

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE MINISTÈRE DE  
L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITÉ ABDERRAHMANE MIRA - BEJAIA  
FACULTÉ DE TECHNOLOGIE  
DÉPARTEMENT GÉNIE ÉLECTRIQUE



**Mémoire de fin d'études**  
EN VUE D'OBTENTION DU DIPLÔME DE MASTER  
OPTIONS RÉSEAUX ET SYSTÈMES DE TÉLÉCOMMUNICATIONS

---

# Etude des codes LDPC et leurs application dans la 5G NR

---

*Réalisé par :*

Mlle.NESSARK Amira  
Mlle.MOUHOUBI Lynda

*Encadré par :*

Mr.AZNI Mohamed

Membre de jury :

Mme. GAGAOUA Meriem

M.MOKRANI Karim

Promotion 2021 - 2022

# Remerciements

Tout d'abord, louange à Dieu le tout puissant, qui nous a donné la santé, le courage, la patience et bien sur la volonté pour accomplir ce modeste travail.

Nous tenons à exprimer toutes nos gratitudees à **Mr. Azni**, qui nous a accordé une grande confiance et nous a orienté dans le bon sens quant à l'élaboration de ce projet.

Toute notre reconnaissance aux nombreux professeurs qui nous ont dirigés et qui ont suivi notre formation tout au long de notre cursus universitaire.

Nous tenons également à remercier **Mr. MOKRANI** et **Mme. Gagaoua**, qui ont bien voulu nous faire l'honneur d'accepter de juger ce travail.

On tient à remercier nos chers parents pour tous les sacrifices consentis à notre égard et à leur énorme soutien.

# Dédicace

Je dédie ce projet à :

Mon père qui est ma source d'inspiration illimitée. Celui qui a toujours été là pour moi, tu n'as vraiment ménagé aucun effort pour me permettre d'atteindre mes objectifs. Merci pour ton écoute, ton sacrifice, ta patience, ta persévérance.

Ma chère et tendre mère, qui est la présence dans l'absence. Toi qui ne peux assister, ni partager cette réussite, à cette étape de ma vie. Sache que je te porte au plus profond de mon cœur. Que la lumière divine soit avec toi. Repose en Paix.

A mes frères et à ma chère famille Symbole de ma sûreté et ma sécurité.

Amira

# Dédicace

Je dédie ce Project de fin détude accompagné dun profond amour à mes très chers parents qui mont doté dune éducation digne, sans eux je ne serai jamais arrivée jusque-là.

A mes frères Amir et Sid Ahmed, et à mon adorable sur Lyna et a tous les membres de ma grande famille pour leurs encouragements permanents, conseils précieux et leurs soutien moral

A mes amies, au nom de lamitié qui nous ont réunis, et au nom de nos souvenirs inoubliables  
A tous ceux qui me sont chers

Lynda

# Table des matières

<b>Table des matières</b>	<b>6</b>
<b>Liste des figures</b>	<b>7</b>
<b>Liste des tableaux</b>	<b>8</b>
<b>Introduction</b>	<b>11</b>
<b>1 Communications dans les réseaux cellulaires</b>	<b>12</b>
1.1 Introduction . . . . .	12
1.2 Définitions et concepts de base . . . . .	12
1.2.1 Réseaux sans fil . . . . .	12
1.2.2 Réseaux cellulaires . . . . .	12
1.3 Evolution des réseaux mobiles . . . . .	13
1.4 Généralités sur les réseaux 5G . . . . .	14
1.4.1 Objectif de la 5G . . . . .	14
1.4.2 Spécifications techniques de la 5G . . . . .	14
1.4.3 Bandes des fréquences de fonctionnement . . . . .	15
1.5 La couches physique de la 5G NR . . . . .	16
1.5.1 L'OFDM (Orthogonal Frequency Division Multiplexing) . . . . .	16
1.5.2 CP-OFDM (Préfixe Cyclique- OFDM) . . . . .	16
1.5.3 DFT-S-OFDM . . . . .	17
1.6 Les techniques clé de la 5G NR . . . . .	17
1.6.1 Bande millimétrique . . . . .	17
1.6.2 MIMO massive . . . . .	18
1.6.3 Techniques de codage . . . . .	19
1.7 Conclusion . . . . .	20
<b>2 Codage de canal et codes correcteurs d'erreurs</b>	<b>21</b>
2.1 Introduction . . . . .	21
2.2 Architecture d'une chaîne de transmission numérique . . . . .	21

