Code ASCII	Mot codé (parité paire)	Mode codé (parité impaire)
E	1 0 1 0 0 0 1	1 0 1 0 0 0 1 1	1 0 1 0 0 0 1 0
V	0 1 1 0 1 0 1	0 1 1 0 1 0 1 0	0 1 1 0 1 0 1 1
A	1 0 0 0 0 0 1	1 0 0 0 0 0 1 0	1 0 0 0 0 0 1 1

Tableau 1.1. Exemple de contrôle de parité [3].

L'émetteur transmet alors le message contrôlé puis le récepteur vérifie la parité du message reçu. Si lors de la transmission une erreur survient, la parité du message reçu est incorrecte et le récepteur détecte l'existence d'une erreur sans pour autant être capable de la localiser. Il demande alors une nouvelle émission du message. Et si lors de la transmission deux erreurs surviennent, la parité du message reçu est correcte, les erreurs ne sont pas décelées [3].

b. Le contrôle de parité: parité longitudinale et transversale

Le bloc de données est disposé sous une forme matricielle ($k = a \cdot b$). On applique la parité (uniquement paire) sur chaque ligne et chaque colonne. On obtient une matrice $(a + 1, b + 1)$. Avec un rendement très faible :

$$R = a \cdot b / (a + 1) \cdot (b + 1) \tag{1.7}$$

Une erreur simple modifie simultanément la parité d'une ligne et d'une colonne. La correction se fait en inversant le bit situé à l'intersection de la ligne et de la colonne ayant une parité incorrecte [3].

1.2.3 Les codes en blocs linéaires

a. Définition

Code linéaire [4] : si c_1 et c_2 désignent les mots codes respectifs des deux suites d'information de k bits d_1 et d_2 , alors à la suite d'information de k bits $d_1 \oplus d_2$ est associé le mot code $c_1 \oplus c_2$. Où \oplus est l'addition bit à bit modulo 2. Dans un code en bloc, les n éléments binaires des mots code sont calculés uniquement avec les k bits d'information du bloc courant. Au besoin, le message initial est découpé en paquets, ou blocs, de taille k . Pour un code en bloc linéaire, le mot code c s'obtient à partir du mot d'information d par une expression matricielle de la forme :

$$c = d * G \tag{1.8}$$

où, d est un vecteur ligne de taille $1 \times k$, c est un vecteur ligne de taille $1 \times n$ et G est une matrice de taille $k \times n$ appelée matrice génératrice du code.

Exemple code de parité :

Le taux d'information est $r = (n - 1)/n$, et sa matrice génératrice, de taille $(n - 1) \times n$ est :

$$G = \begin{bmatrix} & & & 1 \\ I_{n-1} & & & \\ & & & 1 \end{bmatrix} \tag{1.9}$$

avec I_{n-1} la matrice identité de taille $n - 1$. Par exemple pour $n = 3$, on a un code (4, 3,2), de matrice génératrice G tel que

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \quad (1.10)$$

Et par exemple le mot $u = [1 \ 0 \ 1]$ est codé par :

$$c = u * G = [1 \ 0 \ 1 \ 2] = [1 \ 0 \ 1 \ 0][2] \quad (1.11)$$

b. Forme systématique d'un code [4]

Un code est entièrement déterminé par sa matrice génératrice G . Les vecteurs-ligne de G sont eux-mêmes des mots-code, et sont supposés linéairement indépendants. Cela revient à imposer que :

Si $d_1 \neq d_2$, alors

$$(c_1 = d_1 G) \neq (c_2 = d_2 G) \quad (1.12)$$

des mots d'information différents ne peuvent pas être identiquement codés. En d'autres termes, la matrice G est de rang k . On montre de plus que les combinaisons linéaires sur les lignes de G ainsi que les permutations sur les colonnes de G laissent inchangées la probabilité d'erreur par mot code, la distance (ou le poids) du code est inchangé.

D'après ces propriétés, la matrice génératrice G peut s'écrire sous la forme (Pivot de Gauss) :

$$\check{G} = [I_k \quad \vdots \quad P_{k \times (n-k)}] \quad (1.13)$$

où I_k est la matrice identité de taille k , et la matrice P est appelée la matrice de partie du code.

Lorsque G est sous sa forme systématique, les mots code commencent par k bits d'information et sont complétés par $(n - k)$ bits de redondance :

Si $d = [d_1 d_2 \dots \dots d_k]$ alors

$$c = d * G = [d_1 d_2 \dots \dots d_k \ p_1 p_2 \dots \dots p_{n-k}] \quad (1.14)$$

c. Caractérisation d'un mot code

Matrice de contrôle de parité : prenons en considération un code linéaire C de matrice génératrice G . La matrice de contrôle de parité peut être désignée par une matrice H de dimension $(n - k) \times n$ et de rang $(n - k)$ qui vérifie la relation suivante

$$H * G^T = 0 \quad (1.15)$$

Si c'est un bloc, de mot code $d = c * G$, alors, on doit avoir :

$$H * d^T = H(c * G)^T = H * G^T * c^T = (H * G^T) * c^T = 0 \quad (1.16)$$

Cela vérifie la propriété : le bloc d reçu est un mot code (donc jugé non erroné) si et seulement si $H * d^T = 0$. La construction d'une matrice de contrôle de parité se fait comme suit : soit un code linéaire dont la matrice G est mis sous forme systématique [4]

$$G = [I_k \quad ; P_{k \times (n-k)}] \quad (1.17)$$

Alors, la matrice de contrôle de parité a pour expression :

$$H = [(P^T)_{(n-k) \times k} \quad ; I_{n-k}] \quad (1.18)$$

Exemple :

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdot & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & \cdot & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & \cdot & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & \cdot & 1 & 0 & 1 \end{bmatrix} \text{ et } H = \begin{bmatrix} 1 & 0 & 1 & 1 & \cdot & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & \cdot & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & \cdot & 0 & 0 & 1 \end{bmatrix} \quad (1.19)$$

d. Décodage par le syndrome

Soit un code linéaire de matrice de contrôle de parité H , c un mot envoyé, et y le mot reçu tel que :

$$y = c + b \quad (1.20)$$

Alors, on a :

$$Hy^T = H(c + b)^T = Hc^T + Hb^T = Hb^T \quad (1.21)$$

Avec :

$$s = Hy^T \quad (1.22)$$

Ainsi, le vecteur s , ne dépend que du bruit et non pas du mot envoyé. On l'appelle le syndrome associé à y . Le syndrome se calcule à partir du mot reçu et de la matrice de contrôle de parité, et ne dépend que du bruit et non pas du mot de code émis. A partir de la connaissance de H et de y , on peut donc espérer déterminer le bruit b , car

$$s = Hy^T = Hb^T \quad (1.23)$$

Une fois le bruit connu, c se déduit simplement par :

$$c = y \oplus b \quad (1.24)$$

Supposons qu'une erreur e se soit effectivement glissée dans la transmission d'un mot c , codé par le mot :

$$d = c \cdot G \quad (1.25)$$

Et réceptionné selon :

$$y = d + e \quad (1.26)$$

L'erreur s'écrit alors $e = [0 \dots 010 \dots 0]$ le 1 étant à la $j^{\text{ème}}$ position :

$$s = Hy^T = H(d + e)^T = Hd^T + He^T = H_j \quad (1.27)$$

ou H_j est la $j^{\text{ème}}$ colonne de la matrice de contrôle H .

On connaît ainsi la position de l'erreur, et il n'y a plus qu'à corriger le mot reçu [4].

e. Code de Hamming

Le code de Hamming est un code binaire défini par sa matrice de parité plutôt que par sa matrice génératrice. C'est la matrice de dimension $r \times (2r - 1)$ qui contient toutes les colonnes non nulles distinctes que l'on peut écrire sur r bits.

Le code de Hamming est donc l'ensemble des mots de longueur $2^r - 1$ dont le noyau est H . C'est donc un espace de dimension $2^r - 1 - r$. De plus, la distance minimale de ce code est 3 car il n'existe aucun mot de code de poids 1 ou 2 puisque cela signifierait qu'il y a une

colonne nulle ou deux colonnes égales dans H . On a donc un code $[2^r - 1, 2^r - 1 - r, 3]$ dans lequel on doit donc pouvoir décoder une erreur [31].

1.2.4 Codes polynomiaux [5]

a. Définition

C'est une sous-classe des codes linéaires systématiques qui est une représentation polynomiale d'un nombre binaire. Soit un mot binaire $(b_{n-1}b_{n-2} \dots b_1b_0)$ de longueur n . On peut représenter ce nombre par un polynôme $P(X)$ de variable X et de degré $(n - 1)$ donc les coefficients binaires sont tel que :

$$P(x) = b_0 + b_1 X + b_2 X^2 \dots b_{n-1} X^{n-1} \quad (1.28)$$

On note P_B l'ensemble des polynômes à coefficients binaires.

b. Principe du codage

Un code polynomial de $B^m \rightarrow B^n$ est défini par un polynôme générateur $G(x)$ de degré :

$$r = n - m.$$

Le codage s'effectue comme suit : on représente le mot M à coder sous la forme d'un polynôme $M(X)$ de degré $m - 1$ on multiplie ensuite $M(X)$ par le polynôme X^r (décalage de r bits du mot M), puis on effectue la division euclidienne de $X^r \cdot M(X)$ par $G(X)$ on obtient :

$$X^r \cdot M(X) = G(X) \cdot Q(X) + R(X) \quad (1.29)$$

Avec comme degré

$$R(X) \leq r - 1 \quad (1.30)$$

Le mot transmis T a pour représentation polynomiale :

$$T(X) = G(X) \cdot Q(X) \quad (1.31)$$

D'une manière pratique, T est obtenu directement à partir de

$$T(X) = G(X) \cdot Q(X) = X^r \cdot M(X) + R(X) \quad (1.32)$$

$T(X)$ est de degré $n - 1$ et le mot transmis T est de longueur n

Remarque : tout code polynomial est un code linéaire et systématique. Soit un code polynomial de $B^m \rightarrow B^n$ de polynôme générateur $G(X)$, alors les mots de codes sont les polynômes de degré inférieur ou égal à n qui sont multiples du polynôme générateur.

c. Principe du décodage

Le décodage s'effectue par la division euclidienne de la représentation polynomiale $M(X)$ du mot reçu par le polynôme générateur $G(X)$, deux cas peuvent se présenter : si le reste est nul alors $M(X)$ est un mot de code. Mais si le reste est non nul le décodage se fait par calcul direct du syndrome qui est défini comme suit :

Le syndrome du mot $M(X)$ est égal au reste de la division euclidienne de $M(X)$ par le polynôme générateur.

Théorème : soit un code polynomial de $B^m \rightarrow B^n$ de polynôme générateur $G(X)$. Le reste de la division de tout mot reçu par $G(X)$ est égal au reste de la division du polynôme représentant le vecteur d'erreur de transmission.

Capacités de détection des codes polynomiaux : soit un code de polynôme générateur $G(X)$ de degré r qui est de la forme $1 + \dots + X^r$ (i.e. coefficient g_0 non nul) alors ce code détecte toute erreur simple. Les erreurs doubles sont toutes détectées si le polynôme générateur $G(X)$ ne divise pas $X^u + 1$ pour tout $u \in [1, n - 1]$.

Soit un code de polynôme générateur $G(X)$. Tout message comportant un nombre impair d'erreur est détecté si $G(X)$ comporte $X + 1$ en facteur.

1.2.5 Codes cycliques [5]

Les codes cycliques sont une sous-classe des codes polynomiaux aux capacités de correction intéressante.

a. Définition

Soit un code polynomial de $B^m \rightarrow B^n$ de polynôme générateur $G(X)$. Le code est dit cyclique si $G(X)$ divise $X^n + 1$ (ou $X^n - 1$). Toute permutation circulaire d'un mot d'un code cyclique (polynomial ou non) est encore un mot de code.

b. Codes cycliques à polynômes irréductibles primitifs

On appelle période ou ordre d'un polynôme $G(X)$ le plus petit entier u tel que $G(X)$ divise $X^u + 1$.

Exemple : un code polynomial cyclique a une période au moins égale à la longueur du code. Un polynôme $G(X)$ est dit irréductible s'il ne possède aucun diviseur (autre que lui-même) de degré supérieur à zéro. Si un polynôme de degré r est irréductible, sa période divise $2^r - 1$. Et lorsque sa période est égale à $2^r - 1$, on parle de polynôme primitif.

c. Capacités de correction

Un code cyclique dont le polynôme générateur est primitif est capable de corriger toutes les erreurs simples.

1.2.6 Codes correcteurs performants [6]

a. Les codes BCH (Bose-Chaudhuri-Hocquenghem)

Sont ceux qui ont la plus grande capacité de correction d'erreurs indépendantes pour une redondance et une longueur données. Leur rendement n'est pas très élevé. Ce sont une extension des codes cycliques, ils sont non pas construits sur un alphabet binaire mais un alphabet composé d'un grand ensemble de symboles.

b. Les codes RS (Reed-Salomon)

Sont des codes correcteurs très puissants. Ils peuvent être présentés comme des codes BCH dans lequel chaque bit des mots du code est remplacé par un entier défini modulo 2^v (avec $n = 2^v - 1$). La distance d'un code RS (m, n) est égale à $n - m + 1$.

1.3 Codes convolutifs (ou convolutionnels)

1.3.1 Généralités

Le principe des codes convolutifs, inventés par Peter Elias en 1954, est non plus de découper le message en blocs finis, mais de le considérer comme une séquence semi-infinie $a_0 a_1 a_2 \dots$ de symboles qui passe à travers une succession de registres à décalage, dont le nombre est appelé mémoire du code. Pour simplifier, et également parce que c'est ainsi pour la quasi-totalité des codes convolutifs utilisés, nous considérerons le cas où le message est constitué de bits. Dans l'exemple représenté en Figure 1.3, a_t parvient au codeur à l'instant t . Les bits de sortie seront :

$$X = a_t + a_{t-1} + a_{t-2} \quad (1.33)$$

$$Y = a_t + a_{t-2} \text{ (Addition modulo 2)} \quad (1.34)$$

Supposons que le codeur reçoive le message 1011, les deux registres étant initialement tous deux à 0. On constate (figure 1.4) que la séquence codée devient 11 10 00 01, et que les registres seront finalement à l'état 11. On représente le code par un diagramme, appelé diagramme des états de transition, donné en figure 1.4. Les cases représentent les différents états des registres, et les flèches les transitions possibles (le premier chiffre correspond au bit d'entrée, les deux autres aux bits de sortie X et Y) [7].

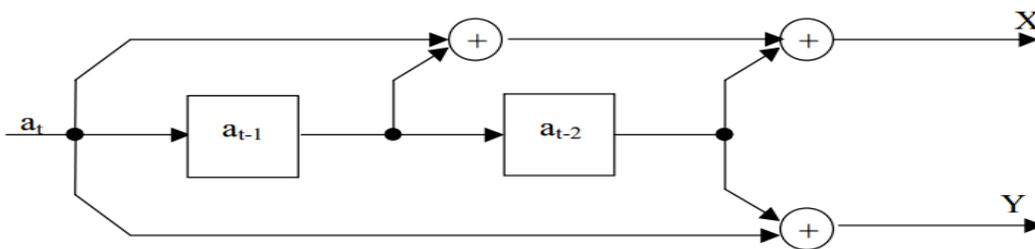


Figure 1.3. Exemple de codeur convolutif [7].

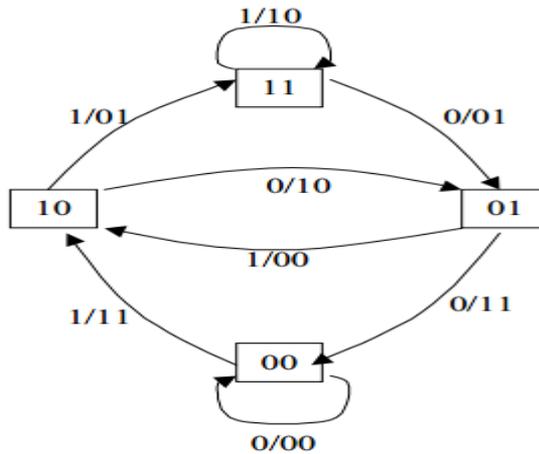


Figure 1.4. Diagramme des états de transition du codeur de la Figure 1.3 [7].

1.3.2 Présentation du codeur

Le codeur des codes convolutionnels est un codeur (n, k, m) qui a des blocs de k bits pour la séquence d'information U et produit une séquence codée V de n symboles (k entrée et n sorties). Dans ce type de codes chaque bloc codé ne dépend pas seulement des blocs de message de k bits correspondant dans la même unité de temps mais aussi des m précédents blocs de messages d'où l'existence de codeur à mémoire d'ordre m qui doit être implémenté avec un circuit logique séquentiel c'est à dire à l'aide des multiplexeurs pour rendre en série l'information en entrée et en sortie un modulo additionneurs (ou exclusif) et des registres à décalage [8] :

- n : le nombre de sorties,
- k : le nombre d'entrées,
- m : le nombre de cellules de tous les étages.

Il nécessite par suite un décodeur à mémoire d'ordre m . Les r $(n-k)$ bits redondants qui sont ajoutés à la séquence d'information croît avec m d'où k est strictement inférieur à n ou bien le rendement $R < 1$.

Pour illustrer ce dernier on prend comme exemple un codeur $(2, 1, 3)$ figure 1.5 :

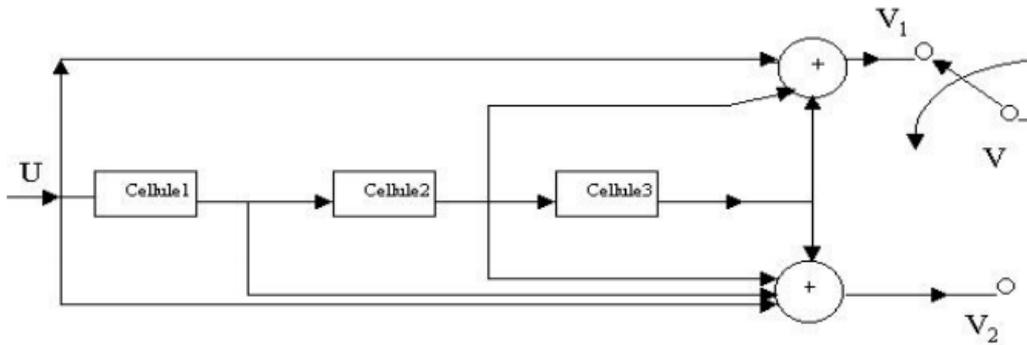


Figure 1.5. Codeur (2, 1, 3) [8].

Considérons la séquence d'information $U = (10110\dots\dots)$ avec le bit le plus à gauche est supposé entré le premier. En utilisant les règles de l'addition avec le ou exclusif ($1+1=0$) ainsi que le multiplexeur qui sert à rendre la séquence codée en série. On obtient en sortie donc la séquence en sortie $V = (11,01,00,01, \dots\dots\dots)$.

1.3.3 Analyse des Codes Convolutionnels [8]

Pour analyser les codes convolutionnels. On préfère donc représenter graphiquement le fonctionnement du codeur, l'étude du codage convolutionnel par un diagramme arborescent, devient vite impraticable dès que la séquence à coder dépasse quelques bits, puisque la dimension de l'arbre double à chaque étage. On préfère donc représenter le fonctionnement du codeur à l'aide d'un diagramme d'état ou le diagramme en treillis. On suppose que les registres du codeur sont initialisés à zéro

a. Diagramme d'état

Le diagramme d'état dans lequel chaque état représente un état particulier des registres du codeur et chaque branche représente le changement d'état du codeur en fonction de l'arrivée d'un nouveau bit. Les différentes branches sont repérées par la valeur du bit d'entrée qui provoque le changement d'état et par le mot de code produit à l'arrivée de ce bit. Dans le cas de la figure 1.5 correspondant au codeur (2,1,3), le codeur comprend trois registres et peut prendre huit états auxquels correspond huit nœuds du diagramme d'état figure 1.6. Les états si correspondant aux états des cellules à décalage du codeur (2, 1, 3)

sont $S_0 = 000$, $S_1 = 100$, $S_2 = 010$, $S_3 = 110$, $S_4 = 001$, $S_5 = 101$, $S_6 = 011$ et $S_7 = 111$. On procède en posant S_0 comme étant un état initial et final à la fois, on étiquette chaque branche avec le bit qui cause la transition et les deux bits sortants.