2.3	Codage de canal . . . . .	23
2.4	Codes correcteurs d'erreurs . . . . .	23
2.4.1	Codes convolutifs . . . . .	23
2.4.2	Codes linéaires en bloc . . . . .	24
2.4.3	La matrice génératrice d'un code linéaire . . . . .	25
2.4.4	Matrice de contrôle de parité . . . . .	27
2.4.5	Syndrome . . . . .	27
2.4.6	Distance de Hamming . . . . .	27
2.5	Familles des codes linéaires en bloc . . . . .	27
2.5.1	Les codes cycliques . . . . .	27
2.5.2	Codes de Hamming . . . . .	27
2.5.3	Les codes BCH (Bose-Chaudhuri-Hocquenghem) . . . . .	28
2.5.4	Les codes de Reed-Solomon (RS) . . . . .	28
2.5.5	Les codes LDPC . . . . .	28
2.6	Métriques de mesure et évaluation des performances des codes . . . . .	28
2.6.1	SNR et $E_b/N_0$ . . . . .	28
2.6.2	BER . . . . .	29
2.6.3	Taux d'erreur par bloc . . . . .	29
2.7	Conclusion . . . . .	29
<b>3</b>	<b>Les codes LDPC</b>	<b>30</b>
3.1	Introduction . . . . .	30
3.2	Bref historique . . . . .	30
3.3	Description des codes LDPC . . . . .	31
3.3.1	Codes à matrices creuses . . . . .	31
3.3.2	Représentation des codes LDPC . . . . .	31
3.3.3	Représentation matricielle . . . . .	31
3.3.4	Graphe de Tanner . . . . .	32
3.3.5	Codes LDPC réguliers et irréguliers . . . . .	33
3.4	Construction des codes LDPC . . . . .	33
3.5	Les codes LDPC 5G NR . . . . .	34
3.6	Les codes LDPC Quasi-cycliques . . . . .	34
3.6.1	Préliminaires . . . . .	34
3.6.2	Introduction aux codes quasi-cycliques . . . . .	34
3.6.3	Structure des codes QC-LDPC 5G NR . . . . .	35
3.7	Graphes de base et matrice de contrôle de parité . . . . .	36
3.7.1	Analyse de graphes de base . . . . .	36
3.7.2	Calcul de la matrice de contrôle de parité . . . . .	37

3.8	Encodage des codes LDPC . . . . .	37
3.9	Décodage des codes LDPC . . . . .	39
3.10	Propagation de croyance . . . . .	40
3.11	Algorithmes de décodage LDPC . . . . .	42
3.11.1	Algorithme décodage Bit-Flipping . . . . .	42
3.11.2	Algorithme Sum-Product . . . . .	44
3.11.3	Algorithme Min-Sum . . . . .	45
3.12	Conclusion . . . . .	46
<b>4</b>	<b>Simulation et résultats</b>	<b>47</b>
4.1	Introduction . . . . .	47
4.2	Présentation du logiciel . . . . .	47
4.3	Procédure de travail . . . . .	48
4.3.1	Présentation du système : . . . . .	48
4.3.2	Déroulement de la simulation . . . . .	49
4.4	Implémentation et résultats . . . . .	49
4.4.1	Comparaison entre les deux algorithmes utilisés . . . . .	51
4.4.2	Influence du nombre d'itérations sur le décodage . . . . .	52
4.5	Conclusion . . . . .	54
	<b>Conclusion générale</b>	<b>55</b>

# Table des figures

1.1	Evolution des réseaux mobile . . . . .	14
1.2	Exigences de la 5G . . . . .	15
1.3	Les différentes fréquences de la 5G NR . . . . .	16
1.4	Shéma en bloc de la modulation CP-OFDM . . . . .	17
1.5	Spectre électromagnétique . . . . .	18
1.6	MIMO Massive . . . . .	19
2.1	Chaîne de communication numérique . . . . .	22
2.2	Familles des codes correcteurs d'erreurs . . . . .	23
2.3	Shéma d'un code convolutif . . . . .	24
2.4	Codage en bloc . . . . .	25
3.1	Graphe de Tanner . . . . .	32
3.2	Un exemple de code QC-LDPC . . . . .	35
3.3	La matrice de contrôle de parité H . . . . .	38
3.4	Les matrices utilisées pour l'encodage LDPC . . . . .	38
3.5	Types d'algorithmes de décodage . . . . .	40
3.6	Décodeur LDPC . . . . .	41
3.7	Itérations entre les noeuds de variables et les noeuds de contrôle . . . . .	41
3.8	Logigramme de décodage Bit-Flipping . . . . .	43
4.1	Logo Matlab . . . . .	48
4.2	Organigramme de la simulation . . . . .	48
4.3	BER vs SNR pour BF et pour MSA avec $R=1/2$ . . . . .	50
4.4	BER vs SNR pour BF et pour MSA avec $R=1/4$ . . . . .	50
4.5	BER vs SNR pour BF et pour MSA avec $R=1/6$ . . . . .	51
4.6	Résultat de comparaison entre BF et MSA . . . . .	51
4.7	BER vs SNR pour 5 itérations . . . . .	52
4.8	BER vs SNR pour 8 itérations . . . . .	53
4.9	BER vs SNR pour 20 itérations . . . . .	53



# Liste des tableaux

- 3.1 Table de levage . . . . . 36
- 4.1 Table de paramètres utilisés . . . . . 49

# Liste des abréviations

<b>1G</b>	1ème génération
<b>2G</b>	2ème génération
<b>3G</b>	3ème génération
<b>3GPP</b>	3rd Generation Partnership Project
<b>4G</b>	4ème génération
<b>5G</b>	5ème génération
<b>ADSL</b>	Asymmetric Digital Subscriber Line
<b>AWGN</b>	Additive White Gaussian noise
<b>BCH</b>	Bose-Chaudhuri-Hocquenghem
<b>BER</b>	Bit Error Rate
<b>BF</b>	Bit-Flipping
<b>BG</b>	Base Graph
<b>BP</b>	Beleif propagation
<b>BPSK</b>	Binary Phase Shift Keying
<b>CRC</b>	Cyclic redundancy check
<b>dB</b>	decibel
<b>DVB</b>	Digital Video Broadcast
<b>E<sub>b</sub>/N<sub>0</sub></b>	Energy per bit to noise power spectral density ratio
<b>CDMA</b>	Code Division Multiple Access
<b>CP</b>	Cyclic Prefix
<b>CP-OFDM</b>	Cyclic Prefix Orthogonal Frequency Division Multiplexing
<b>DFT</b>	Discret Fourier Transform
<b>DFT-S-OFDM</b>	Discret Fourier Transform Spread Orthogonal Frequency Division Multiplexing
<b>eMBB</b>	Enhanced Mobile Broadband
<b>FDM</b>	Frequency Division Multiplexing
<b>LDPC</b>	Low Density Parity Check
<b>LLR</b>	Log-Likelihood Ratios
<b>LTE</b>	Long Term Evolution
<b>mMTC</b>	Massive Machine Type Communications
<b>MIMO</b>	Multiple Input Multiple Output
<b>MSA</b>	Min-Sum Algorithm
<b>NR</b>	New Radio

# Introduction

Lors des deux dernières décennies, les systèmes de communications numériques sont omniprésents dans notre vie quotidienne et sont devenus une composante majeure pour le déploiement de nouvelles technologies permettant de rendre les réseaux de télécommunication plus performants et plus fiables. Quel que soit le moyen utilisé pour communiquer, la transmission du message d'un émetteur au récepteur est transmise à travers un canal de communication. Quel que soit le type de canal utilisé, des erreurs viendront altérer le message transmis. De ce fait, l'utilisation des codes détecteurs ou correcteurs d'erreurs est nécessaire et le code utilisé doit être adapté au type de canal. Dans le deuxième théorème de Shannon (codage canal), Shannon a affirmé que pour tout rendement inférieur à la capacité du canal, il est possible de transmettre l'information avec une probabilité d'erreur arbitrairement faible en ajoutant de la redondance au message initial afin de construire des codes capables d'atteindre ou de s'approcher de la limite de Shannon.

Pour répondre à la problématique soulevée par Shannon, une large variété de codes correcteurs d'erreurs furent proposés tels que les codes convolutifs, les turbo codes et encore les codes LDPC (Low Density Parity Check) qui vont faire l'objet de notre étude.

Ce mémoire se concentre sur les codes LDPC, qui sont les premiers codes de contrôles de parité à faible densité découvert par Gallager, au début des années 1960. Ces dernières années avec le déploiement des nouvelles générations (4G, 5G), ces codes ont suscité une grande attention grâce à leurs remarquables capacités de correction d'erreurs proches à la limite de Shannon qui est difficile à atteindre. Pour cette raison, les codes LDPC ont un rôle essentiel pour les communications dans les réseaux de cinquième génération et ont été choisis comme schéma de codage pour le canal de données eMBB (enhanced Mobile Broad Band) 5G. Il existe de nombreux types d'algorithmes de décodage LDPC qui présentent de bonnes performances et qui peuvent être classés en algorithmes de décodage à décision douce, comme l'algorithme Sum-Product (SPA) et l'algorithme Min-Sum (MSA), et en algorithmes de décodage à décision dure, comme l'algorithme Bit-Flipping (BF).

Dans ce mémoire nous étudions les codes LDPC. Nous nous intéressons à leurs structures, à leur principe de codage, aux algorithmes de décodage et à leurs performances dans la technologie 5G. Le mémoire est constitué de quatre chapitres où nous avons opté pour une représentation progressive des concepts dans l'ordre de leur apparition dans le mémoire.

Le premier chapitre présente un petit rappel sur les réseaux mobiles (1G, 2G, 3G, 4G) et une étude détaillée sur le réseau de la cinquième génération où nous discuterons son déploiement ainsi que son objectif et ces spécifications. A la fin du chapitre nous allons parler sur les techniques de codage dans ce nouveau réseau.

Le second chapitre sera consacré à la présentation des notions de bases des codes correcteurs d'erreurs nécessaires à la compréhension de notre mémoire. Il contient comme première partie la description d'une chaîne de communication numérique et ses différents éléments en définissant particulièrement le codage de canal sur lequel se base notre étude. Dans la deuxième partie nous parlerons d'une manière générale sur les codes correcteurs d'erreurs en présentant leurs familles (convolutifs et linéaires en bloc).

Dans le troisième chapitre nous allons décrire précisément les codes LDPC, leurs encodages et leurs décodages. Les algorithmes de décodage les plus pertinents sont analysés et décrits. Nous parlerons aussi de leurs utilisations dans la 5G.

Dans le dernier chapitre, nous allons faire une étude pratique des algorithmes étudiés dans le chapitre précédent en termes de performances BER (Bit Error Rate) en faisant

des simulations sur MATLAB. Enfin, nous terminons notre travail par une conclusion et quelques perspectives.

# Chapitre 1

## Communications dans les réseaux cellulaires

### 1.1 Introduction

Avec l'évolution et la progression technologiques qui ont connu une grande propagation dans le monde, les réseaux mobiles sont devenus sans aucun doute obligatoires dans notre vie quotidienne. L'usage des services de communications mobiles a connu un succès remarquable ces dernières années. Plusieurs générations ont vu le jour (1G, 2G, 3G, 4G) et actuellement le déploiement de la 5ème génération offre un niveau de performance bien supérieur à celui des anciennes générations. En effet, l'apparition de l'Internet des objets (IoT Internet of Things) est due au déploiement de ce nouveau réseau 5G. Tout au long de ce chapitre introductif, nous allons essayer d'expliquer quelques généralités sur la 5ème génération.

### 1.2 Définitions et concepts de base

#### 1.2.1 Réseaux sans fil

Un réseau sans fil (Wireless Network en anglais) est un réseau informatique ou numérisé dans lequel au moins deux terminaux (ordinateurs, routeurs, imprimantes; etc.) peuvent communiquer sans liaison filaire sur une distance d'une dizaine de mètres à quelques kilomètres. Les réseaux sans fil ont recours à des ondes radioélectriques au lieu des câbles habituels. Grâce à ce type de réseaux, un utilisateur a la possibilité de rester connecté tout en se déplaçant dans un périmètre géographique plus ou moins étendu, c'est la raison pour laquelle on entend parfois parler de mobilité.