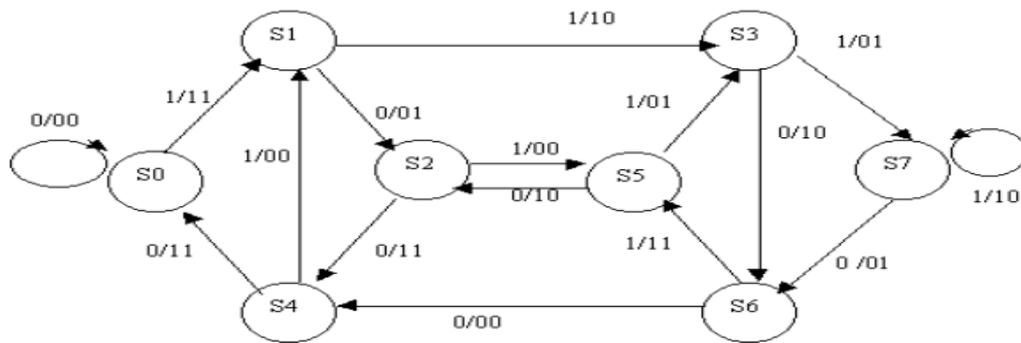


Figure 1.6. Diagramme d'état du codeur (2, 1, 3) [8].

Si l'information à coder est $U = (11101)$ son mot code correspondant est $V = (1110010111101111)$ de longueur $n * (L + m)$ donc 16. Pour connaître le mot code d'une quelconque séquence d'information il suffit de suivre le chemin tout au long du diagramme et donc on constate qu'il revient toujours à S_0 .

b. Diagramme en Treillis

Le fonctionnement d'un codeur convolusionnel peut être également représenté par un diagramme en treillis du type de celui de la figure 1.7 qui correspond au codeur de la figure 1.5, et dans lequel le cheminement du point sur le graphe s'effectue de la gauche vers la droite. Le treillis provient de l'étude des automates d'états finis conçus à l'intention d'expliquer le fonctionnement interne de l'algorithme de Viterbi pour le décodage des codes convolusionnels. A ce jour les treillis pour le décodage de Viterbi restent la motivation fondamentale pour la recherche. Chaque transition d'un état de registre à l'état suivant est repérée par une ligne pleine ou en pointillé selon que le bit qui l'a provoqué est 1 ou 0, et comporte l'indication du mot de code produit.

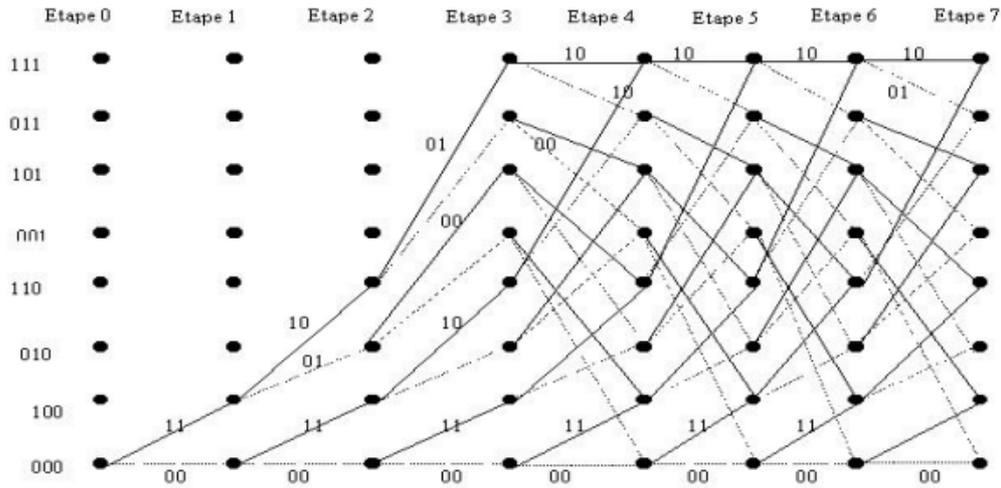


Figure 1.7. Diagramme en treillis du codeur (2, 1, 3) [8].

Cette structure peut être considérée comme un diagramme d'état répété une multitude de fois, le temps que la séquence d'information soit codée. Il permet de connaître un chemin correspondant à un mot code donné. La complexité du treillis croît exponentiellement avec le nombre d'états 2^m , c'est à dire avec la longueur de contrainte Lc .

1.3.4 Décodage des codes convolutionnels [8]

En pratique, il est possible de tirer un meilleur parti des possibilités de détection et de correction des codes convolutionnels en faisant appel à des méthodes probabilistes dont le plus connue est l'algorithme de Viterbi.

- Principe de l'algorithme de Viterbi:

L'algorithme de Viterbi applique au décodage le principe du maximum de vraisemblance et il est basé sur la programmation dynamique. L'algorithme de Viterbi peut être considéré comme une estimation de l'état du codeur. Si le codeur comporte $(K-1)$ étages de m bits, l'état Q_i du codeur juste avant l'arrivée du mot d'information x_i^T peut être considéré comme défini par la valeur des $(K-1)$ mots qui précèdent x_i^T avec :

$$Q_i = (x_{i-1}^T, x_{i-2}^T, \dots, x_{i-k+1}^T) \quad (1.35)$$

Puisque chaque mot d'information à une longueur de m bits Q_i peut prendre $2^{(k-1)m}$ valeurs différentes possibles. L'arrivée du bloc d'information x_i^T fait passer le codeur à l'état Q_{i+1} en provoquant l'émission d'un mot de code y_i^T de n bits. Inversement, il est possible de déterminer le mot d'information x_i^T et le mot de code y_i^T sachant que le codeur est passé de l'état Q_i à l'état Q_{i+1} . Le décodage de la séquence reçue y^T peut être effectué en estimant la séquence des états successifs \hat{Q} qui a donné naissance à y^T . Les séquences émission estimées \hat{x}^T et \hat{y}^T sont ensuite déterminées à partir de la séquence \hat{Q} .

L'algorithme peut être mis en œuvre de façon commode à partir d'un diagramme en treillis en appliquant les règles suivantes :

- Partir du premier étage du diagramme où deux branches arrivent à chaque état, calculer pour chacun des états la distance $d(y^T, \hat{y}^T)$ de chacun des deux chemins qui arrivent à cet état, pour chaque état, conserver le chemin dont la distance est la plus faible (le survivant), et éliminer l'autre.
- Répéter l'opération pour chaque étage du treillis.
- Lorsque l'origine de tous les survivants passe par un chemin unique, ce dernier représente une partie du mot décodé.

1.4 Entrelacement

L'entrelacement consiste à permuter une séquence de bits de manière à ce que deux symboles proches à l'origine soient le plus éloignés possibles l'un de l'autre. Cela permet en particulier de transformer une erreur portant sur des bits regroupés en une erreur répartie sur l'ensemble de la séquence. Un exemple classique de l'utilité de l'entrelacement est celui des codes correcteurs d'erreurs protégeant les disques compacts : il s'agit d'un code en bloc de Reed-Salomon entrelacé qui permet de corriger plus de 4 000 erreurs consécutives dues à une poussière, une éraflure... On cherche également, en particulier pour les turbo-codes, à réaliser une permutation aussi « chaotique » que possible [7].

1.5 Les turbo-codes

Les turbo-codes ont été inventés en 1991, et présentés à la communauté scientifique en 1993, par une équipe de l'Ecole Nationale Supérieure des Télécommunications de Brest dirigée par Claude Berrou et Alain Glavieux. Les spécialistes des codes correcteurs d'erreurs ont tout d'abord accueilli cette invention avec beaucoup de scepticisme, du fait des extraordinaires performances annoncées. Cependant, d'autres équipes, dans le monde entier, sont parvenues à peu après aux mêmes résultats, ce qui a contribué au développement des turbo-codes. Ils ont été adoptés par toutes les agences spatiales mondiales, et seront utilisés dans la transmission des données des nouveaux standards de téléphonie mobile (UMTS, LTE...etc.) [7].

Les turbos codes sont les premiers codes correcteurs d'erreur à approcher d'aussi près la limite d'efficacité prévue par la théorie fondatrice des sciences de l'information et des télécommunications, c'est la théorie de Shannon.

Rappel sur le théorème de Shannon [6] : on définit un canal de transmission comme un système physique permettant la transmission d'une information entre deux points distants. Le taux d'erreurs binaire (TEB) d'un message est le rapport du nombre de bits erronés par le nombre de bits du message. En 1948, Shannon énonce dans « A Mathematical Theory of Information » le théorème fondamental de la théorie de l'information : tout canal de transmission admet un paramètre C , appelé capacité du canal, tel que pour tout $\varepsilon > 0$ et pour tout $R < C$, il existe un code de taux R permettant la transmission du message avec un taux d'erreurs binaire de ε . En d'autres termes, on peut obtenir des transmissions aussi fiables que l'on veut, en utilisant des codes de taux plus petits que la capacité du canal. Cependant, ce théorème n'indique pas le moyen de construire de tels codes. On cherche donc à construire des codes ayant un taux le plus élevé possible (pour des raisons de temps et de coût) et permettant une fiabilité arbitrairement grande. Les codes classiques présentés plus haut ne permettent pas d'atteindre cette limite. D'où le développement d'autres systèmes de codages tel que les turbo codes et les codes LDPC.

1.5.1 Codeur

Le principe des turbo-codes est l'utilisation conjointe de deux codeurs convolutifs récurrents, non pas en série, comme cela était déjà fait depuis de nombreuses années, mais en parallèle (Figure 1.8). L'entrelaceur permet ainsi de coder avec le même codeur deux séquences d'autant plus différentes que l'entrelacement sera chaotique. On constate sur le schéma représenté Figure 1.8 que le taux de codage R des turbo-codes est de $1/3$: trois bits de sortie pour un bit d'entrée. On peut le ramener à $1/2$ par un poinçonnage qui consiste à ne garder à tout instant que l'un des bits Y_1 ou Y_2 . Les turbo-codes peuvent ainsi être comparés aux codes convolutifs classiques, mais cette opération complexifie encore le décodage ; nous n'en tiendrons donc pas compte dans le paragraphe suivant [7].

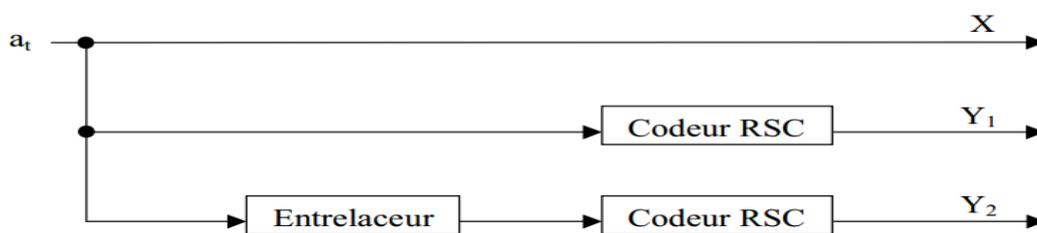


Figure 1.8. Schéma de principe d'un turbo-codeur [7].

1.5.2 Décodeur

Si le codage est relativement simple, le décodage est beaucoup plus complexe. Nous n'en présentons ici que le principe. La Figure 1.9 donne un schéma simplifié du turbo-décodeur, pour information. L'entrelaceur et le désentrelaceur utilisent la même permutation que l'entrelaceur du codeur. Ils permettent par exemple la comparaison de la séquence X avec la séquence Y_2 , entrelacée au moment du codage. Cette procédure de décodage est itérée un certain nombre de fois fixé à l'avance. La décision n'intervient qu'après. A chaque étape, le décodeur dispose des probabilités calculées à l'étape précédente, et se base sur ces probabilités, et non sur une « pré-décision ». C'est ce qui a valu aux turbo-codes leur nom : comme dans un moteur turbo, les « gaz d'échappement » sont réutilisés pour augmenter les performances [7].

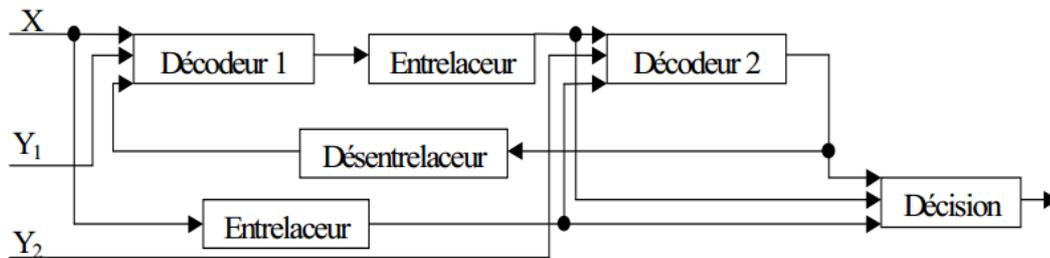


Figure 1.9. Schéma de principe d'un turbo-décodeur [7].

1.6 Les codes LDPC

1.6.1 Historique

Les codes LDPC ont été découverts par Gallager, dans les années 60, mais il a proposé seulement une méthode générale pour construire des codes LDPC pseudo aléatoires ; les bons codes LDPC sont générés par ordinateur (en particulier les codes longs) et leur décodage est très complexe dû au manque de structure. Ces codes ont été ignorés jusqu'à 1981 quand Tanner leur a donné une nouvelle interprétation d'un point de vue graphique, Sa théorie a été aussi ignorée pour les prochaines 14 années jusqu'au jour où quelques chercheurs en codage ont commencé à étudier les codes en graphes et le décodage itératif. La première construction algébrique et systématique de codes LDPC basée sur les géométries finies a été proposée par Kou, Lin et Fosshorier dans les années 2000 [9].

1.6.2 Principe

Les codes LDPC font partie de la classe des codes blocs linéaires et s'approchent davantage de la limite de Shannon (capacité d'un canal) [9].

Un code LDPC est un code dont la matrice de contrôle de parité H est de faible densité. La faible densité signifie qu'il y a plus de 0 que de 1 dans la matrice H . Un code LDPC peut être représenté sous forme matricielle ou bien sous la forme d'un graphe bipartite (représentation de Tanner). Par exemple, la matrice suivante :

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} \quad (1.37)$$

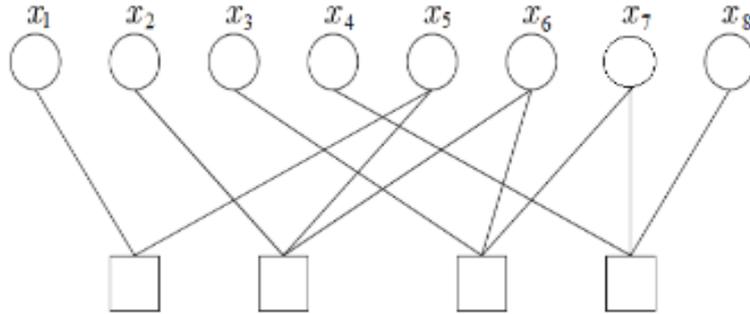


Figure 1.10. Graphe bipartite d'un code LDPC [10].

La matrice peut être représentée par le graphe de la figure 1.10. Les lignes de la matrice sont représentées par des carrés et sont appelées nœuds de contrôle, les colonnes de la matrice sont représentées par des cercles et sont appelées nœuds de données et les «1» représentent les arrêtes du graphe. Il y a deux familles de codes LDPC: les codes réguliers et les codes irréguliers [10].

Les codes LDPC réguliers sont les codes dont le nombre de «1» par ligne et le nombre de «1» par colonne sont constants. Par extension, les codes LDPC irréguliers sont les codes définis par des matrices de contrôle de parité où le nombre de «1» par ligne ou par colonne n'est pas constant. L'irrégularité de ces codes se spécifie à travers deux polynômes et $\rho(x)$.

$$\lambda(x) = \sum_{i \geq 1} \lambda_i X^{i-1} \quad (1.38)$$

$$\rho(x) = \sum_{i \geq 1} \rho_i X^{i-1} \quad (1.39)$$

Où λ_i (resp. ρ_i) caractérise la proportion du nombre de branches connectées aux nœuds de données (aux nœuds de contrôle) de degré i par rapport au nombre total de branche. Le degré est défini comme le nombre de branches connectées à un nœud [10].

1.7. Conclusion

Dans ce chapitre, nous avons fait une étude générale des techniques de détection et de correction d'erreur ; en détaillant dans un premier temps les codes en blocs à savoir les codes en blocs linéaires, polynomiaux, cycliques et un bref détail sur les codes en blocs performant tel que les codes BCH et Reed Salomon , puis les codes convolutifs et turbo codage et enfin le chapitre se termine par un bref aperçu sur les codes LDPC.

Ce chapitre nous a permis d'avoir une idée générale sur les techniques les plus utilisés de détections et de corrections directe d'erreurs de transmission (c'est-à-dire sans intervention d'un processus de retransmission des données erronées).

Des techniques plus performantes en raison de détection et correction d'erreur feront l'objet du deuxième chapitre de ce mémoire à savoir la technique ARQ et HARQ.

Chapitre 2 Techniques de détection et correction d'erreurs ARQ et HARQ

2.1 Introduction

Le système initial ARQ a été inventé par Van Duïren en 1943 afin d'assurer une transmission fiable des caractères. Le premier modèle commercial de la technique ARQ, construit au Pays-Bas, a été mis en service en 1947 [11]. Le principe consiste à ajouter un champ CRC à la trame envoyée, ce champ permet de détecter l'intégrité de la trame. A la réception, le champ CRC permet de vérifier la trame, si elle est fautive un acquittement négatif est envoyé (NACK) et la trame est ensuite retransmise, si elle est correcte un acquittement positif (ACK) est envoyé. Cette technique permet donc de détecter les erreurs de transmission sans les corriger; quand une erreur est détectée, la trame est retransmise [12]. Et pour mieux rendre fiable la transmission on fait appel à la technique HARQ dont le principe est de combiner ces protocoles ARQ avec les FEC présentés dans le premier chapitre, et cela afin de bénéficier des atouts apportés par ces deux techniques. Ces techniques (ARQ et HARQ) font l'objet de notre deuxième chapitre.

2.2 Rappel sur la couche liaison de données

La couche de liaison de données réalise l'empaquetage des données, c'est à dire leur mise en trames de façon à ce que chaque trame soit distinguable de l'autre. Les trames utilisent une chaîne de bits comme leur séparateur pour les distinguer et les séparer. La couche liaison émettrice construit des trames et ajoute un contrôle d'erreur (CRC). La couche liaison réceptrice reconstitue les trames, vérifie la validité avec le CRC, et dans le cas où la trame est erronée ou n'est pas arrivée, elle demande le renvoi [14].

Parmi les responsabilités importantes de la couche liaison de données nous avons : le contrôle du flux et le contrôle d'erreur.

2.2.1 Contrôle d'erreurs

Dans la couche de liaison de données le contrôle d'erreur est réalisé par une répétition de requêtes automatique (ARQ) qui retransmet les données : c'est le protocole ARQ qui est associé à un code de détection des erreurs (voir figure 2.1). L'émetteur joint à k bits d'information, $(n - k)$ bits de redondance. Si le récepteur détecte des erreurs, le paquet est jeté et une retransmission est demandée à l'émetteur. Sinon, le paquet est transmis aux couches supérieures. Le récepteur informe l'émetteur par un mécanisme d'acquittement (ACK ou NACK) [14].

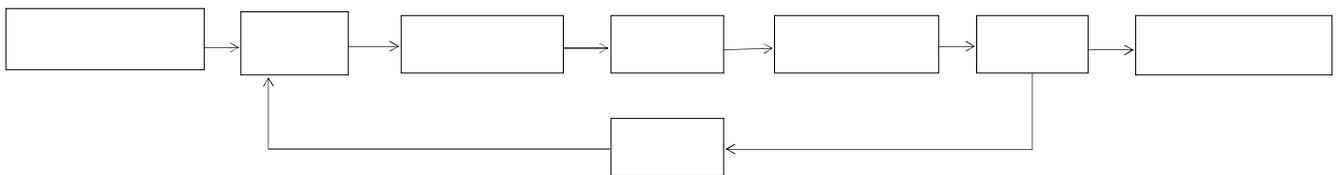


Figure 2.1. Principe de contrôle d'erreur par ARQ [14].

Les techniques de contrôle d'erreurs utilisées par les protocoles de reprise sur erreurs les plus connus dans la littérature sont [15] :

- mécanisme de détection des erreurs
- mécanisme d'acquittement positif ou négatif
- techniques de mémorisation des trames et decorection par retransmission
- utilisation de temporisateurs
- identification des trames (une numérotation permet d'identifier les trames et leur acquittement en détectant : leurs absences, duplications et ordonnancement)

2.2.2 Contrôle du flux [15]

Lorsqu'un émetteur émet de façon systématique plus de trames que le récepteur peut en accepter, il se pose un problème que doit résoudre la couche liaison de données. Même si la transmission s'effectue sans erreur, le récepteur ne peut traiter toutes les trames émises et en perdra certaines. Il faut mettre des mécanismes pour éviter cette situation.

La solution habituelle consiste à instaurer un contrôle de flux pour contraindre l'émetteur à ne pas envoyer plus de trames que le récepteur ne peut en accepter. Son principe de base est le suivant: il est interdit à l'émetteur d'envoyer des trames sans avoir auparavant reçu une permission implicite ou explicite du récepteur. Les mécanismes mis en œuvre :

- utilisation d'acquittements.
- gestion de temporisateurs.
- numérotation des trames.
- limitation du nombre de trames pouvant être envoyées par l'émetteur.

2.3 Calcul du CRC [16]

Le calcul utilise le principe de la division polynomiale. Ainsi, la trame d'entrée notée A est divisée par un polynôme, qu'on appelle polynôme générateur, et donc on peut récupérer le CRC qui est le reste de la division euclidienne. A la réception on ré-divisera la trame $B = A + CRC$ par ce même polynôme. Si le reste est nul, c'est que la trame de départ est la même que la trame d'arrivée, si non la trame a subi des erreurs.

En effet, si le résultat de la division de A par le polynôme générateur donne le résultat res et un reste CRC , donc A s'écrit :

$$A = (pol * res) + CRC(2.1)$$

La trame envoyée B est formée de la trame utile A et du champ $CRC(B = A + CRC)$. Puisqu'on est en logique binaire, donc d'après l'équation précédente on a :

$$A + CRC = (pol * res)(2.2)$$

Si l'ensemble $A + CRC$ ne subit pas d'erreur, à la réception, sa division par pol devra donner res et le reste est donc forcément nul.

Exemple : On veut transférer le mot : 1101011011. On décide d'utiliser le polynôme générateur de degré 4 [10011]. D'abord on décale 1101011011 de 4 rangs vers la gauche, résultat : 1101011011 0000. Ensuite on cherche le reste de la division de 11010110110000 par 10011, résultat : 1110. Enfin on applique un XOR entre le mot d'entrée (décalé de 4 rangs) et le reste de la division,

$$1101011011\ 0000\ \text{XOR}\ 1110 = 1101011011\ 1110$$

Ce qui correspond bien au mot de départ suivi du CRC. On transfère cette trame. Si on la récupère telle quelle, on obtient 1101011011 1110. On divise ceci par 10011, et on trouve un reste nul ; le transfert s'est donc correctement déroulé et aucune perte de données n'a été produite. Par contre si on divise la trame reçue par le polynôme générateur et on trouve un reste non nul, on conclut que la trame a subi une erreur et on demande sa retransmission. Le choix du polynôme générateur est fonction de la qualité du résultat du CRC. Il existe des polynômes générateurs qui sont définis comme étant bons voir (Tableau 2.1).

Nom	Polynôme générateur
CRC-4	$X^4 + X^2 + X^1$
CRC-12	$X^{12} + X^{11} + X^3 + X^2 + X^1 + 1$
CRC-16 SDLC(CCITT)	$X^{16} + X^{12} + X^5 + 1$
CRC-16	$X^{16} + X^{15} + X^2 + 1$
CRC-16 Reverse	$X^{16} + X^{14} + X^1 + 1$
SDLC Reverse	$X^{16} + X^{11} + X^4 + 1$
CRC-32 (ethernet)	$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X^1 + 1$

Tableau 2.1. Liste des polynômes générateurs pour chaque degré de CRC[16].

2.4 La technique ARQ

2.4.1 Principe de l'ARQ

La détection d'erreur suivie d'une retransmission est la solution la plus utilisée dans les réseaux numérique. Des mécanismes d'accusés de réception permettent de confirmer à l'émetteur que les données transmises sont bien arrivées sans erreur. Ces accusées de réceptions sont généralement des blocs de données spéciales [33].

L'émetteur formate les trames en ajoutant aux données des champs de supervision, d'adresse, de données et de détection d'erreur, puis il les transmet en conservant une copie en mémoire (voir figure 2.2). Le récepteur émet des acquittements positifs ou négatifs. A chaque acquittement positif, le bloc de données correspondant est détruit et à chaque acquittement négatif, la trame qui n'a pas été correctement reçue est retransmise [47].

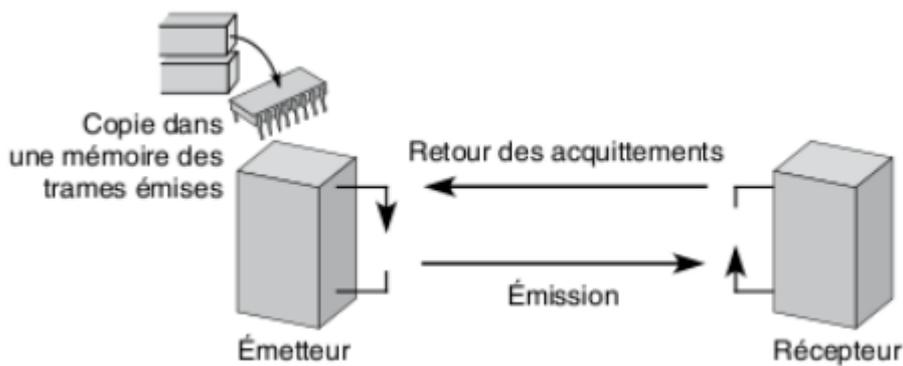


Figure 2.2. fonctionnement d'un protocole de liaison [47]

Divers types de politiques d'acquiescement et de retransmission peuvent être adaptés:

- Stop-and-Wait (ou Send-and-Wait)
- Go-back-N
- Selective Reject (ou Selective Repeat)
- Stop-and-Wait parallèle

a. Protocole Stop-and-Wait [13]

L'émetteur transmet le bloc de données (composé de bits de données + FCS ou CRC) et attend un accusé de réception positif ou négatif [33]. L'émetteur envoie la trame suivante si et seulement si il est assuré que la trame précédente a été reçue correctement par le récepteur, le récepteur acquiesce une trame (ou non) par l'envoi d'une petite trame de contrôle ACK si la trame qu'il vient de recevoir est correcte et NACK sinon (voir figure 2.3 et 2.4).

L'émetteur attend une trame ACK ou NACK après avoir envoyé la trame de données mais seulement pendant un temps déterminé (TIMEOUT) car ACK ou NACK risquent de ne jamais arriver ou arriver de manière erronée. L'émetteur renvoie la trame s'il y a réception d'un NACK ou dépassement du délai (TIMEOUT).

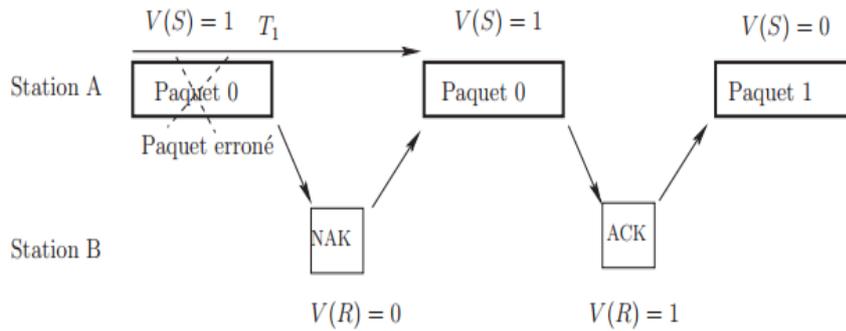


Figure 2.3. Protocole Send and Wait ($V(S)$: numéro de séquence du prochain paquet à transmettre. $V(R)$: numéro de séquence du paquet attendu) [11].

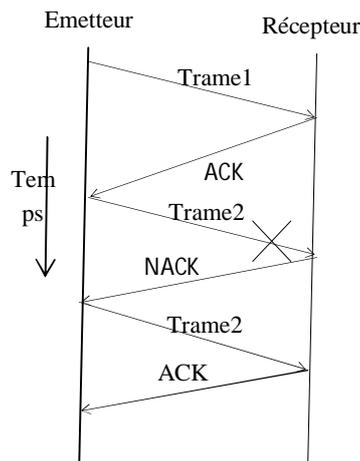


Figure 2.4. Le principe d'acquittement lors du protocole Stop-and-Wait

Remarque [11]: Le principal avantage du protocole Stop-and-Wait est la simplicité des règles d'émission et de réception, ainsi que son très faible coût mémoire en effet, il y a un seul paquet à mémoriser coté émetteur et le récepteur mémorise le paquet attendu lorsqu'il est correctement reçu. Le principal inconvénient du protocole Stop-and-Wait est que l'émetteur ne fait rien tant qu'il n'a pas reçu l'acquittement de la station réceptrice.

b. Protocole Go-back-N [11]

Avec cette méthode, l'émetteur envoie une série de blocs successifs contenant des numéros de séquence sans attendre d'accusé de réception entre deux blocs, il ne s'interrompt que lorsqu'il reçoit un accusé de réception négatif (NACK). Dans ce cas-là l'émetteur retransmet alors le bloc erroné ainsi que les blocs suivants. Par exemple, si l'émetteur envoie 10 blocs numérotés de 1 à 10, et que le bloc 7 est altéré, le récepteur envoie un accusé de réception négatif pour ce bloc à l'émetteur, qui lui envoie les blocs 7,8,9 et 10 [33].

Le protocole Go-Back-N consiste à transmettre successivement des paquets sans attendre la réponse pour chacun. Il vise à remédier au délai d'attente en Stop-and-Wait par une transmission en continue sur une fenêtre appelée fenêtre d'anticipation. Cette stratégie d'envoi de paquets permet d'améliorer l'efficacité des transmissions (meilleur taux d'utilisation du lien). Le nombre maximum de paquets autorisés à être émis sans attendre un acquittement est appelé taille de la fenêtre d'anticipation, et est notée w . Achaque instant, la station A détient la liste des numéros de séquence des paquets qu'elle peut envoyer. De son côté, la station B détient le numéro de séquence du prochain paquet attendu. La station A transmet en continu les w paquets de sa fenêtre d'anticipation. Lorsque la station A reçoit un acquittement positif de la station B, elle déplace sa fenêtre d'une position et transmet les paquets suivants. Lorsque A reçoit un acquittement négatif, elle interrompt la transmission en cours et retransmet tous les paquets de la fenêtre à partir du paquet erroné. la station A doit donc stocker en mémoire les paquets non acquittés à l'intérieur de la fenêtre courante, pour pouvoir les retransmettre en cas de perte ou de mauvaise réception. De son coté, tout comme dans le protocole Stop-and-Wait, la station B rejette tout paquet autre que le prochain paquet attendu. La taille de la fenêtre d'anticipation est calculée en fonction des paramètres du système (délai de propagation, taille de paquets, etc.). Le nombre de bits b réservés à la numérotation de paquets est fonction de w . On montre qu'il est donné par :

$$b = \log_2(w + 1) \tag{2.3}$$

La figure 2.5 et 2.6 illustrent des exemples de transmission s'appuyant sur le protocole Go-Back-N.

Grâce à l'utilisation de la fenêtre d'anticipation, le protocole Go-Back-N améliore le taux d'utilisation du lien comparativement au protocole Stop-and-Wait, en contrepartie, d'une complexité mémoire (on stocke w paquets) et d'une complexité de traitement supérieure coté émetteur. Coté récepteur, le fonctionnement du protocole est similaire au Stop-and-Wait.

Son principal inconvénient est qu'en cas de réception d'un acquittement négatif, on doit retransmettre tous les paquets déjà envoyés à partir du paquet signalé en erreur, ce qui réduit d'autant le débit utile du protocole.

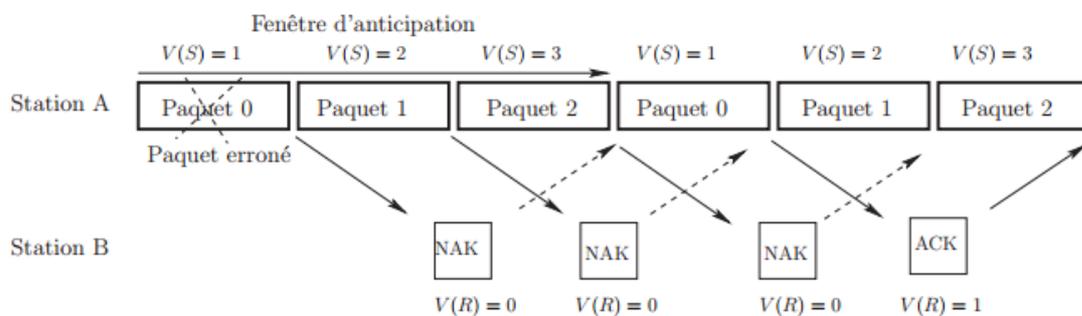


Figure 2.5. Protocole Go-Back-N avec une fenêtre d'anticipation de taille 3 [11].

$V(S)$ et $V(R)$ des compteurs avec $V(S)$ contenant le numéro de la prochaine trame à envoyer et $V(R)$ celui de la trame à recevoir.

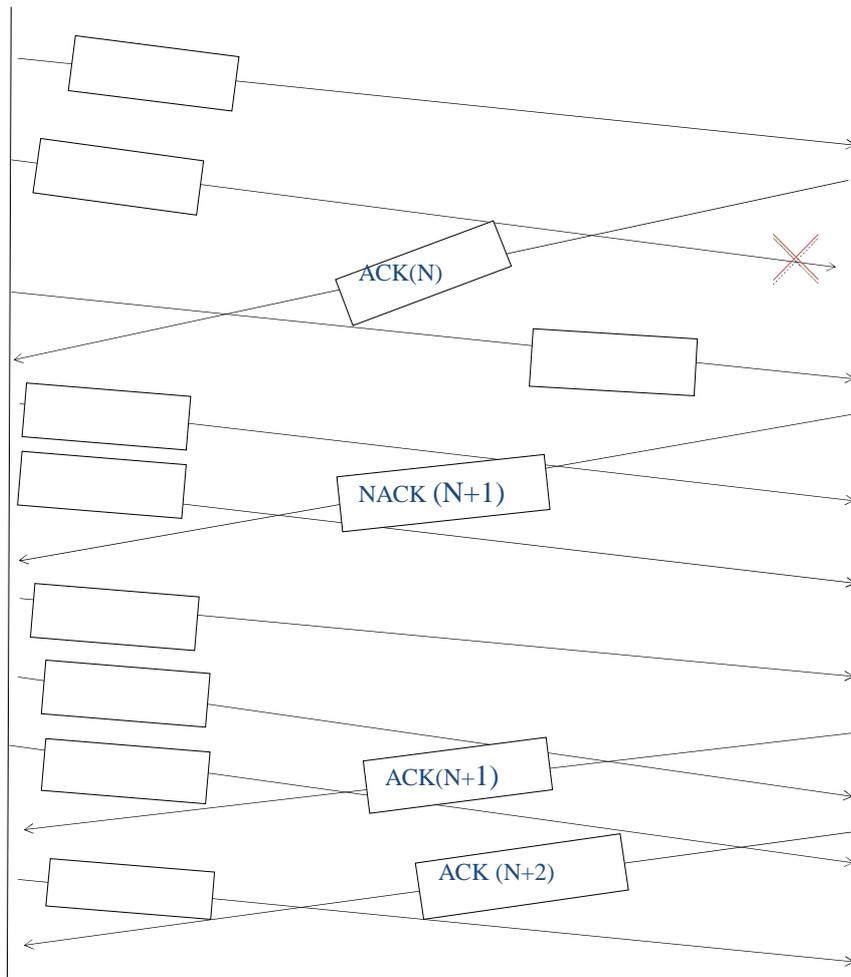


Figure 2.6. Principe des acquittements dans le protocole Go-Back-N [34]

c. Protocole Selective-Repeat

Dans ce protocole, comme le protocole Go-Back-N, les paquets sont envoyés en continu à partir de l'émetteur. L'émetteur continue à envoyer des nouveaux paquets tant que des messages ACK sont reçus. Lorsqu'un message NACK est reçu, l'émetteur envoie seulement le paquet erroné. Puisque les paquets devraient être livrés en bon ordre aux couches supérieures au niveau du récepteur, un tampon doit être prévu au niveau du récepteur pour stocker les paquets qui sont reçus sans erreur qui ont suivi un paquet erroné. Lorsque le premier paquet reconnu négatif est reçu avec succès, le récepteur libère alors les paquets sans erreur dans un ordre consécutif jusqu'à la rencontre du prochain paquet erroné. Selective-

Repeat est le plus efficace des trois protocoles ARQ. Son principal inconvénient est la complexité accrue qu'elle induit à la fois au niveau émetteur et le récepteur qui doivent être équipés de tampons. Le protocole Selective-Repeat est illustré à la figure 2.7 [17].

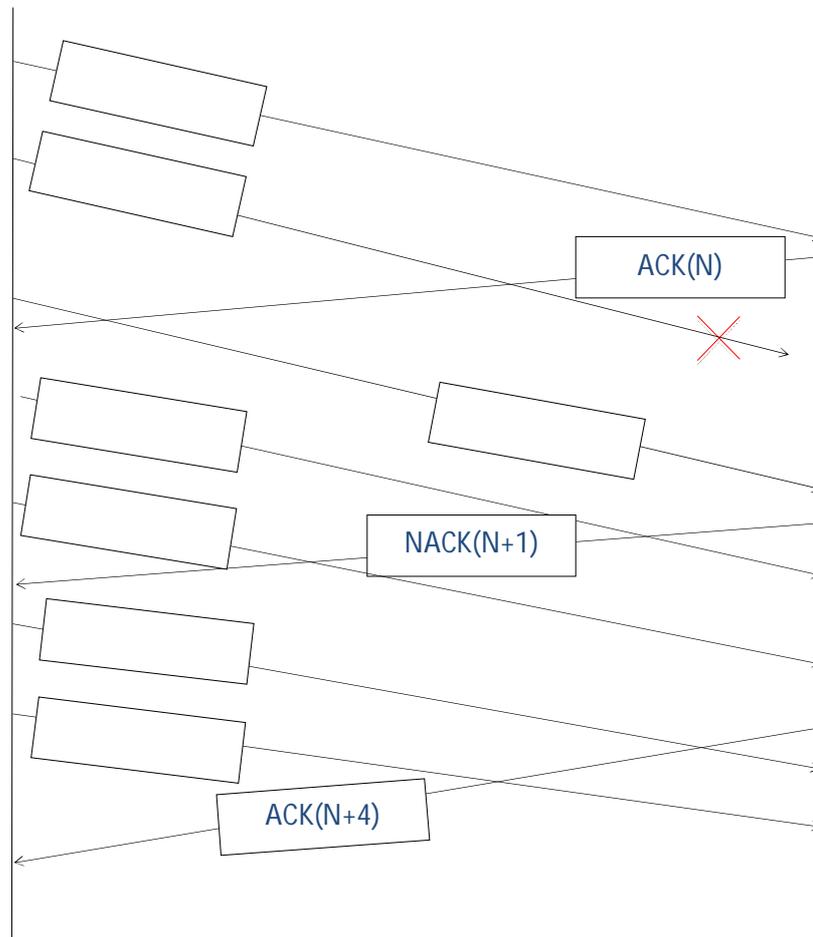


Figure 2.7. Principe d'acquiescement dans le protocole Selective-Repeat[34]

d. Protocole Stop-and-Wait parallele[11]

Le Protocole Stop-and-Wait parallele consiste à faire tourner plusieurs processus Stop-and-Wait en parallele. Chaque processus fonctionne comme un protocole classique Stop-and-Wait. Le nombre maximal de processus w doit être égal au nombre maximal de paquets que l'on peut transmettre pendant la durée d'aller-retour (taille d'une fenetre d'anticipation). L'augmentation du nombre de processus Stop-and-Wait parallele au-delà du nombre maximum de paquets autorisés introduit des retards sur les retransmissions et n'est donc pas souhaitable. La figure 2.8 illustre le principe du protocole Stop-and-Wait parallele sur un

exemple avec trois processus en parallèle (P_1 , P_2 et P_3). L'avantage de ce protocole est qu'il permet d'améliorer le débit du protocole Stop-and-Wait grâce à une transmission en continu, tout en conservant pour l'essentiel, la simplicité des règles du Stop-and-Wait classique.