Il peut être associé à un réseau de télécommunication pour réaliser des interconnexions entre les noeuds [1].

#### 1.2.2 Réseaux cellulaires

Un réseau cellulaire est un système de télécommunication qui doit répondre aux contraintes de la mobilité de l'abonné dans le réseau, par l'étendue du réseau et par les ondes radio qui lui sont allouées.

Un réseau cellulaire couvre l'ensemble d'infrastructures spécialement destinées aux équipements d'acheminement de communication vers les mobiles où les ondes radio, dans

le cas d'un réseau cellulaire, servent de lien entre le terminal de l'abonné et l'infrastructure de l'opérateur [2].

## 1.3 Evolution des réseaux mobiles

L'usage des systèmes de communication sans fil a connu un succès remarquable ces dernières années et ils tendent à remplacer l'utilisation excessive des câbles. Bien que les connexions à haut-débit de type ADSL se multiplient dans le monde, elles ne permettent pas la souplesse d'utilisation que procure un réseau radio sans fil.

En effet, durant les deux dernières décennies, la qualité des signaux a connu des grands changements en passant de la 1G à la 5G. Pour le comprendre, voyons d'abord l'histoire des réseaux mobiles.

La première génération des téléphones mobiles a été intégrée au réseau de télécommunications dans les années 80, en offrant un service médiocre et très coûteux de communication mobile. Ce réseau qui repose sur une technologie dite (analogique) n'a pas connu le succès et a commencé à céder la place à la 2G dans de nombreux pays vers la fin des années 80 à cause de certains problèmes de communication. Il est désormais obsolète.

La deuxième génération (notée 2G) a marqué une rupture avec la 1G grâce au passage de l'analogique au numérique. Ce système avait utilisé des techniques de communication numériques avec multiplexage temporel (TDM), multiplexage en fréquence (FDM) ou le Code Division Multiple Access (CDMA).

La troisième génération de téléphonie mobile englobe notamment les technologies Universal Mobile Telecommunications System (UMTS), connu aussi sous le nom W-CDMA, a été impulsée pour fournir des services vocaux de radio messagerie et pour permettre aux périphériques mobiles de se connecter sur internet.

La 3G a été remplacée progressivement par la 4G qui est actuellement le réseau standard de communication mobile le plus récent, il a été introduit par le 3GPP dans les Releases 8 et 9. Il propose des débits élevés pour le trafic temps-réel, avec une large portée. Théoriquement, le LTE peut atteindre un débit de 50Mbit/s en lien montant et 100Mb/s en lien descendant mais réellement il n'atteint que 40Mbit/s [3]. Les opérateurs de téléphonie mobile sont actuellement en train de mettre en place le réseau 5G.

La 5G, ou 5G NR (New Radio), est la cinquième génération de réseau mobile, qui succède à la 4G LTE et présente une génération de rupture qui permettra un saut de performance. Bien que présente dans certains pays avant cela, c'est en 2020 que la 5G s'est vraiment propagée sur l'ensemble du globe, même si l'adoption va prendre du temps.

Parmi les principales promesses de la 5G, on trouve un débit multiplié par 10, mais aussi une latence fortement réduite, divisée par 10, et elle offre aux usagers l'ultra haut débit mobile, avec des débits dépassant les 10 Gbit/s. Grâce à une telle performance, la 5G sera utilisée dans différents domaines comme l'automobile, la domotique, les objets connectés ainsi que pour le visionnage de vidéo 4K [4]. La figure 1.1 résume schématiquement l'évolution des réseaux cellulaires.

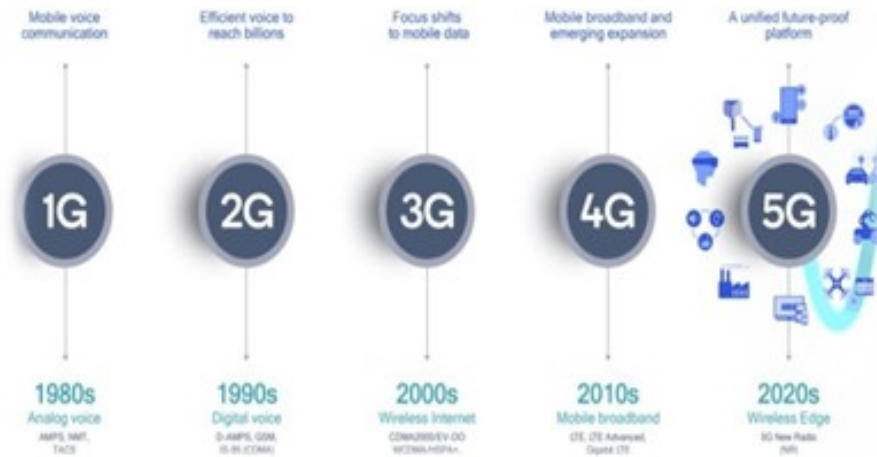


FIGURE 1.1 – Evolution des réseaux mobile

## 1.4 Généralités sur les réseaux 5G

### 1.4.1 Objectif de la 5G

Parmi les cinq principaux objectifs de la 5ème génération, on trouve l'augmentation de la vitesse et de la capacité des réseaux, mais aussi la préparation de la grande émergence d'IoT, qui nous permettra la construction d'un monde entièrement connecté. Les principales caractéristiques sont les suivantes :