Par l'utilisation de w processus Stop-and-Wait parallèle, le protocole Stop-and-Wait peut être vu comme un protocole à fenêtre d'anticipation de taille w . Lorsque le nombre de processus est ajusté au délai d'aller-retour, le protocole Stop-and-Wait permet d'occuper au maximum les délais d'attente. Grâce à un taux d'utilisation du lien maximum, il aura un débit comparable au protocole Selective-Repeat, et par conséquent un débit supérieur au protocole Go-Back-N. Comme ce protocole est composé de plusieurs processus (protocoles) Stop-and-Wait, la complexité des règles de dialogue est réduite par rapport au Selective-Repeat. Les demandes en mémoire sont également réduites. Notons toutefois que les entêtes des paquets doivent inclure un champ supplémentaire pour la numérotation des processus Stop-and-Wait parallèle. Par ailleurs, tout comme en Selective-Repeat, le ré-ordonnement des paquets en réception est nécessaire.

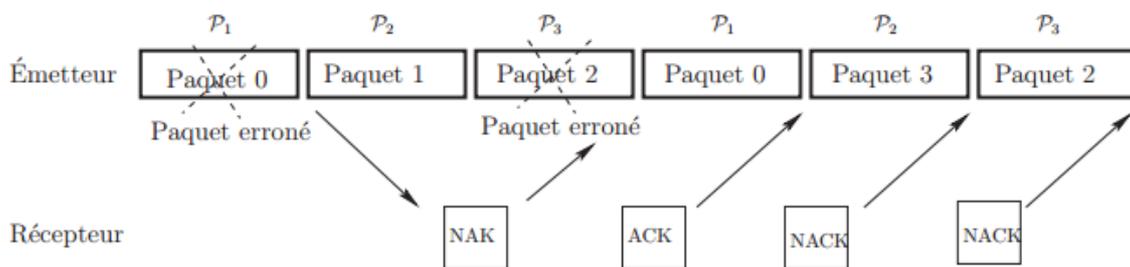


Figure 2.8. Protocole ARQ Stop-and-Wait parallèle avec un nombre de $w = 3$ processus Stop-and-Wait parallèle [11].

2.4.2 Remarques sur la gestion des files d'attente [14]

La conception d'un protocole ARQ dépend de la mémoire disponible chez l'émetteur et chez le récepteur. L'émetteur doit mémoriser des paquets ($B1$) pour une possible retransmission. Le récepteur doit mémoriser les paquets ($B2$) correctement reçus mais qui ne peuvent pas être livrés en séquence.

- Si $B1 = 1$, l'émetteur ne peut mémoriser qu'un seul paquet : c'est le cas du protocole Stop-and-Wait.
- Si $B1 = N$ et $B2 = 0$, l'émetteur peut émettre N paquets sans recevoir d'accusé de réception ; en cas d'erreur il pourra réémettre les N paquets. Dans ce cas le récepteur n'est pas capable de livrer les paquets en séquence, sans demander la retransmission de tous les paquets à partir de l'erreur : c'est le cas du protocole Go-Back-N.
- Si $B2 > 0$, le récepteur est capable en cas d'erreur de conserver des paquets correctement reçus mais non en séquence : c'est le cas du protocole Selective-Repeat.

Remarque : quel que soit le type de retransmission, si l'émetteur ne reçoit pas d'accusé de réception après un délai donné, le bloc non accusé est réémis automatiquement, c'est ce qu'on appelle retransmission après dépassement de délai d'attente.

Rappel sur la Notion de taux d'erreurs : pratiquement on mesure la qualité d'une liaison numérique (qualité de transmission) par le taux d'erreur binaire appelée BER (Bit Error Rate), il est donné par le nombre de bits erronées rapporté au nombre total de bit transmis [33].

$$\text{BER} = \frac{\text{Nombre de bits erronés}}{\text{Nombre de bits transmis}} \quad (2.4)$$

2.5 La technique HARQ

Afin d'assurer la fiabilité des transmissions, le protocole HARQ est utilisé, et qui consiste en une retransmission des données perdues ou mal reçues par l'utilisateur. Il combine les principes de la retransmission ARQ et de la correction des erreurs FEC. Après chaque transmission, le récepteur essaye de décoder le paquet transmis. Si le décodage réussit, un accusé positif (ACK) est envoyé à l'émetteur, sinon un accusé négatif (NACK) lui est envoyé pour demander une retransmission du paquet [18].

La technique HARQ est classé en trois catégories selon la capacité de correction d'erreurs. Ces catégories sont : HARQ type I, type II et type III [19].

a. HARQ type I

Le fonctionnement du schéma HARQ type-I (souvent notée HARQ-I) est similaire au fonctionnement des protocoles ARQ. La station réceptrice (station B) décode le paquet reçu. Si le décodage est correct, elle envoie un acquittement positif pour informer l'émetteur (station A) de la bonne réception du paquet (ACK). Si le d'écodage échoue, elle rejette le paquet erroné et renvoie une demande de retransmission (NACK). Une fois le paquet erroné retransmis, la station B tente à nouveau de décoder l'observation reçue. Cette procédure de retransmission se répète jusqu'à ce que le paquet soit correctement décodé, ou que le nombre maximal de retransmissions autorisées soit atteint.

La technique HARQ type-I décrit un mécanisme ARQ combiné à un codeur FEC. Un code CRC est d'abord généré pour chaque message à transmettre puis ajouté aux données. Le paquet résultant est ensuite codé par un FEC de taux de codage R . L'intérêt principal du HARQ type-I est d'utiliser la capacité de correction du code FEC afin de récupérer les bits d'information qui peuvent être bruités au cours de leurs transmissions et diminuer ainsi la probabilité de retransmission du mécanisme ARQ. Du côté du récepteur, si le paquet reçu contient toujours des erreurs, il est éliminé et un NACK est envoyé à l'émetteur [19].

La technique HARQ type-I reste adaptée aux canaux avec niveau de bruit et interférence à peu près constants [14].

b. HARQ type II

Si le récepteur n'est pas capable de corriger les erreurs d'un paquet, il en conserve tout de même une copie et demande une retransmission. Ces informations contenues dans les paquets erronés mémorisés améliorent la probabilité de décodage [14]. Contrairement au HARQ type I, HARQ type II combine les multiples sous-mots de code reçus à travers les retransmissions. Les deux principales méthodes de combinaisons sont : *Chase Combining* et *Inrcemental Redundancy* [19].

➤ Chase Combining HARQ(CC-HARQ)

Dans cette catégorie du HARQ (CC-HARQ) après la réception d'un NACK, l'émetteur renvoie le même mot de code utilisé dans les tentatives précédentes. A la réception de chaque retransmission, le récepteur combine le signal reçu avec les anciens signaux via la technique de la combinaison de diversité MRC (Maximum Ratio Combining). Ensuite, le signal passe au décodeur qui, à son tour, essaye de décoder à nouveau le paquet. Cette technique apporte un gain en diversité au récepteur. En effet, les signaux émis à chaque tentative subissent des variations indépendantes à leur passage du canal, puisque le canal est supposé indépendant (la variation de la fonction de transfert du canal d'un bloc à l'autre est supposée indépendante)[20].

Avec Chase combinaison (CC-HARQ) comme le montre la figure 2.9, chaque retransmission contient les mêmes bits de parité et de l'information que la transmission initiale par conséquent, il est également appelé codage à répétition. Au récepteur tous les paquets antérieurs sont stockés dans un tampon de telle sorte que les paquets retransmis sont combinés avec des paquets erronés antérieurs avant qu'ils ne soient introduits dans le décodeur. Comme ce système ne transmet pas toute nouvelle redondance, CC-HARQ augmente seulement $\frac{E_b}{N_0}$ reçu accumulé (de bit de l'énergie sur la densité spectrale de bruit) et ne donnent pas de gain de codage supplémentaire pour chaque retransmission [37].

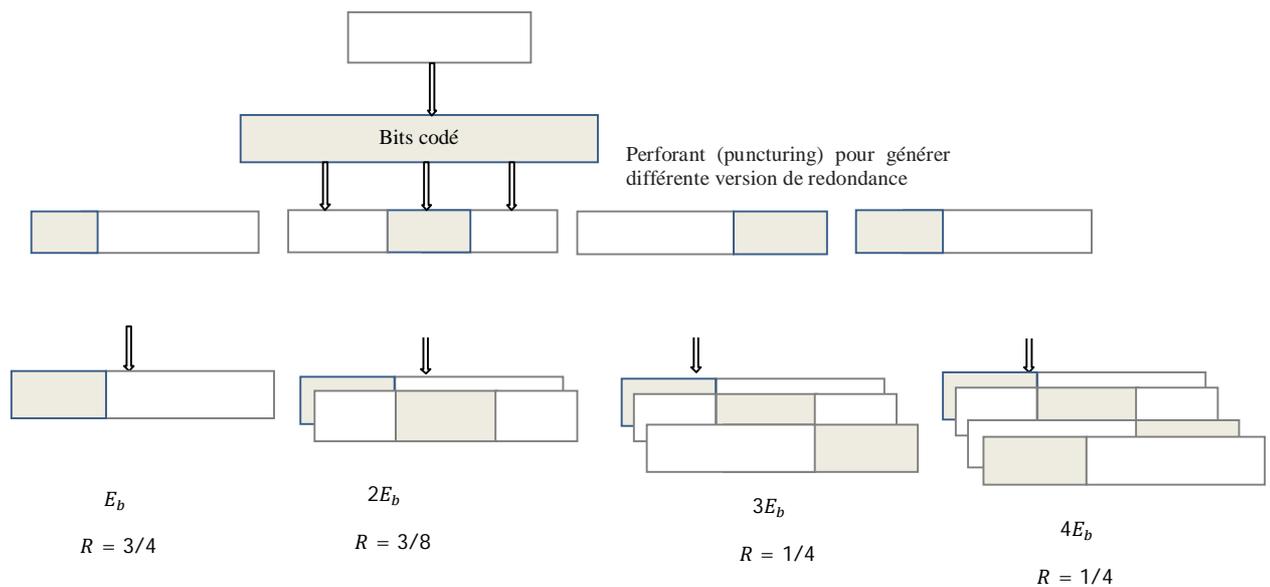


Figure 2.9. Bloc du diagramme du CC- HARQ [37]

➤ **Incremental Redundancy HARQ (IR-HARQ)** [11], [19], [18]

Le protocole IR-HARQ est basé sur le principe suivant, les bits de correction d'erreurs sont envoyés au récepteur uniquement au besoin. En effet, la première transmission est constituée par les bits d'information et les bits de détection d'erreurs. Si le récepteur détecte des erreurs, la transmission est conservée au niveau de sa file d'attente et un NACK est envoyé à l'émetteur. Les retransmissions sont généralement différentes et contiennent les bits de parité permettant la correction des erreurs. Ainsi, après chaque retransmission, la redondance additionnelle (redondance incrémentale) combinée avec les retransmissions précédentes, engendre un mot de code avec une capacité de correction plus grande. Il se construit donc ainsi en réception un code correcteur de plus en plus puissant au fur et à mesure des retransmissions.

En pratique, les paquets de redondance sont habituellement construits par poinçonnage compatible en rendement d'un code ou schéma de modulation et codage (MCS) à faible rendement, appelé code mère C. A chaque round, la station A transmet un fragment différent du mot de code initial produit par C. Comme la redondance est envoyée par morceaux lors de la détection d'erreur, IR-HARQ est un système polyvalent, et donne un bon compromis entre ARQ et FEC. Le principe est illustré dans la figure 2.10.

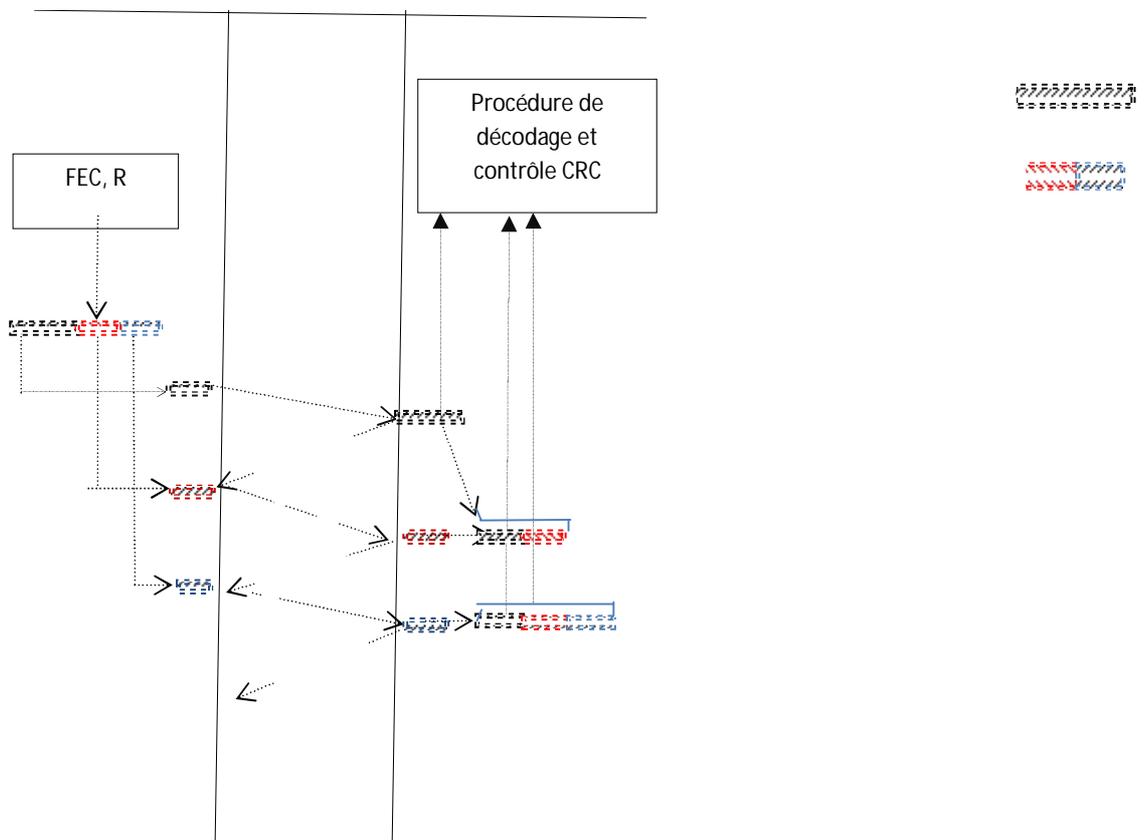


Figure 2.10. HARQ- IR protocole de retransmission pour $N_{max} = 3$. Chaque paquet réémis, la redondance est ajouté aux paquets précédemment erronés [21].

c. HARQ type III

HARQ type III pourrait être considéré comme un cas particulier du HARQ type II où chaque transmission est décodable. En effet, chaque sous mot de code contient suffisamment de bits d'information et de parité pour pouvoir le décoder. Chaque sous mot de code pourrait ainsi être décodé seul ou combiné avec les autres transmissions du même paquet reçues précédemment [19].

2.6 Conclusion

Au cours de ce chapitre nous avons eu à faire une étude en détail de la technique ARQ, en expliquant ces différentes variantes à savoir le Stop-and-Wait, Go-back-N, sélective repeat et le Stop-and-Wait parallèle; une étude descriptive de la technique hybride ARQ (HARQ) qui constitue le noyau de notre étude est aussi abordé dans ce chapitre, cela en détaillant ces trois variantes : le HARQ type I, type II (CC-HARQ et IR-HARQ) et enfin le HARQ type III.

Ce chapitre nous a donc permis non seulement de comprendre le principe de fonctionnement des techniques de reprise sur erreur ARQ, mais aussi de comprendre leur version évoluée l'hybride ARQ combinant celle-ci avec les techniques FEC.

Lors du prochain chapitre une étude des techniques d'application du protocole HARQ sera abordée.

Chapitre 3 Domaines d'applications du protocole

HARQ

3.1. Introduction

Le protocole HARQ est présent dans divers normes des réseaux radio mobile tel que le Wimax et dans certaines mises à jour (*release*) de la 3GPP comme le HSPA et le LTE et ce dans un but d'améliorer la fiabilité du transfert des données en minimisant le nombre de retransmission du protocole ARQ. Parmi les techniques d'accès multiple où l'on retrouve le protocole HARQ on cite la SC-FDMA, cette dernière, a été adoptée en 2008 pour les communications en voix montante (*uplink*) de la norme 3GPP LTE dit de 4G [22] et permet le partage des ressources radio dans un système de communication radio mobile.

Tout au long de ce chapitre nous passerons en revue l'implémentation de la technique HARQ dans les normes LTE, HSPA et Wimax et nous nous intéresserons plus particulièrement à la technique SC FDMA puisqu'elle a été choisie comme chaîne de transmission dans la partie simulation en chapitre IV.

3.2. La technologie HARQ en HSPA

Rapidement, la volonté apparut d'effacer les limites de la Release 99 en matière de débits. Les évolutions HSPA, aujourd'hui connues commercialement sous le nom de 3G+, furent introduites : le HSDPA pour la voie descendante et HSUPA pour la voie montante.

Ces évolutions ont été définies par le 3GPP (voir figure 3.1) respectivement en Release 5 (2002) et Release 6 (2005) afin d'accroître les débits possibles et de réduire la latence du

système. Cette dernière désigne le temps de réponse du système à une requête de l'utilisateur, et est un facteur clé de la perception des services de données par l'utilisateur.

L'innovation principale du HSPA concerne le passage d'une commutation circuit sur l'interface radio, où des ressources radio sont réservées à chaque UE pendant la durée de l'appel, à une commutation par paquets, où la station de base décide dynamiquement du partage des ressources entre les utilisateurs actifs. L'allocation dynamique des ressources est effectuée par la fonction d'ordonnancement (*scheduling*), en fonction notamment de la qualité instantanée du canal radio de chaque utilisateur, de ses contraintes de qualité de service, ainsi que de l'efficacité globale du système. La commutation par paquets optimise ainsi l'usage des ressources radio pour les services de données. La modulation et le codage sont rendus adaptatifs afin de s'adapter aux conditions radio de l'utilisateur au moment où il est servi, les débits instantanés étant accrus via l'utilisation de modulations à plus grand nombre d'états qu'en Release 99. La modulation 16 QAM est introduite pour la voie descendante en complément de la modulation QPSK en vigueur en Release 99. De même, la modulation QPSK est introduite pour la voie montante en complément de la modulation BPSK utilisée en Release 99.

Enfin, un nouveau mécanisme de retransmission rapide des paquets erronés, appelé HARQ, est défini entre l'utilisateur (UE) et la station de base (eNodeB), afin de réduire la latence du système en cas de perte de paquets. Ces évolutions offrent aux utilisateurs des débits maximaux de 14,4 Mbit/s en voie descendante et de 5,8 Mbit/s en voie montante, ainsi qu'une latence réduite [40].

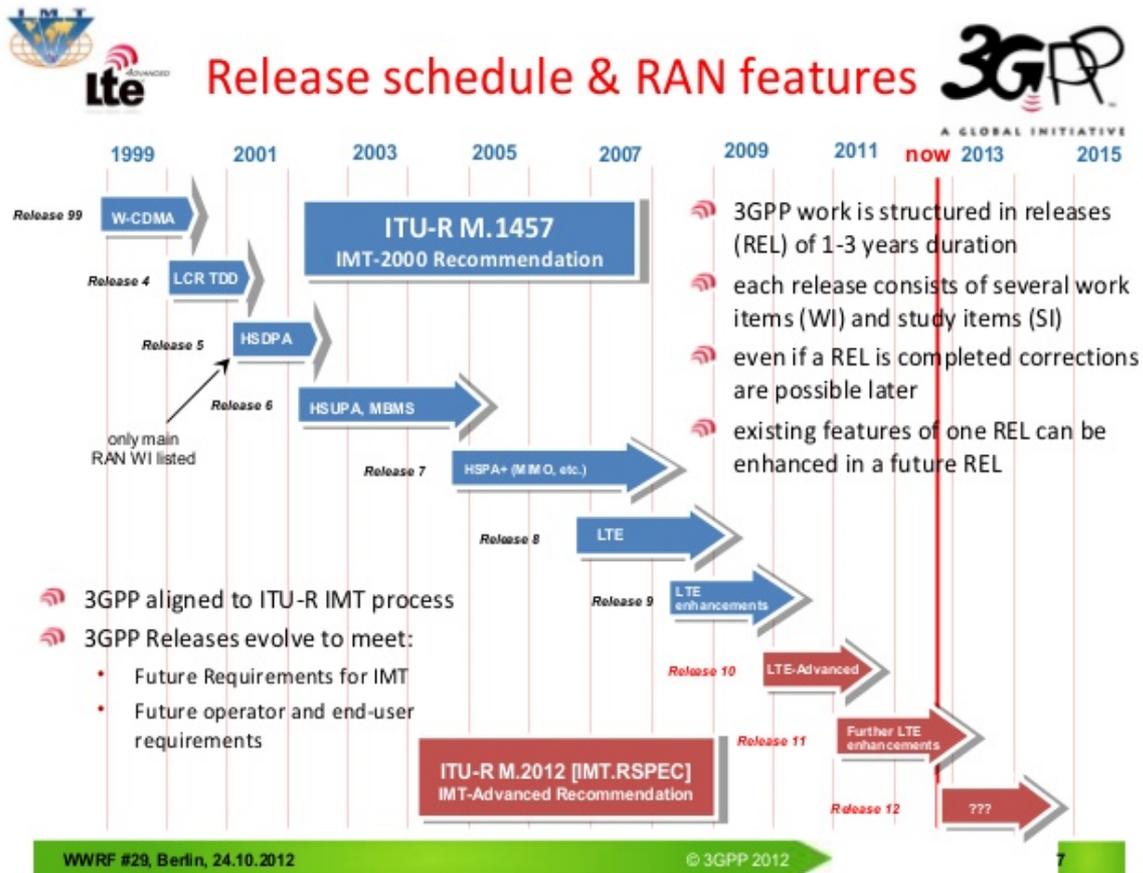


Figure 3.1. Les principales mises à jour (release) de la 3GPP [48]

3.2.1. Mécanisme de retransmission hybride HARQ en HSDPA

Le HARQ peut être caractérisé par certains paramètres tels que la synchronisation, l'adaptabilité ainsi que la manière dont est faite la combinaison [43]. Quand la relation temporelle entre la transmission originale et la (ou les) retransmission(s) est fixe, l'opération HARQ est dite alors synchrone. Si, par contre, les retransmissions sont programmées à n'importe quel moment après avoir reçu un ACK, on parlera alors d'opération HARQ asynchrone.

On dit qu'un système HARQ est adaptatif si on peut réaliser des retransmissions en utilisant un autre type de modulation autre que celui qui a été utilisé pour la transmission originale. Par exemple, si la modulation QPSK est utilisée durant la première transmission d'un paquet et que celle-ci échoue, la 16-QAM peut être utilisée pour la retransmission de ce même paquet si

les conditions du canal ou des ressources radio (puissance du signal et/ou codes de canalisation) changent entre la première transmission et les retransmissions qui s'en suivent.

Le SC-HARQ fait appel à la retransmission par l'émetteur du même paquet de données codées. Le décodeur au niveau du récepteur combine ces copies multiples du paquet envoyé, pondéré par le rapport de signal/bruit SNR reçu ; et qu'en Incremental Redundancy (IR-HARQ), contrairement à la précédente qui envoie des répétitions simples de tout le paquet encodé, envoie une information redondante additionnelle d'une manière incrémentale si le décodage échoue à la première tentative.

La technologie HSDPA s'appuie sur un mécanisme HARQ asynchrone et adaptatif et peut fonctionner avec les deux variantes SC-HARQ ou CC-HARQ et IR-HARQ. Il est à noter qu'avec la deuxième variante, on obtient de meilleures performances, mais elle nécessite plus de mémoire dans le récepteur du terminal mobile pour stocker les informations de redondance. Comme illustré à la figure 3.2, la fonctionnalité HARQ, utilisant la variante IR-HARQ dans cet exemple, est implémentée à travers deux blocs d'adaptation au débit (*rate matching*) dans la chaîne de codage du canal HS-DSCH. Cette dernière est illustrée à la figure 3.3. Le principe réside dans l'existence d'un buffer entre les deux blocs afin de permettre de raffiner les paramètres de redondance pour les différentes retransmissions entre les deux blocs de rate-matching.

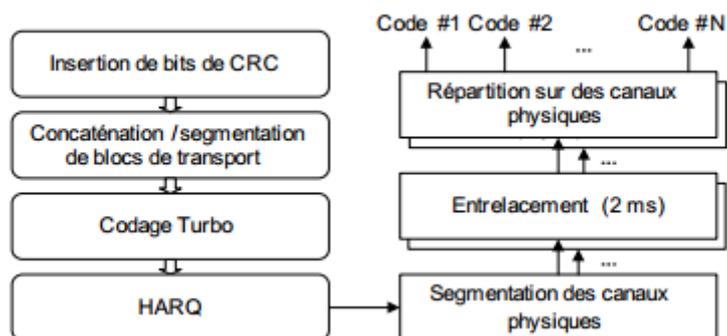


Figure 3.2. Technique HARQ dans la chaîne de codage du canal HS-DSCH [43]

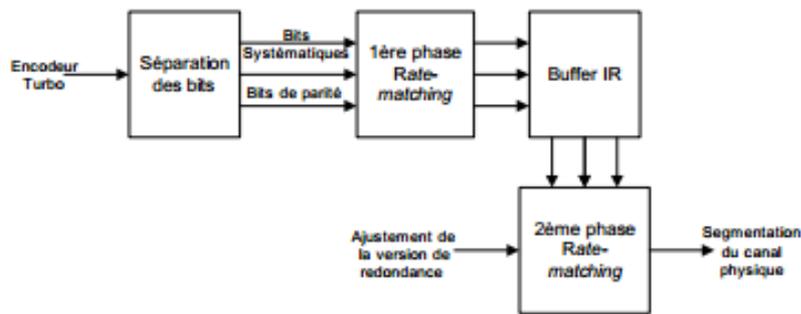


Figure 3.3. Principe de fonctionnement de l’HARQ [43]

3.2.2. La technique HARQ en HSUPA

Le HSUPA est l’extension haut débit de la voie montante qui est introduite par la Release 6 des spécifications 3GPP-UMTS. L’utilisation du HARQ et l’attribution rapide des ressources augmentent l’efficacité spectrale par rapport au procédé WCDMA existant. Le débit peut atteindre des valeurs jusqu’à 5,76 Mbit/s en réduisant la redondance au niveau de la correction d’erreurs.

Le HSUPA doit obtenir toutes les améliorations réalisées par le HSDPA en voix descendante (*downlink*). D’où l’évidence d’utiliser des méthodes similaires. Ainsi, pour une transmission de données efficace, les processus HARQ ont été ici également introduits.

Le processus HARQ déjà connu du HSDPA s’applique aussi en HSUPA. La voie de retour correspondante est l’E-HICH qui transmet à travers ce canal (*E-HICH*) les accusés de réception de l’eNodeB vers l’UE, nécessaires pour les processus HARQ [44].

3.3. Le HARQ dans la LTE

3.3.1. Rappel sur la couche MAC du LTE

La couche MAC est parmi les couches les plus importantes du modèle. La sous-couche MAC (couche 2) permet l’accès et l’adaptation au support de transmission grâce aux fonctions suivantes [40] :

- le mécanisme d'accès aléatoire sur la voie montante ;
- la correction d'erreurs par retransmission HARQ lors de la réception d'un acquittement HARQ négatif ;
- les allocations dynamique et semi-statique de ressources radio (*scheduling*) ;
- le maintien de la synchronisation sur le lien montant ;
- la priorisation des flux sur le lien montant.

La fonction de scheduling s'appuie sur les mesures effectuées par la couche physique, tandis que le mécanisme HARQ est couplé avec le codage de canal. Ces fonctions sont donc étroitement liées à la couche physique et sont optimisées pour cet interfonctionnement.

Elle assure le mappage des données entre les canaux logiques et les canaux de transport en utilisant une fonction de multiplexage de RLC et assure aussi l'ordonnancement.

a. Ordonnancement

L'eNodeB [39] possède un ordonnanceur pour contrôler les ressources comme le temps et la fréquence pour un temps donné. Dépendamment de l'état des canaux, l'ordonnanceur choisit le meilleur multiplexage pour l'UE en prenant toujours en considération:

- Les paramètres de qualité de service (QoS);
- Les données utiles dans la mémoire tampon;
- La capacité de l'UE;
- Les cycles de mise en veille de l'UE;
- Les informations du système telles que la bande passante et le niveau d'interférence.

b. La technique de retransmission HARQ en LTE ([38] [36] [37] [39])

Le protocole HARQ (voir figure 3.4) est le noyau de notre étude. Le HARQ utilise un protocole d'arrêt et d'attente (*Stop-and-Wait*). Quand une transmission a été faite, l'entité émettrice arrête et attend jusqu'à ce qu'il reçoive un accusé de réception (ACK) ou accusé de réception négatif (NACK) retour de la destination avant de transmettre le prochain bloc de données ou de retransmettre le même bloc de données. Dans les deux cas (ACK ou NACK), l'entité émettrice est nécessaire pour planifier et traiter la prochaine transmission dans un délai précis. Pour la norme LTE en FDD sur la liaison montante, il a été fixé à huit (08) sous-trames de 1 ms. Etant donné qu'il suffit d'une seule sous-trame pour transmettre les données, il en

résulte en sept sous-trames de la bande passante non utilisée. Pour utiliser pleinement cette bande passante, LTE utilise plusieurs processus HARQ parallèles décalées dans le temps les uns des autres. Chaque processus transmet un bloc de données. Au moment où l'allocation de transmission suivante arrive, elle aura déjà reçu l'ACK ou NACK à l'entité de réception et créé le paquet suivant pour la transmission (retransmission). Il y a une seule entité HARQ par UE avec huit (08) processus Stop-and-Wait pour chaque entité HARQ. Le CC-HARQ et éventuellement, IR-HARQ sont aussi pris en charge en LTE. Le nombre de retransmissions HARQ ciblés par le protocole HARQ dépend de la configuration du réseau. C'est un mécanisme de retransmission de la couche MAC, il peut être synchrone ou asynchrone.

Le protocole HARQ synchrone signifie que le eNodeB sait exactement quel processus HARQ et la version de redondance (RV) que l'utilisateur transmettra à l'avance, il nécessite une retransmission à des instants connus et par conséquent il n'a pas besoin de signalisation explicite. Par contre, pour le HARQ asynchrone (c'est à dire que le récepteur ne sait pas à l'avance ce qui est en cours de transmission ou quand) de sorte que l'identificateur de processus HARQ et la version de redondance doivent être envoyés avec les données, une signalisation explicite est obligatoire (comme le numéro de processus HARQ est nécessaire pour transmettre chacun des paquets de retransmission). Le RV spécifie la combinaison de données, ED, et les bits FEC envoyés à l'utilisateur.

Le mécanisme HARQ peut être aussi adaptatif et peut donc changer la modulation, l'allocation des blocs de ressource et la durée de la transmission.

Le mode synchrone nécessite moins de signalisation et il est avantageux lorsqu'il est adaptatif. Ce mode est choisi pour le lien ascendant, tandis que pour le lien descendant le mode asynchrone adaptatif est retenu.

Dans l'adaptative HARQ, les paramètres de transmission tels que le taux de codage, le nombre d'allocation des ressources, et ordre de modulation peuvent changer au cours des retransmissions.

Nous obtenons plus de flexibilité dans la programmation en ajoutant ces attributs avec adaptations à l'état du canal variable. Ainsi, le gain d'ordonnancement peut être atteint facilement. Cependant, il y a plus de frais généraux de contrôle unique pour HARQ non adaptatif puisque les paramètres de la transmission seront informés au récepteur à toutes les retransmissions. En non-adaptative HARQ, le format de paquet de retransmission

ne change pas. Par conséquent, aucune surcharge de commande du signal n'est nécessaire, mais difficile à obtenir un gain d'ordonnancement en raison de ses caractéristiques fixées dans des formats de paquets.

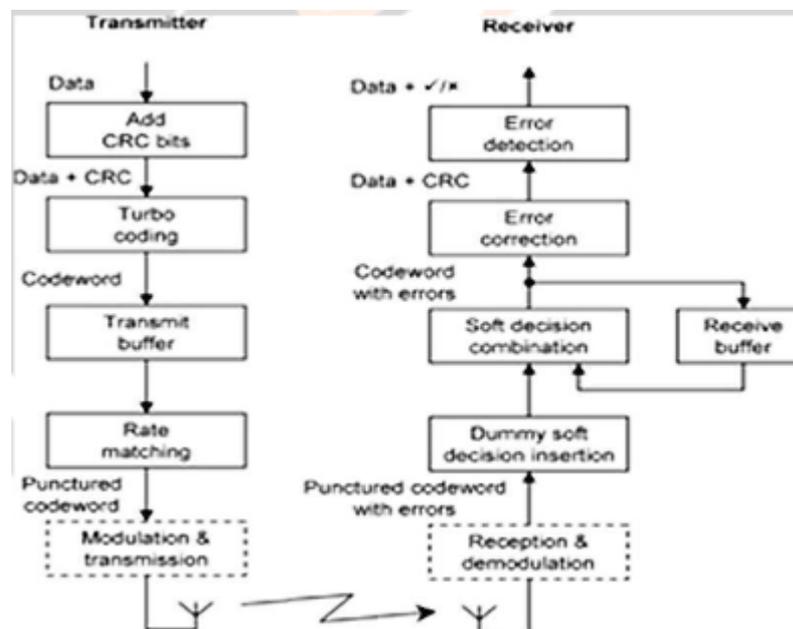


Figure 3.4. Diagramme en blocs de transmission utilisant le HARQ [37]

La station de base et le contrôleur de réseau radio sont fusionnés dans un seul nœud nommé eNodeB, afin d'éliminer le délai de transmission entre la station de base et le contrôleur de réseau radio. D'autre part dans E-UTRAN, le protocole Stop-and-wait-HARQ au niveau de la couche MAC traite les erreurs de transmission sans faire appel au protocole ARQ. Les protocoles HARQ et ARQ interagissent uniquement pour traiter les erreurs HARQ résiduelles. La figure 3.5 explique la façon d'envoyer les données entre ces deux entités en utilisant les techniques ARQ et HARQ selon les besoins: au niveau d'entité eNodeB, elle commence d'utilisée la couche RLC pour envoyer la trame vers sa couche MAC, en passant par un temporisateur qui sert à garder temporairement les trames avant de les envoyer, et au niveau de la couche MAC elle utilise un ordonnanceur afin de bien recevoir en ordre les paquets, après elle renvoie les trames vers l'UE qui passe par les mêmes étapes afin de les recevoir, en utilisant à la nécessité les protocoles ARQ et HARQ.

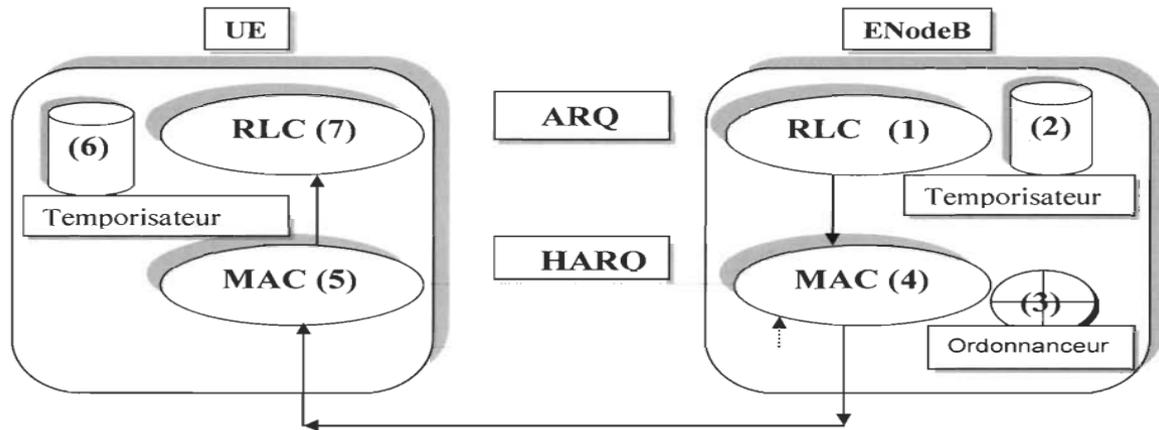


Figure 3.5. Le diagramme d'interaction pour HARQ-ARQ [39].

3.3.2. La couche physique du LTE

Le rôle de la couche physique (couche 1) [40] est d'assurer la transmission des données sous une forme capable de se propager dans l'air et de résister aux différentes perturbations inhérentes au canal radio mobile. D'un point de vue fonctionnel, la couche physique offre un service de transport sur l'interface air à la couche MAC. La couche physique réalise les fonctions suivantes pour la transmission de données :

- le codage de canal, qui protège les bits d'information contre les erreurs de transmission, en introduisant de la redondance dans la séquence de bits transmis
- la modulation, qui associe les bits à transmettre à des symboles de modulation capables d'imprimer une onde électromagnétique
- les traitements spatiaux dits MIMO, qui précèdent les symboles de modulation afin de les transmettre de plusieurs antennes (par exemple pour donner une direction au signal émis)
- la modulation multi porteuse, qui associe le signal à transmettre sur chaque antenne à des porteuses multiples, selon l'OFDMA pour la voie descendante et la SC-FDMA en voie montante.

Les opérations inverses sont effectuées par la couche physique en réception, ainsi que des traitements de lutte contre l'interférence (par exemple l'égalisation). En outre, la couche physique assure des fonctions n'impliquant pas de transmission de données, mais nécessaires à son fonctionnement, ainsi qu'à certaines fonctions de la couche MAC :

- les mesures radio, pour estimer le canal de transmission, la qualité du signal de la cellule servante, ou encore les niveaux de puissance reçus d'une autre cellule, ou d'un autre système radio
- la synchronisation, afin d'acquérir et de maintenir la synchronisation en temps et fréquence avec la porteuse de l'émetteur
- la détection de cellule, afin de détecter la présence des cellules et de s'y connecter, à l'allumage de l'UE ou pour préparer un handover
- la signalisation d'informations de contrôle entre eNodeB et UE.

3.3.3. Les canaux logiques du LTE avec HARQ

Divers canaux offrent des fonctionnalités HARQ en Up Link (UL) comme en Down Link (DL) tel que :

a. Les canaux physiques [41] [40]

Un canal physique est un canal de la couche physique qui fournit le moyen de transmettre par radio les données issues de la couche transport. Un canal physique va donc correspondre à un sous ensemble constitué d'éléments de la ressource radio disponible. Généralement, on lui associe certaines caractéristiques radios particulières.