- **Les forfaits de data plus volumineux :** l'échange de données sur ce réseau devra se faire avec un volume supérieur à ce que ne peut supporter la norme 4G (LTE).
- **Connectivité des masses :** sachant que la technologie 5G sert à connecter plus d'appareils, les terminaux qui peuvent être connectés par antenne devront être 100 fois plus élevés. Le nombre d'antennes dépend principalement des fréquences utilisées, en fonction de différents phénomènes, par exemple l'absorption des ondes dans les murs, tours, varies...etc [5].
- **Un débit de 1Gbps :** actuellement, les meilleurs débits pouvant être offerts par la 4G mobile sont dix fois inférieurs, il peut être équivalent ou supérieur à celui constatés actuellement dans certaines offres Internet résidentielles basées sur la fibre optique [5].
- **Faible latence :** 5G offre aussi un temps de latence extrêmement faible, c'est à dire le délai entre l'envoi et la réception des informations est très court.
- **Plus de batterie, moins de consommation :** la durée de vie des terminaux doit être considérablement améliorée, pour connecter des objets à Internet. Pour arriver à ça les constructeurs ont pensé à une amélioration des batteries mais aussi à la réduction de consommation en doublant la durée de vie des terminaux dix fois pour une même charge. Les caractéristiques de la 5G sont montrées par la figure 1.2.

### 1.4.2 Spécifications techniques de la 5G

Les deux acteurs principaux qui spécifient une nouvelle génération de téléphonie mobile sont : l'UIT (Union Internationale des Télécommunications) et le 3GPP (3rd Generation Partnership Project).

L'UIT est une agence des Nations Unies qui est particularisée dans les technologies de l'information et de la communication, elle fait des études via son (*WorkingParty5D*),

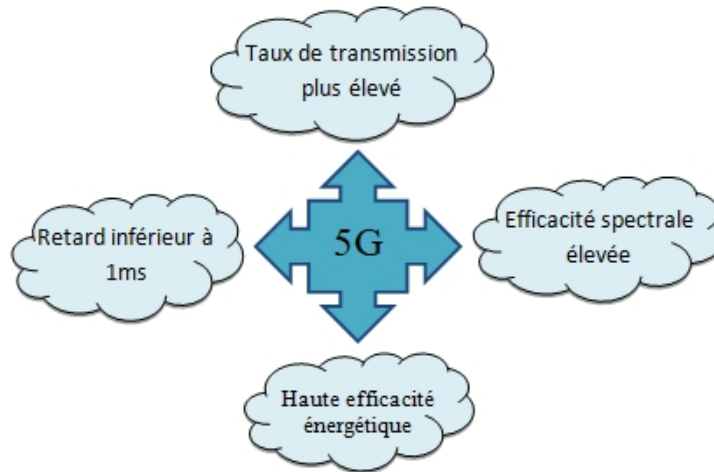


FIGURE 1.2 – Exigences de la 5G

son sous-groupe responsable du traitement des problèmes techniques et opérationnels liés aux radiocommunications.

A partir de 2013, ce groupe a commencé les travaux pour définir les caractéristiques de la nouvelle norme IMT-2020, comme il l'avait fait au début des années 2000 pour définir la 4G (IMT-Advanced). Parallèlement aux travaux de l'UIT, le 3GPP fait aussi des études.

Le **3GPP** a été établi en 1998 et il regroupe sept organismes de normalisation d'industriels, des associations et des organismes publics [6]. Il est chargé du développement et de la maintenance des spécifications techniques relatives aux normes de téléphonie mobile et il travaille aussi sur les solutions techniques permettant la réponse aux objectifs définis par l'UIT lors de l'apparition d'un nouveau standard.

La première norme 3GPP relative à la 5G (la release 15) a été lancée en décembre 2016, autant que la définition de la nouvelle architecture a commencée à partir de mars 2017, pour l'étude de la nouvelle interface air (appelée New Radio NR). Une deuxième release (3GPP release 16) a été publiée en mars 2020. Le passage d'une génération à la suivante s'est fait de façon progressive, le LTE a continué d'évoluer en parallèle de la NR et ces deux standards étaient complémentaires dans un premier temps. Il a été envisagé qu'avec les premiers déploiements 5G, le LTE reste une partie essentielle du réseau et contrôle les antennes NR [6].

### 1.4.3 Bandes des fréquences de fonctionnement

Les bandes de fréquences pour la nouvelle radio (5G NR) sont séparées en deux plages de fréquences différentes : la première est la plage qui comprend des bandes de fréquences sous 6 GHz appelée FR1 (Frequency Range 1), dont certaines sont des bandes utilisées par les normes précédentes, mais qui a été étendue pour couvrir de nouvelles offres de spectre potentielles de 410 MHz à 7125 MHz. Pour fonctionner sur cette bande, trois espacements de sous-porteuse (15, 30 et 60KHz), sont identifiés. L'autre est la gamme de fréquences 2 (FR2), dans laquelle les bandes de fréquences vont de 24,25 GHz à 52,6 GHz. En FR2 il existe, deux espacements de sous-porteuse (60 et 120KHz) pour fonctionner en bandes millimétriques.



La bande FR2+ est une nouvelle bande de fréquence, composée d'une bande allant de 52 jusqu'à 52.6 GHz et une autre bande de 57 à 71 GHz. La figure 1.3 montre la répartition des fréquences dans la nouvelle norme 5G [7].