Les canaux physiques fournissent le moyen de transmettre par radio les données des canaux de transport :

- PDCCH en DL: sert à la transmission d'information de contrôle, schéma de modulation, codage, allocation des ressources pour le DL-SCH et le PCH en downlink, allocation de

ressources HARQ pour le UL-SCH. Il est placé dans les premiers symboles OFDM de chaque sous trame ...etc

- PHICH en DL : sert à la transmission des acquittements HARQ (réponses ACK ou NACK aux transmissions HARQ d'un UE) et Il est placé dans les premiers symboles OFDM de chaque sous trame. Le HARQ information ACK ou NACK indiquant si le eNodeB a correctement reçu une transmission sur le physique UplinkShared CHannel (*PUSCH*). Pour les multiples PHICH différents UE sont mappés sur le même ensemble d'éléments de ressource de liaison descendante.
- PUCCH en UL : sert à la transmission des acquittements HARQ et à l'allocation de ressources.

b. Les canaux de transport [40] [42]

Un canal de transport est caractérisé par la façon dont les données sont transportées sur les ressources physiques, notamment : la méthode d'accès aux ressources radio (scheduling dynamique, semi-persistant ou statique) ; les formats de transport autorisés, qui définissent les traitements de la couche physique à appliquer (type de codage de canal, schéma de transmission MIMO, etc.) ; la possibilité d'effectuer des retransmissions d'HARQ, et si oui, de quel type. Plusieurs canaux de transport sont définis dans l'interface radio du LTE pour les sens montant et descendant.

Les canaux de transport fournis par la couche physique (dans notre cas on ne s'intéresse qu'aux canaux assurant une fonction HARQ) :

- DL-SCH en DL : pour la transmission de données dédiées (contrôle ou usager) ou d'informations cellulaires non basiques (comme la liste des cellules voisines) en downlink. Ce canal va donc porter des canaux logiques DCCH, DTCH, CCCH ou BCCH. Permet le HARQ et l'adaptation du lien, allocation dynamique ou semi-persistante des ressources.
- UL-SCH: Pour la transmission de données dédiées (contrôle ou usager) en uplink. Ce canal va donc porter des canaux logiques DCCH, DTCH, ou CCCH. Permet le HARQ et l'adaptation du lien, allocation dynamique ou semi-persistante des ressources

- HI en DL: ACK ou NACK après une transmission sur le UL-SCH, associé au PHICH
- DCI en DL: informations de contrôle véhiculées sur le PDCCH (formats de transport, allocations de ressources, informations HARQ et MIMO, commandes de contrôle de puissance) UCI (*UL Control Indicator*) en UL: contrôle véhiculé sur le PUCCH (qualité du canal CQI, informations MIMO boucle fermée PMI/RI, ACK/NACK de l'HARQ, requêtes de scheduling).

3.4. La technologie HARQ dans le Wimax

L'utilisation de l'ARQ avec le soutien du FEC est connue comme HARQ. La méthode HARQ utilise non seulement les paquets retransmis à reconstituer, les paquets libres d'erreur d'origine, mais elle a aussi utilisé les paquets reçus avec des erreurs. Le paquet original peut être reconstruit par une combinaison de plusieurs versions de paquets avec des erreurs. Le HARQ décrit dans (IEEE802.16e, 2006) utilise deux types différents de reconstruction: La combinaison de Chase (*CC*) et incrémental Redondance (*IR*).

Dans la version du HARQ notée CC-HARQ, des blocs de données avec un code CRC sont codés en utilisant un codeur FEC avant la transmission.

Si la qualité du canal est faible et les erreurs de données sont identifiées, le bloc de donnée n'est pas rejetée, cependant il est conservé dans la mémoire. Dans la phase suivante, le récepteur demande la retransmission de ce bloc de données. Le bloc réémis des données est ensuite combiné avec les blocs précédemment reçus avec des erreurs. La combinaison de plusieurs versions des blocs de données améliore la probabilité de décodage correcte, même si chacun d'entre eux est reçu avec des erreurs.

Le cas échéant, la norme IEEE802.16 supporte également le type Redondance incrémentielle HARQ (*IR-HARQ*). Dans le cas d'IR HARQ, le codeur des codages FEC subdivise un paquet en plusieurs sous-paquets chacun des sous-paquets est codé avec un rapport de code différent.

Les sous-paquets sont distingués par 2 bits SPID (*sous-paquet Identifier*). Si le paquet est transmis pour la première fois, le sous-paquet avec SPID = 00 est envoyé. Le succès de réception du paquet à la station de destination est indiqué par ACK (dans le cas contraire, un NACK) et l'émetteur doit envoyer un autre paquet portant l'un des quatre sous-paquets. Deux paquets reçus (La première transmission et retransmissions) sont à nouveau réunis par le récepteur pour augmenter la probabilité d'un décodage correcte [45].

Lors de la couche physique du Wimax, l'Adaptive modulation and Coding (AMC), Hybrid Automatic Repeat Request et Fast Channel Feedback (CQICH) ont été introduits dans la version mobile du Wimax pour améliorer les performances et la couverture. Le support de la QPSK, 16QAM et 64QAM est indispensable dans le sens descendant alors que dans le sens ascendant, la 64QAM est optionnelle. Les codes Convolutional et Convolutional Turbo Code avec un taux de codage variable sont aussi dans la norme.

Le HARQ est supporté par le Wimax Mobile. L'utilisation de technique HARQ permet de réduire les pertes de paquets. En rappelant que le HARQ est une technique permettant de combiner la technique FEC avec une stratégie de retransmission permettant l'échange d'informations entre l'émetteur et le récepteur.

Lors du Wimax Mobile, le HARQ utilise N canaux Stop-and Wait permettant de fournir une réponse rapide dans le cas d'erreur sur les paquets et aide à améliorer la couverture de la cellule. Le CC-HARQ et éventuellement, IR-HARQ sont pris en charge pour améliorer encore la fiabilité de la retransmission. UN ACK canal dédié est également prévu dans la liaison montante pour l'opération de la signalisation HARQ ACK / NACK.

Le HARQ Combiné au CQICH et à l'AMC permet d'assurer un lien robuste dans le cas d'environnement mobile pour des vitesses dépassants les 120 Km/h [35].

Remarque : plusieurs processus HARQ permettent l'utilisation efficace du temps de traitement et de transmission des paquets. Si un paquet d'un processus est transmis, il n'est pas nécessaire d'attendre sa confirmation. Un paquet d'un autre processus peut ainsi être immédiatement transmis car l'émetteur sait que la confirmation du succès ou de l'échec de la réception du paquet n'interviendra qu'ultérieurement. L'efficacité se trouve encore accrue par ce processus qui ne rejette pas complètement les transmissions erronées mais les combine à

l'aide d'algorithmes intelligents aux autres transmissions du même bloc de données (utile), ce qui augmente la probabilité d'un décodage final réussi du paquet.

3.5 Technique d'accès au canal SC-FDMA

3.5.1 Rappel sur la technique Single Carrier FDE : SC/FDE

A côté des systèmes multi-porteuses, le système mono-porteuse ; plus adapté aux canaux sélectifs a aussi fait ses preuves. Il s'agit de la transmission mono porteuse avec égalisation dans le domaine fréquentiel SC/FDE. Ce système associe la technique mono porteuse avec une technique d'égalisation fréquentielle mise en place au niveau récepteur afin de combattre la sélectivité du canal (voir figure 3.6) La principale différence de ce système avec la transmission mono-porteuse est le traitement par bloc qu'il met en œuvre. En effet les symboles ne sont plus directement envoyés de façon séquentielle dans le canal comme dans la transmission mono-porteuse, mais sont groupés d'abord en paquet avant toute transmission. Chaque paquet se voit ensuite augmenté d'un intervalle de garde ou CP placé en entête, qui correspond à la copie de la fin du paquet. Le rôle de l'intervalle de garde est double : Il permet d'une part d'absorber l'interférence entre paquets, mais également de périodiser le signal transmis pour transformer la convolution linéaire du canal en une convolution circulaire. Par ailleurs, nous savons qu'une convolution circulaire dans le domaine temporel est équivalente à une simple multiplication dans le domaine fréquentiel. La représentation fréquentielle du paquet SC/FDE récupéré à la réception correspond donc à la multiplication élément par élément entre les représentations fréquentielles du canal et du paquet émis. Une égalisation fréquentielle FDE est ainsi plus appropriée. Son avantage étant d'une part sa simplicité de mise en œuvre, mais également sa robustesse par rapport à la sélectivité du canal. Ceci se vérifie, lorsque le coefficient du canal est nul sur une porteuse donnée.

En effet, seul le symbole affecté par ce coefficient sera perdu, le reste du paquet pourra être démodulé ce qui n'était pas le cas de la transmission mono-porteuse [22].

NB : rappelons que le SC-FDMA est une extension de Single Carrier avec *Frequency Domain Equalization* (SC/FDE).

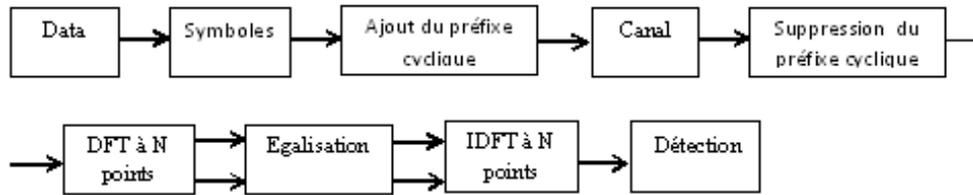


Figure 3.6. Schéma de principe d'une transmission SC FDE

3.5.2 Principe de la SC-FDMA

Comme pour d'autres techniques à schéma d'accès multiples (TDMA, FDMA, CDMA, OFDMA), le but est l'attribution et le partage d'une ressource radio (bande de fréquence) entre plusieurs utilisateurs. Le SC-FDMA peut être considéré comme une variante linéaire des codages OFDM et OFDMA, dans le sens où il consiste aussi à répartir sur un grand nombre de sous-porteuses (plusieurs centaines) le signal numérique ; il impose aussi un écart de fréquence entre les sous-porteuses égales à la fréquence des symboles ce qui garantit l'orthogonalité des sous-porteuses et permet une plus grande efficacité spectrale, mais il utilise, en plus, une transformation de Fourier discrète (DFT) du signal pour pré-coder l'OFDMA conventionnel. Le principe de répartition du signal numérique à transmettre en sous-porteuses orthogonales permet de limiter les problèmes d'interférences inter-symboles et de l'évanouissement (*fading*) qui est une forte atténuation du signal liés aux trajets multiples de propagation qui existent dans les liaisons radio de moyenne et longue portées car quand le débit binaire sur une porteuse est élevé, l'écho d'un symbole arrivant en retard à cause d'une propagation multi-trajets qui perturbe le ou les symboles suivants. La figure 3.7 décrit l'utilisation des sous-porteuses orthogonales.



Figure 3.7. Principe des sous-porteuses orthogonales [24]

Les porteuses en bleu, vert et noir transportent les données des utilisateurs, celles en rouge transportent les informations de signalisation entre l'équipement mobile et le réseau. L'écart entre chaque sous-porteuse est donné par la relation :

$$\Delta f = \frac{K}{T_U} \text{ (Hertz)} \quad (3.1)$$

Avec T_U (seconde) est la durée utile d'un symbole (c'est-à-dire la taille de la fenêtre de capture du récepteur) et k un entier positif, généralement égal à 1 [24].

Un filtrage séparé de chaque sous-porteuse n'est pas nécessaire pour le décodage dans le terminal récepteur, une ' Transformée de Fourier ' FFT est suffisante pour séparer les sous porteuses l'une de l'autre [29].

3.5.3 Transmission des signaux en SC-FDMA

La modulation SC-FDMA est une technique de transmission mono-porteuse qui consiste à répartir sur un grand nombre de porteuses la représentation fréquentielle des symboles après les avoir réparties sur la bande du système. La Figure 3.8 montre la chaîne de transmission d'un tel système. Elle est constituée de trois parties principales :

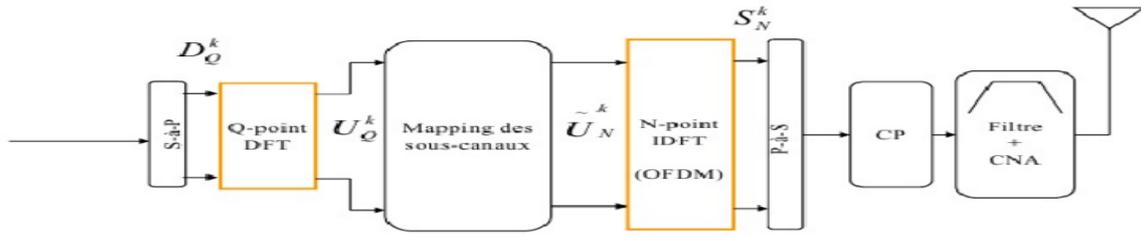


Figure 3.8. Emetteur SC-FDMA [24].

On a d'abord un modulateur DFT dont l'entrée constitue le bloc de symboles source de la modulation QAM de l'utilisateur k prenant ses valeurs dans un alphabet fini. Ce modulateur transforme le bloc de symbole D d'utilisateur k en Q symboles fréquentiels qui viennent ensuite moduler un ensemble de Q sous-porteuses choisies parmi N . Cette répartition des symboles fréquentiels a pour objectif l'étalement spectral du signal et le multiplexage fréquentiel des signaux des différents utilisateurs. La sortie du multiplexeur est ramenée dans le domaine temporel par un modulateur IDFT (DFTinverse) pour donner les symboles de sortie. La deuxième partie de la chaîne de transmission s'agit de l'insertion de l'intervalle d'un préfixe cyclique (CP). Elle consiste à insérer à l'entête de chaque bloc à émettre, la copie de la fin du bloc. Le but est d'abord de réduire l'interférence entre les blocs émis introduite par les multi-trajets du canal, mais également de rendre circulaire la convolution du canal afin d'utiliser des techniques d'égalisation fréquentielle simples à mettre en œuvre. Le principe de la démodulation du système SC-FDMA consiste à démoduler le signal sur chacune des sous-porteuses du système.

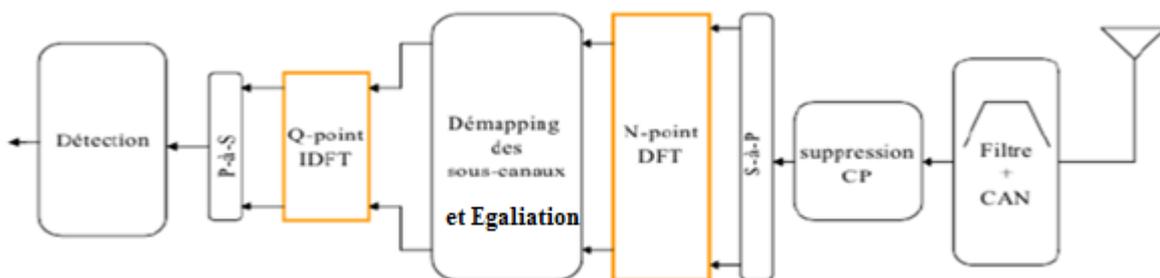


Figure 3.9. Récepteur SC-FDMA [24].

En effet, le signal reçu est d'abord ramené en bande de base, avant d'être échantillonné pour le traitement numérique du signal. Après suppression de l'intervalle de garde, un démodulateur DFT (voir figure 3.9) permet d'obtenir les symboles modulant chaque porteuse. Un égaliseur est ensuite mis en œuvre dans le but de supprimer la contribution du canal sur chaque sous-porteuse du signal, et de récupérer ainsi les symboles fréquentiels (puisque le SC-FDMA utilise la modulation mono porteuse et donc soumis aux interférences inter-symbole alors un égaliseur est nécessaire afin de combattre l'ISI contrairement au CP qui combat l'interférence entre bloque [25]. Les symboles égalisés sont transformés via un démodulateur IDFT en domaine temporel permettant une détection et un décodage dans le dit domaine afin de récupérer les symboles source du système [24].

3.5.4 Allocations des sous porteuses

Le SC-FDMA offre des performances et une complexité globale similaire à l'OFDMA mais utilise en revanche une technique de multiplexage fréquentiel à une seule porteuse. Il utilise deux méthodes pour associer les données aux sous-porteuses (*subcarrier mapping*): l'accès FDMA entrelacé (IFDMA) ou FDMA localisé (LFDMA), un émetteur génère des symboles de modulation pour des types de données différents (par exemple, des données de trafic, la signalisation, et les pilotes) [26]. On dénombre les modes d'allocation des sous porteuses par rapport aux symboles d'information. Soit N le nombre total des sous-porteuses disponibles dans la bande entière du système SC-FDMA et M le nombre de sous-porteuses dédiées à un utilisateur [27].

- Le mode localisé : chaque utilisateur occupe un ensemble de M sous-porteuses adjacentes parmi les N sous-porteuses [27]. Aucun zéro n'est inséré entre les échantillons de sortie de la DFT, ceux-ci étant donc transposés sur des sous-porteuses consécutives [26].
- Le mode distribué : à chaque utilisateur sont attribuées M sous porteuses parmi N d'une manière distribuée. Les sous-porteuses occupées sont dispersées sur la bande passante disponible [27]. Un certain nombre de zéros est inséré entre chaque échantillon de sortie de la DFT, offrant une plus grande diversité fréquentielle [26].

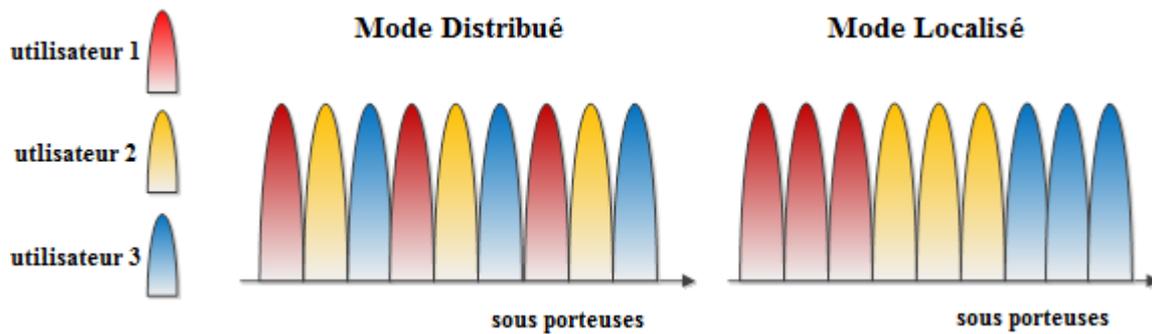


Figure 3.10. Mappage des sous porteuses avec $N=9$, $M= 3$ et 3 utilisateurs.

Dans la figure 3.10 et dans le cas d'une transmission mono utilisateur, exemple seulement l'utilisateur 1 (rouge) qui transmet alors les autres utilisateurs 2 (jaune) et 3 (bleu) sont remplacés par des zéros (voir figure 3.11)

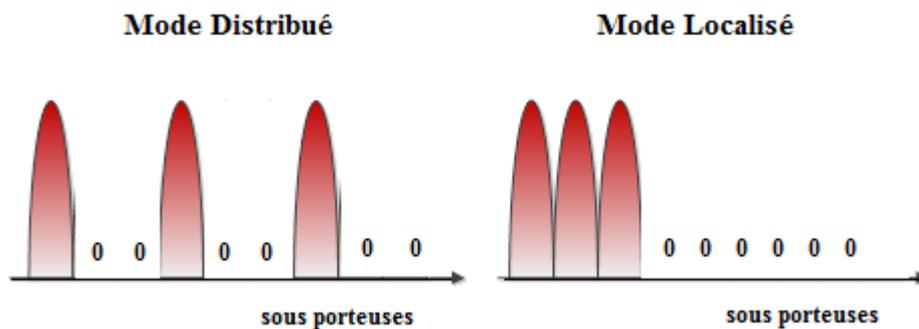


Figure 3.11. Mappage des sous porteuses avec $N=9$ et $M= 3$ et un seul utilisateur

Les symboles transmis en mode distribué sont dispersés sur la bande passante disponible ce qui peut apporter un gain de diversité en fréquence. D'un autre côté, la transmission SC-FDMA en mode localisé peut apporter une diversité d'accès multiple en présence d'évanouissements fréquentiels si chaque utilisateur transmet sur une portion de la bande où les caractéristiques de transmission lui sont favorables (*highchannel gain*). Dans ce cas, la diversité d'accès multiple résulte de l'indépendance des évanouissements entre les différents utilisateurs. Ceci nécessite une allocation qui dépend de l'état du canal de transmission CDS (*channeldependent scheduling*) [27].

3.5.5 Avantages de la SC-FDMA par rapport à l'OFDMA

Le SC-FDMA aussi appelé DFT-Spread OFDMA, consiste à transposer les symboles des données du domaine temporel dans le domaine fréquentiel par une DFT avant d'opérer la modulation OFDM. Le transmetteur convertit le signal d'entrée binaire en une suite des symboles complexe modulés, regroupé en un bloc de M symboles. Une DFT à M points permet ensuite d'obtenir une représentation fréquentielle des symboles d'entrée. Chacune des M sorties de la DFT est transposée sur une des N sous porteuses orthogonales. Enfin, une IDFT à N points permet d'obtenir un unique signal dans le domaine temporel [28]. La figure 4.4 permet de relever les points communs entre l'OFDMA et le SC-FDMA :

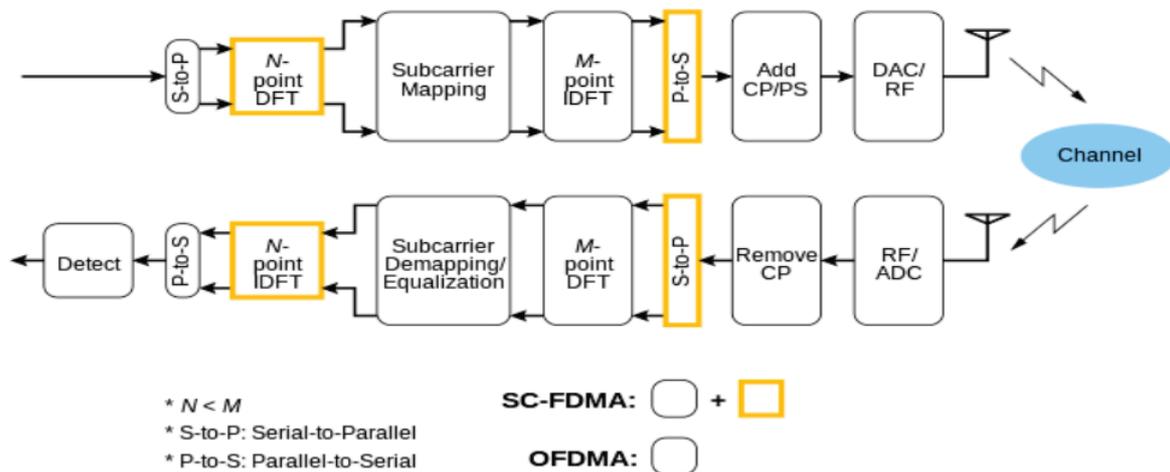


Figure 3.12. Similitude entre une chaîne OFDMA et SC-FDMA [29].

La SC-FDMA a le même principe que l'OFDMA, la différence existe sur un seul point : au lieu de mapper les symboles sur les sous-porteuses, on mappe leur DFT (présence des blocs DFT et IDFT pour la génération du signal SC-FDMA) [30].

L'envoi des données en OFDMA s'effectuant parallèlement sur plusieurs sous-porteuses, le PAPR résultant est élevé. Au contraire le SC-FDMA consiste en l'envoi des données en série sur une même porteuse, permettant de ce fait d'obtenir un PAPR moindre (bien adapter aux exigences de la LTE en *uplink*).

L'OFDMA réalise une égalisation et une détection séparément pour chaque sous-porteuse du système. Ainsi lorsque qu'aucun codage n'est réalisé dans ce système, le canal étant très

sélectif, certaines porteuses seront supprimées par le canal et il serait donc impossible de retrouver le flux de données localisé sur les porteuses supprimées [24]. La détection en OFDMA s'effectuant par sous-porteuse, un zéro sur le spectre dégrade fortement les performances. Au contraire, la détection en SC-FDMA a lieu après l'IDFT et s'effectue sur l'ensemble du signal, ce qui permet de moyenniser le SNR sur la totalité de la bande de fréquences utilisée (ce qui lui rend ainsi moins sensible à la sélectivité fréquentielle du fait que les perturbations sont moyennées sur toute la largeur de bande du système) [28].

Un autre avantage de ce système par rapport à l'OFDMA est le pré-codage réalisé à l'émission. En effet l'OFDMA réalise une égalisation et une détection séparément pour chaque sous-porteuse du système. Ainsi lorsque qu'aucun codage est réalisé dans ce système, le canal étant très sélectif, certaines porteuses seront supprimées par le canal et il serait donc impossible de retrouver le flux de données localisé sur les porteuses supprimées. Par contre dans le système SC-FDMA la détection est effectuée après le modulateur DFT à la réception qui lui rend ainsi moins sensible à la sélectivité fréquentielle du fait que les perturbations sont moyennées sur toute la largeur de bande du système comme on peut le constater sur la Figure 3.13. Le modulateur DFT se comporte donc comme un pré-codeur.

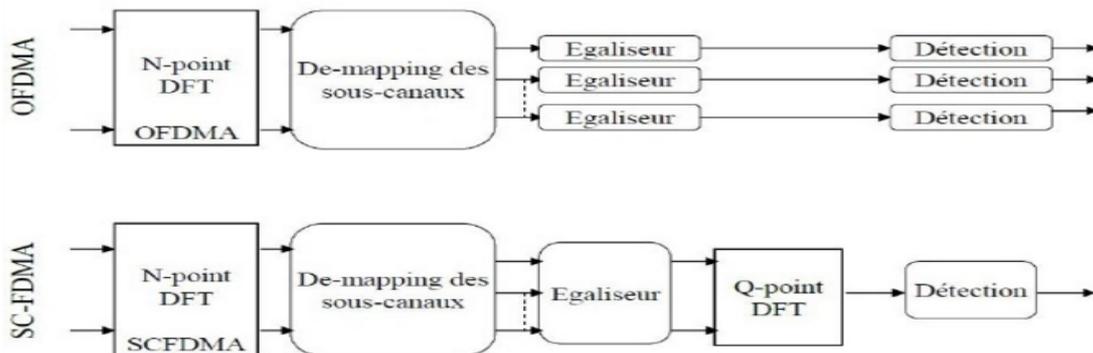


Figure 3.13. Comparaison entre la détection en SC-FDMA et OFDMA [24].

L'envoi de données en OFDM s'effectuant parallèlement sur plusieurs sous-porteuses, le PAPR (Peak-to-Average Power Ratio) résultant est élevé. Au contraire, le SC-FDMA consiste en l'envoi de données en série sur une même porteuse, permettant de ce fait d'obtenir un PAPR moindre[26].

3.6. Conclusion

Dans ce chapitre nous avons passé en revue les normes des réseaux où la technique HARQ est utilisée à savoir le HSPA, Wimax et le LTE, Il nous a permis de savoir, d'une manière générale l'importance, du protocole HARQ dans les réseaux de télécommunications du fait de son utilisation dans divers normes.

Nous nous sommes également intéressés à la technique d'accès au canal SC FDMA utilisé dans la voie montante de la 4G LTE puisque cette dernière, combiné au HARQ, sera utilisée comme chaîne de transmission pour nos tests...

Chapitre 4 Simulations : applications des techniques ARQ et HARQ en SC-FDMA

4.1 Introduction

Dans ce chapitre nous allons faire une simulation avec programmation matlab de l'implémentation du protocole ARQ et HARQ appliqués dans une chaîne de transmission SC-FDMA.

4.1.1 Paramètres de simulation

Lors de nos simulations, les tests sont faits sur une chaîne de transmission SC-FDMA en SISO dans le cas où :

- l'environnement de propagation est un canal de type Rayleigh à 4 trajets
- ajout d'un bruit additif gaussien en réception
- la taille de la DFT à l'entrée de chaîne SC FDMA est 16
- la taille de la FFT est de 64.
- la longueur du préfixe cyclique est de 10.
- le CRC a une taille de 16 bits.
- l'égalisation se fait dans le domaine fréquentielle et est du type MMSE (*Minimum Mean Square Error*). Elle consiste à multiplier les symboles à la sortie du bloc demapping des sous porteuses par la quantité :

$$C = (H^H H + \sigma^2 I)^{-1} H^H \quad (4-1)$$

où H est la matrice du canal dans le domaine fréquentielle et σ^2 la variance du bruit et l'opérateur $(.)^H$ désigne la transposée hermitienne.

Deux cas ont été simulés :

- sans codage canal (ARQ).
- avec codage canal (HARQ) et

Avec :

- schémas de constellation : 16 QAM et 4 QAM
- seuil d'erreur : 80 et 40 bits erronés par trame de 512 bits.
- SNR fixé à 6 dB.

Les trames émises sont placées dans un buffer avant leur traitement par le processus SC-FDMA et ce pour les retransmettre en cas d'erreurs. Ces trames seront retransmises dans le cas du HARQ ou ARQ de manière continue jusqu'à ce que les bonnes trames soient reçues (nombre d'erreur inférieur au seuil d'erreurs fixé soit atteint). Il n'y a donc pas une limite fixe du nombre de retransmissions.

4.2 Les différents cas de simulation

4.2.1. Simulations sans codage canal

Dans ce cas la détection d'erreur de type ARQ. Nous avons scindé cette partie en deux

- a) Dans la première partie on considère une transmission sans codage canal avec une modulation 16 QAM, un seuil d'erreur fixé à 80 bits ou à 40 bits par trame de 512 bits. Les résultats obtenus sont illustrés par les figures 4.1 et 4.2

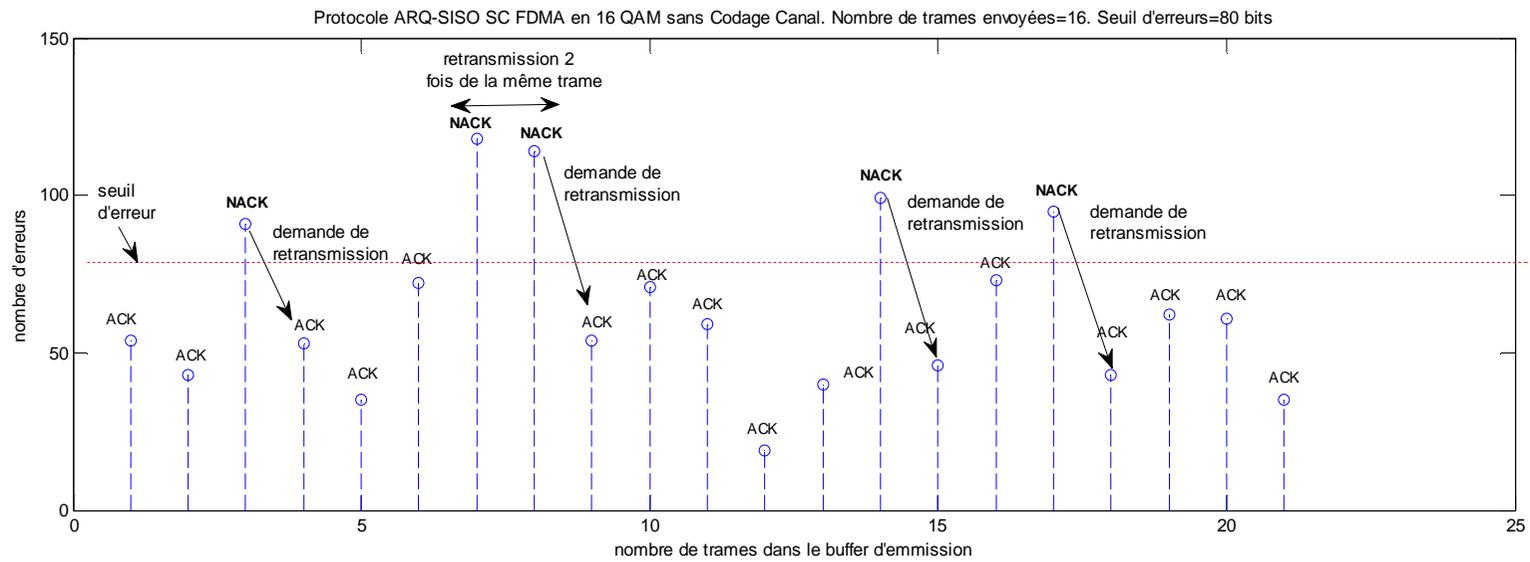


Figure 4.1. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA : Taille de la trame $T=512$ bit, nombre de trames envoyées $N_T=16$; $SNR=6$ dB et seuil d'erreurs fixé à $\text{seuil}=80$ bits (erronés) par trame ($BER = 0,1564$).

Interprétation :

Comme on peut le distinguer, on remarque que si la trame envoyée (d'abord mise dans le buffer pour la renvoyer en cas de réception d'un NACK) est reçue avec un nombre de bit erroné inférieur au seuil fixé à 80 bits ($BER=0,1564$) elle sera acceptée, c'est-à-dire qu'elle est considérée sans erreur, et un accusé de réception positif (ACK) est ensuite envoyé à l'émetteur pour l'informer de l'état de la trame. Une fois l'acquiescement est reçu et vérifié par l'émetteur la prochaine trame est envoyée et le processus d'envoi des nouvelles trames va continuer jusqu'à la réception d'un accusé de réception négatif (NACK) qui est déclenché par un seuil d'erreur supérieur au seuil fixé. Dans ce cas-là la trame sera retransmise au récepteur et le processus continu jusqu'à réception de la bonne trame.

Le nombre de retransmission n'est pas fixé dans notre cas alors qu'en réalité il est fixé grâce à un temporisateur (TIMEOUT). Dans nos simulations la partie temporisation n'est pas simulée.

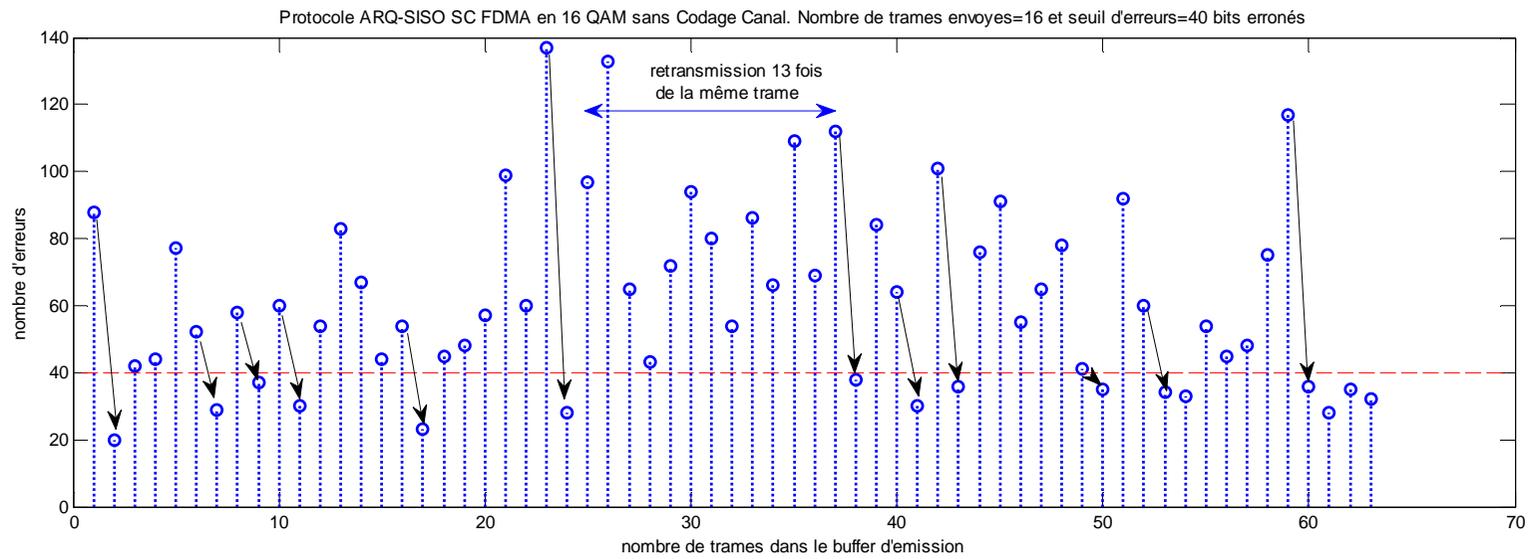


Figure 4.2. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA: $T = 512$ bit, $NT=16$; $SNR=6$ dB et seuil= 40 bits (erronés) par trame (BER =0,07812).

Interprétation :

Dans cette partie on considère une transmission sans codage canal avec une modulation 16 QAM mais cette fois ci avec un seuil d'erreur plus sévère fixé à 40 bits par trame. Les résultats obtenus sont illustrés par la figure 4.2. On constate une similitude avec le premier cas, la seule différence réside au niveau du seuil d'erreur fixé à 40 bits c'est-à-dire (BER_{0,07812}) ce qui engendre, par rapport au premier cas, une augmentation de retransmission des trames .On peut constater, par exemple, jusqu'à 13 retransmissions de la même trame de donnée. Donc d'après notre constat, plus le seuil d'erreur fixé est élevé moins il y a de retransmission mais la qualité de la transmission devient mauvaise.

b) Lors de cette deuxième partie des simulations on considère une transmission sans codage canal mais avec une modulation 4 QAM, un seuil d'erreur fixé qui varie entre 80 ou 40 bits .Les figures 4.3 et 4.4 montrent les résultats obtenus. Tous comme les remarques faites lors de la première partie, plus le seuil d'erreur fixé est élevé plus la transmission est favorisée du fait que let moins il y a de trames perdues mais la qualité de la transmission se dégrade.

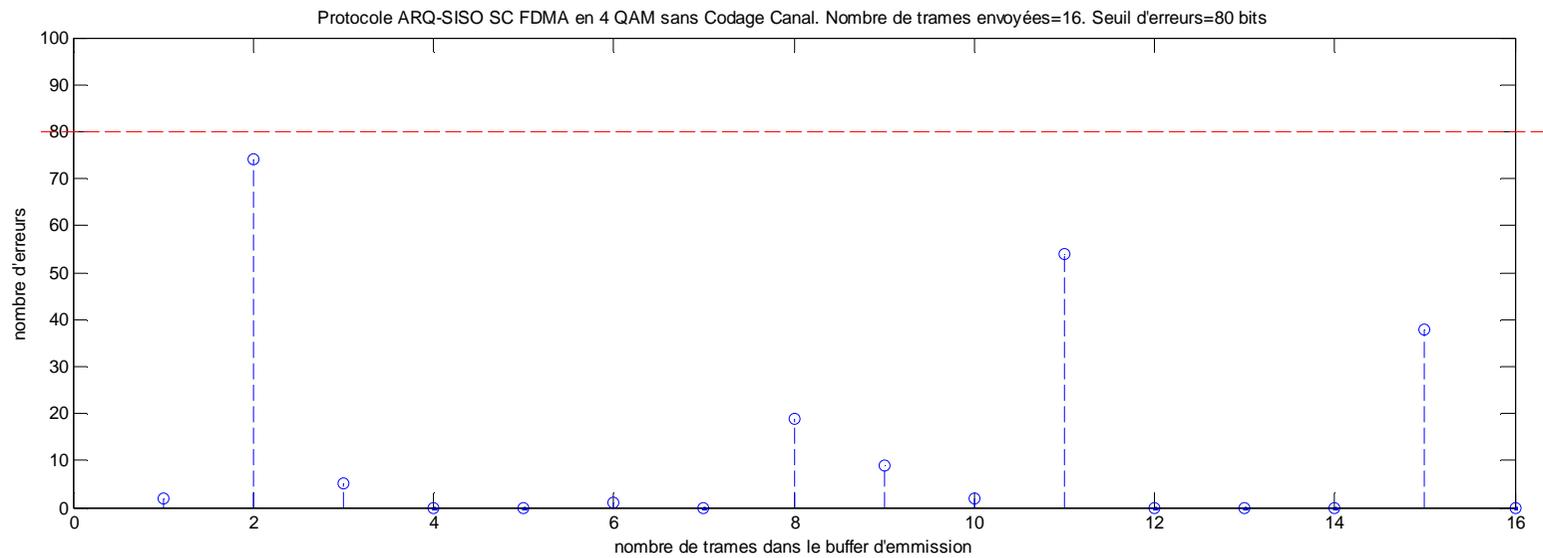


Figure 4.3. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA : $T = 512$ bit, $NT = 16$; $SNR = 6$ dB et seuil = 80 bits (erronés) par trame (BER = 0,1564).