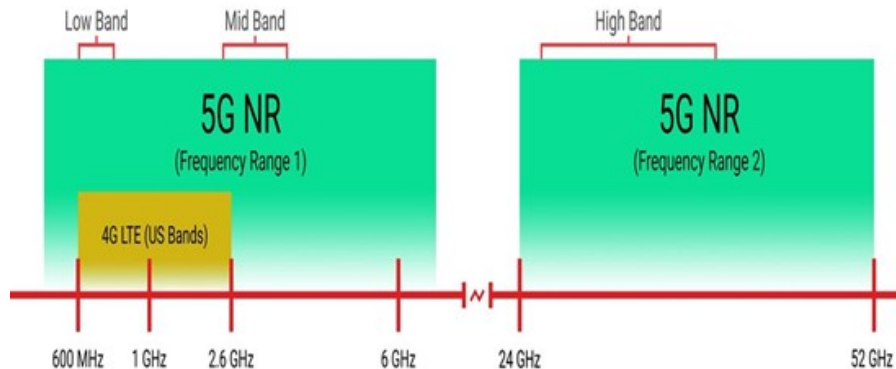


FIGURE 1.3 – Les différentes fréquences de la 5G NR

## 1.5 La couches physique de la 5G NR

Dans la norme 4G/LTE, la technique d'accès qui a été adoptée est OFDMA (Orthogonal Frequency Division Multiple Access). En revanche en 5G les techniques orthogonales OFDMA et non-orthogonales (NOMA) sont conjointement utilisées. Dans cette section, nous allons aborder OFDMA en omettant les spécificités de NOMA.

### 1.5.1 L'OFDM (Orthogonal Frequency Division Multiplexing)

La technique de modulation OFDM, est une évolution des techniques de multiplexage par répartition en fréquence (FDM). Elle se base sur l'utilisation des formes d'impulsions orthogonales pour multiplier les données dans le domaine fréquentiel. L'utilisation de l'OFDM dans les applications adressées vers les consommateurs a débuté à la fin des années 80 et après le déploiement de la ligne numérique d'abonné asymétrique (ADSL). La modulation OFDM a été utilisée dans les applications sans fil avec le Wifi (Wireless Fidélité) et LTE [8].

### 1.5.2 CP-OFDM (Préfixe Cyclique- OFDM)

La version spécifique d'OFDM utilisée pour la liaison descendante 5G NR est le préfixe-cyclique OFDM (voir la figure 1.4). C'est la même version utilisée par LTE dans sa liaison descendante.

La caractéristique principale de CP-OFDM, est l'ajout de la dernière partie des données de la trame OFDM au début de la trame suivante et la longueur du préfixe cyclique est supérieure à l'étalement de retard de canal afin de supporter les interférences. Donc, on note que la longueur du CP est adaptative en fonction des conditions de la liaison de transmission.

Les formes d'onde utilisées par la liaison montante 5G NR diffèrent de la 4G LTE, car elles sont basées sur CP-OFDM et DFT-S-OFDM. De plus, la 5G NR vise à utiliser

un espacement flexible entre les sous porteuses de 30KHz, avec un espacement maximal de 240KHz, par contre les sous-porteuses LTE avaient un espacement fixe de 15KHz [9].

Afin de prendre en charge correctement les types de bandes de fréquence et de modèles de déploiement de la 5G, un espacement flexible des porteuses est utilisé. A titre d'exemple, la 5G NR doit être capable de fonctionner dans les bandes d'ondes millimétriques qui ont des canaux plus larges allant jusqu'à 400MHz [9].

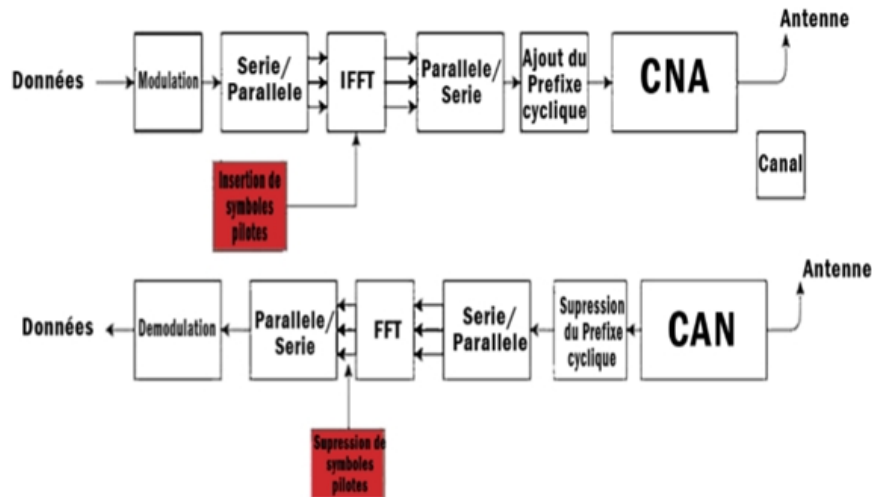


FIGURE 1.4 – Schéma en bloc de la modulation CP-OFDM

### 1.5.3 DFT-S-OFDM

L'OFDM à diffusion directe par transformée de Fourier (DFT-S-OFDM), est un schéma de transmission de type SC (Single Carrier) c-à-d à porteuse unique. Il peut être combiné avec OFDM qui offre une grande flexibilité pour un système de communication mobile. La technique de multiplexage à division fréquentielle à simple porteuse (SC-FDMA), est très similaire à celui d'OFDM à la différence qu'une DFT est appliquée avant la modulation OFDM.