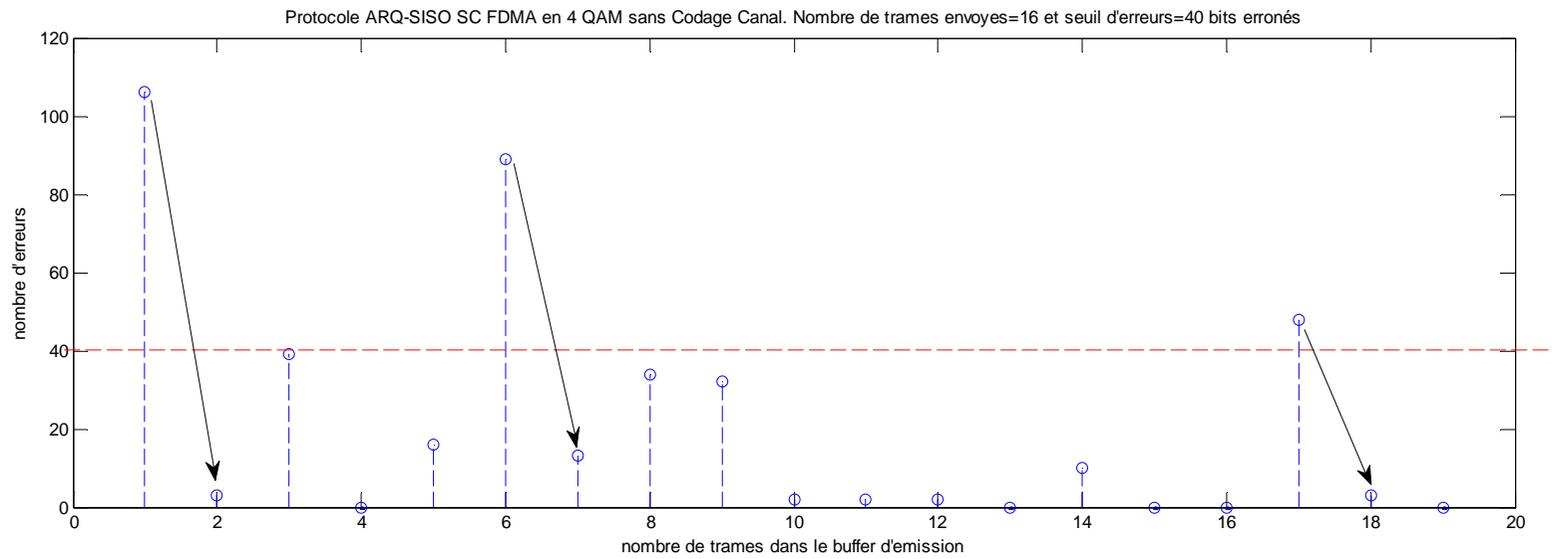


Figure 4.4. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA. $T = 512$ bit, $N_T = 16$, $SNR = 6$ dB et seuil = 40 bits (erronés) par trame (BER = 0,07812).

Comparaison entre les deux cas de simulation (a et b).

Les deux cas de simulation utilisent le même principe de fonctionnement, la seule différence réside de la variation de l'ordre de modulation. Pour la première partie la 16 QAM est utilisée par contre la deuxième partie utilise la 4 QAM. Pour les deux seuils d'erreurs fixés (80 ou 40 bits) la deuxième partie donne des meilleurs résultats du point de vue de la perte des données. Cela nous amène à dire que plus l'ordre de modulation est faible, plus on a des meilleurs résultats du fait qu'on assiste à une diminution brusque des retransmissions.

4.2.2. Simulations avec codage canal

Dans ce cas on est en présence d'une transmission en mode HARQ (combinaison entre l'ARQ et le FEC):

Cette partie est identique à la partie sans codage canal (§ 4.2.1) sauf que dans ce cas une transmission avec codage canal est prise en considération.

Les schémas de constellation sont de type 16 QAM et 4 QAM et le seuil d'erreur fixé à 80 ou à 40 bits. Dans ce cas les résultats obtenus sont illustrés par les figures 4.5, 4.6, 4.7 et 4.8.

En ce qui concerne la variation du seuil d'erreur (*BER*) et la variation de l'ordre de la modulation ou de la constellation, les interprétations de la partie sans codage canal sont aussi valides. Nous remarquons que l'apport du codage canal améliore les performances de la transmission et ce en minimisant le nombre de retransmission du protocole ARQ et même pour un faible seuil d'erreur, d'où l'avantage de la technique HARQ.

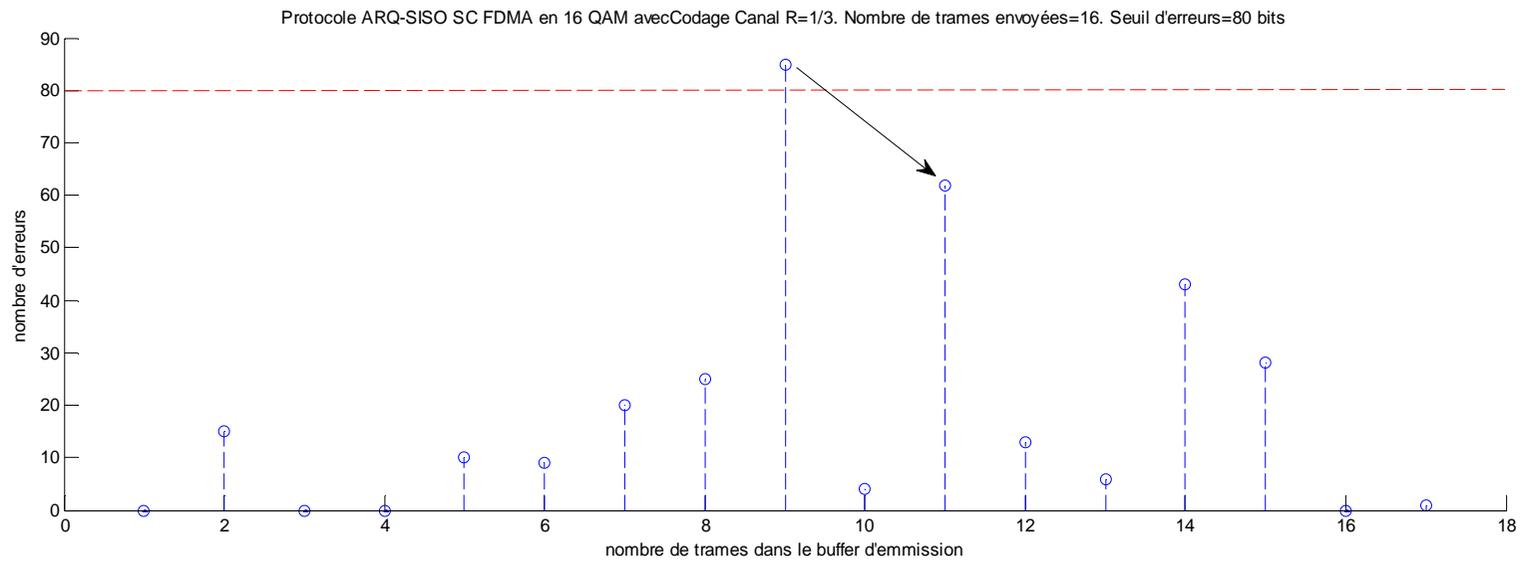


Figure 4.5. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA. $T = 512$ bit, $N_T = 16$, $SNR = 6$ dB et seuil= 80 bits (erronés) par trame (BER = 0,07812).

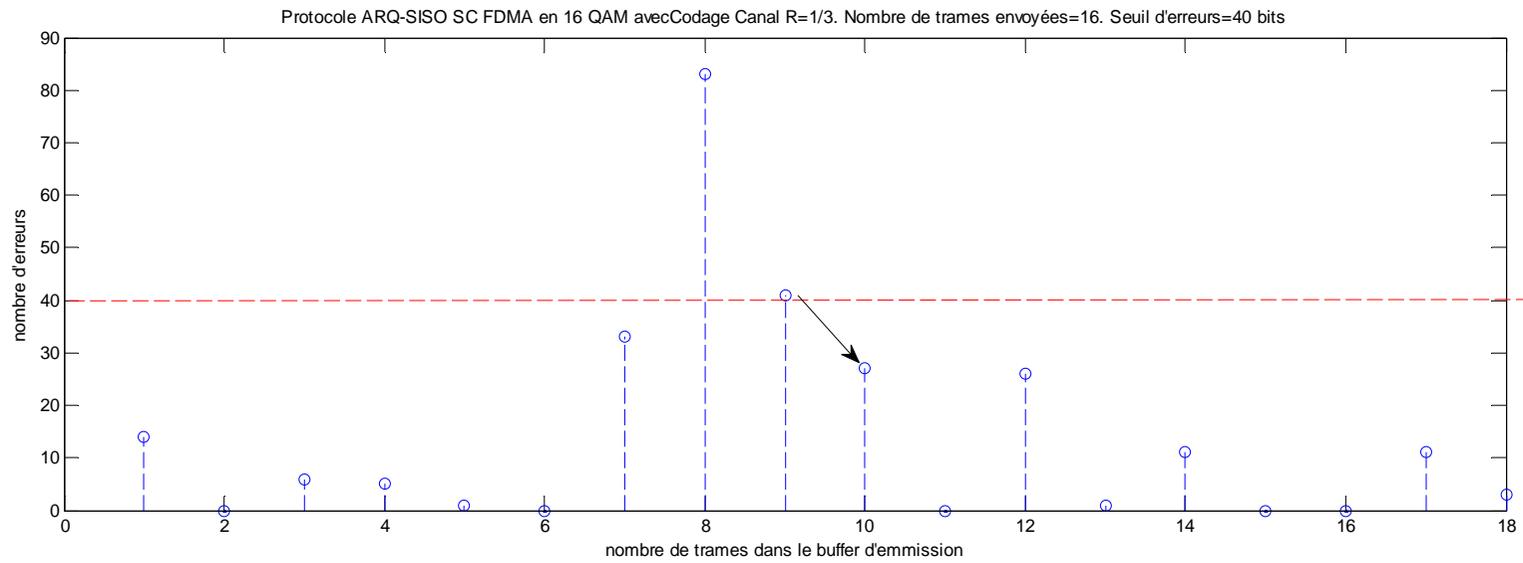


Figure 4.6. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA. $T = 512$ bit, $NT=16$, $SNR=6$ dB et seuil= 40bits (erronés) par trame (BER =0,07812).

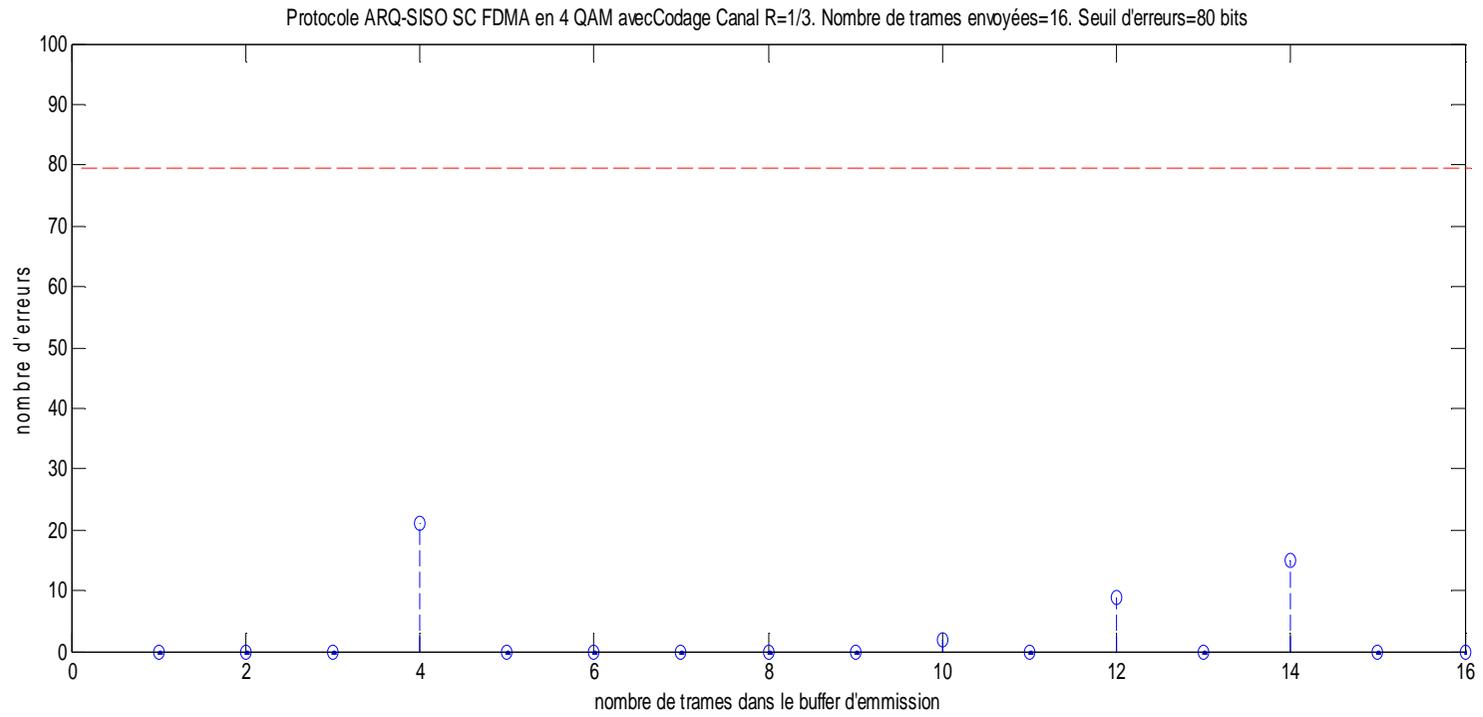


Figure 4.7. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA. $T = 512$ bit, $NT=16$, $SNR=6$ dB et seuil= 80 bits (erronés) par trame (BER =0,07812).

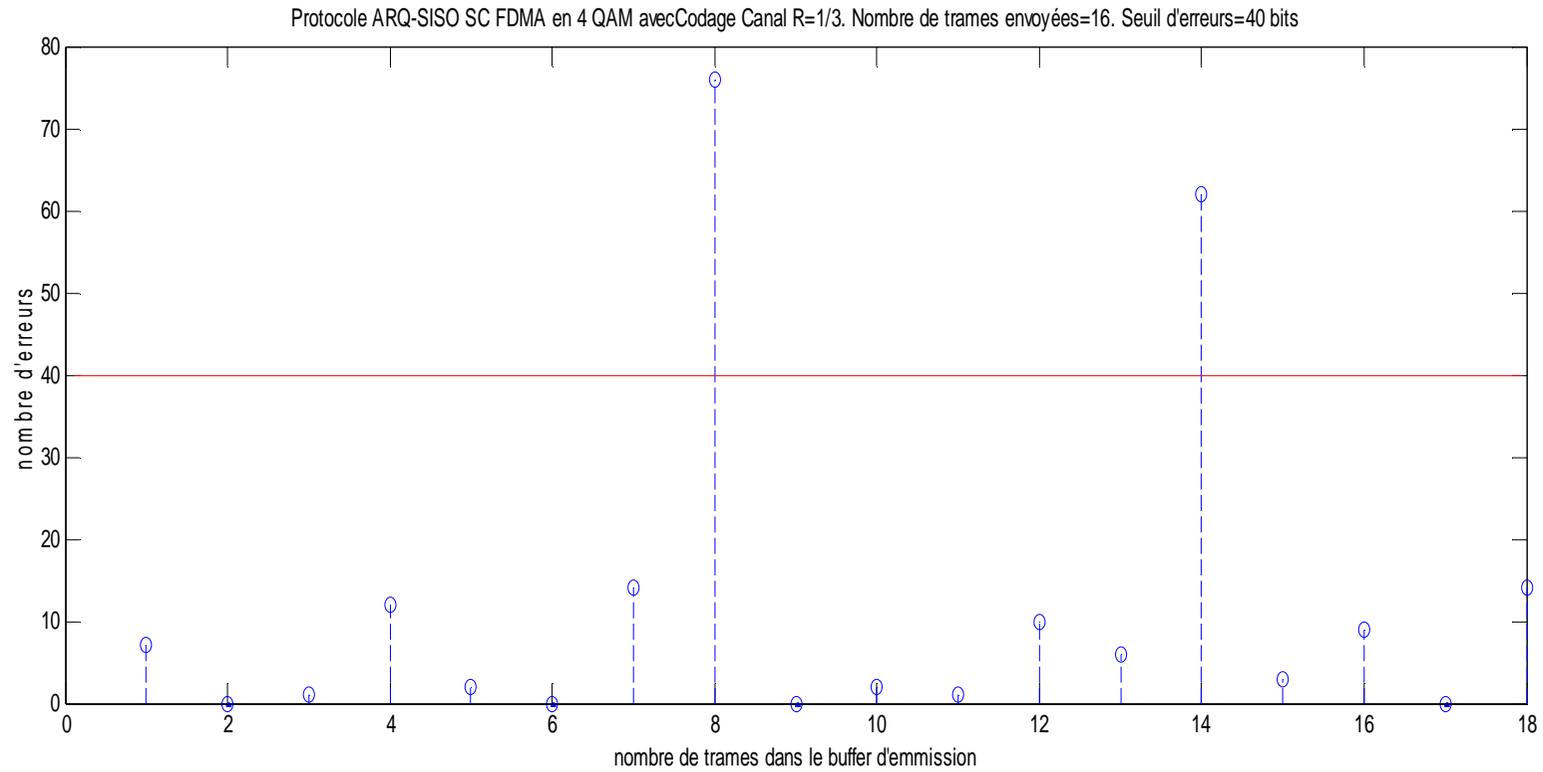


Figure 4.8. Exemple d'application du protocole ARQ en transmission SISO SC- FDMA. $T = 512$ bit, $N_T = 16$, $SNR = 6$ dB et seuil= 40bits (erronés) par trame (BER =0,07812).

4.3. Conclusion

Ce chapitre nous a permis d'une manière générale de savoir le principe de fonctionnement d'une chaîne de transmission SC-FDMA, les avantages de celle-ci par rapport à l'OFDMA ont aussi été abordés et nous a permis aussi de faire une étude de la simulation des protocoles ARQ et HARQ dans une chaîne de transmission SC-FDMA.

Conclusion générale

Tout au long de ce mémoire, nous avons eu à faire : une étude générale sur les techniques directes de détections et correction d'erreurs FEC, l'étude des techniques se basant sur les retransmissions des paquets erronés ARQ et HARQ, une description de la technique SC-FDMA, et enfin notre travail se termine par l'interprétation des résultats obtenus suite à la programmation sur matlab. Les simulations pour le calcul des erreurs de transmission se sont faites pour un rapport signal sur bruit (SNR), où le bruit est du type additif gaussien fixe et ce dans un environnement de propagation constitué par un canal de type Rayleigh à quatre (04) trajets. Les simulations ont eu pour objectif de mettre en évidence les performances des technique ARQ et HARQ.

Nos simulations nous ont permis de distinguer que la technique HARQ améliore les performances de la transmission par rapport à la technique ARQ, du fait qu'elle minimise le nombre de retransmission de cette dernière et ceux même pour des faibles valeurs du BER. Mais aussi la technique HARQ est plus complexe (besoin de plus d'infrastructure) car réside de la combinaison des FEC et de la technique ARQ classique.

Ce travail nous a permis de renforcer nos connaissances dans le domaine des télécommunications sans fil et particulièrement dans la partie qui traite la détection et la correction des erreurs dues à la transmission. Il nous a également permis de nous familiariser avec l'outil de programmation matlab ainsi qu'avec la rédaction des documents scientifiques.

Les difficultés rencontrées sont le manque de documents en langue française vue que l'application de certaines de ces de techniques de détection et correction d'erreurs telle que HARQ est assez récente.

Enfin, comme suite à ce travail nous proposons des simulations pour plusieurs valeurs du SNR pour ainsi évaluer le BER en fonction du SNR pour les techniques que nous avons simulée et d'autres telles que CC-HARQ et IR-HARQ.