La DFT-S-OFDM est utilisée pour la transmission en liaison montante, en particulier dans les scénarios de couverture limitée. La séquence de bits envoyée par l'émetteur est mappée simultanément sur toutes les sous-porteuses, mais avec des intervalles de transmission plus courts (modulation d'amplitude BPSK, QPSK ou M-Quadrature). Des intervalles de garde (préfixe cyclique) sont ajoutés entre les blocs de symboles afin d'éviter les interférences de propagation temporelle entre les symboles adjacents causés par la propagation des trajets multiples [10].

## 1.6 Les techniques clé de la 5G NR

Les réseaux 5G NR ont besoin de connecter des milliards d'appareils. Pour que cela fonctionne, ils auront besoin de la précision millimétrique et de l'efficacité énergétique du MIMO massive.

### 1.6.1 Bande millimétrique

Il s'agit des fréquences supérieures à 6 GHz. Pour des raisons technologiques et de qualité de propagation, ces fréquences n'ont encore jamais été prises pour le déploiement

des réseaux mobiles. Ces hautes fréquences offrent un débit beaucoup plus important. Pour répondre à cette augmentation des débits et des volumes de données à transmettre il faut offrir des bandes disposant de plus de 1001 MHz par utilisateur, et leur utilisation nécessite le développement de toutes les technologies nécessaires. La communication par ondes millimétriques peut être utilisée sur une distance allant de 200 à 300 m pour une couverture extérieure pour des réseaux à haute densité. Chacune des cellules aura une couverture réduite, ce qui nécessitera l'implémentation de la technique de *beamforming*.

La figure montre 1.5 la correspondance entre les longueurs d'ondes et les bandes des fréquences standardisées [11].

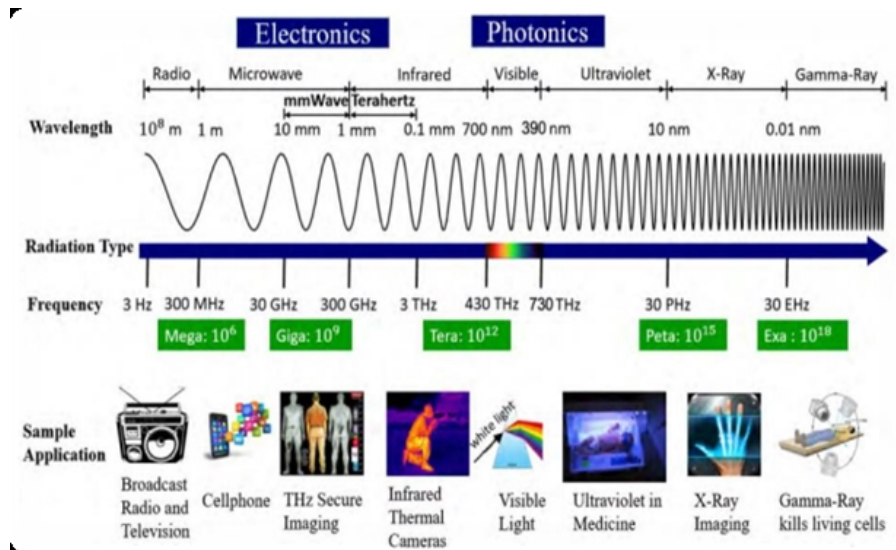


FIGURE 1.5 – Spectre électromagnétique

## 1.6.2 MIMO massive

Cette technique consiste à augmenter le nombre d'antennes pour l'émission et la réception de la liaison sans fil et cela dans le but d'augmenter le débit pour les communications mobiles de dernière génération. Cette technique est utilisée aussi pour réduire les évanouissements de signal provoqués par l'environnement de propagation des trajets multiples et le bruit de canal.

Pour que cela fonctionne il faut que chaque réseau utilise plusieurs antennes, grâce au traitement de signaux et à toutes ces antennes, les tours relais peuvent utiliser des modes de propagation différents pour envoyer plus de données avec une meilleure fiabilité [11].

Le MIMO massive a le même concept avec le système MIMO. La différence est que le MIMO massive utilise un nombre excessif d'antenne, de telle façon que le nombre de transmetteurs dépasse le nombre d'utilisateurs, cela lui offre plus d'avantage que le MIMO actuel.

Cet aspect est schématisé par la figure 1.6

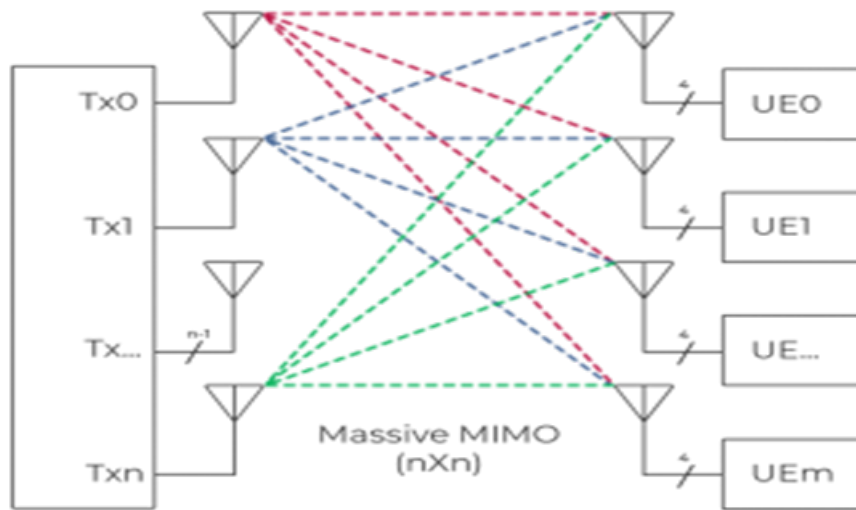


FIGURE 1.6 – MIMO Massive

### 1.6.3 Techniques de codage

Dans la norme 5G, les codes polaires et les codes LDPC (Low Density Parity Check) ont été adoptés. Les codes polaires sont utilisés dans le plan de contrôle alors que les codes LDPC sont utilisés dans le plan utilisateur.

#### A- Codage LDPC

Les codes LDPC ont été adoptés en 5G en remplacement des turbo-codes. Ces codes sont des codes linéaires en blocs basés sur une matrice de contrôle de parité creuse. En raison de leur capacité de calcul limitée, ils ont été oubliés pendant des dizaines d'années.

Ces dernières années, les codes LDPC ont attiré d'avantage d'attention en raison de leurs excellentes capacités de correction d'erreur, de leurs algorithmes de décodage efficace et de leurs performances proches de la limite de Shannon pour les grandes longueurs de code [12].

Les codes LDPC peuvent répondre aux exigences de la 5G. Les graphes de base définis dans la norme 3GPP TS 38.21 [13] peuvent prendre en charge efficacement la HARQ (Hybrid Automatic Request) et la compatibilité de taux qui peut prendre en charge une quantité arbitraire de bits d'information transmis avec des taux de code variables.

#### B- Codage polaire

Les codes polaires ont été proposés pour la première fois par Erdal Arikan en 2009 [14]. Il représente une classe de codes correcteurs d'erreurs en bloc linéaires. Ces codes ont une architecture de codage fondée sur le circuit de transformée de Fourier rapide discret, consistant à fixer un certain nombre de bits d'entrée à 0, un algorithme d'encodage permettant de calculer la probabilité d'un bit d'entrée connaissant les bits d'entrée précédents et les probabilités que les bits de sortie valent. Son algorithme de décodage par annulations successives est suffisant pour atteindre la capacité du canal pour les canaux sans mémoires symétriques [7].

Les performances des codes polaires lui ont permis d'être adopté pour les canaux de contrôle de la 5G du standard de communication mobiles 5G [15].

## 1.7 Conclusion

Dans ce chapitre nous avons défini les réseaux cellulaires et les réseaux sans fil. Nous avons aussi suivi l'évolution des réseaux cellulaires à travers les différentes générations, en commençant de la première génération analogique jusqu'à la 5ème génération que nous avons présentée d'une manière assez détaillée. Nous avons présenté quelque généralité sur la 5ème génération en introduisant son objectif, ces spécifications techniques et les bandes de fréquence. Aussi nous avons revu la couche physique de la 5G NR, en parlant de la technique massive MIMO et les différents types de codage.

Dans le chapitre suivant nous allons considérer d'une manière globale les codes correcteurs d'erreurs. Nous considérerons les deux familles de codes que sont les codes linéaires en bloc et les codes convolutifs.

# Chapitre 2

## Codage de canal et codes correcteurs d'erreurs

### 2.1 Introduction

Les systèmes de communication numériques transmettent de l'information depuis une source vers un ou plusieurs destinataires, à travers un canal de transmission qui est loin d'être parfait. Alors le problème majeur que le codage de canal s'efforce de combattre est les erreurs de transmission dues aux perturbations intervenant sur le canal de transmission. Pour cela, l'information devra être codée d'une manière spéciale permettant de détecter les erreurs, ou ce qui est encore mieux de les corriger automatiquement.

Dans ce chapitre nous allons présenter quelques généralités sur les codes correcteurs d'erreurs. Nous allons considérer les deux classes de codes : les codes convolutifs et les codes linéaires en bloc. Dans la classe des codes en bloc nous allons nous intéresser en particulier aux codes LDPC qui font l'objet de notre étude dans ce mémoire.

### 2.2 Architecture d'une chaîne de transmission numérique

Les systèmes de communications numériques véhiculent de l'information entre une source et un ou plusieurs destinataires à travers un canal de transmission (le câble, la fibre optique ou encore, la propagation sur un canal hertzien). Les signaux transportés peuvent être soit directement d'origine numérique comme dans les réseaux de données, soit d'origine analogique (parole, image...) mais convertis sous une forme numérique [16].

Le schéma synoptique d'un système de transmission numérique est illustré par la figure 2.1.

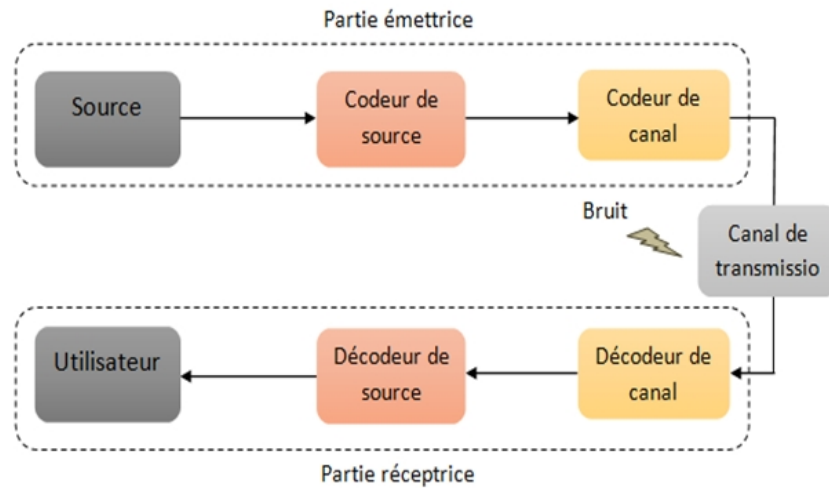


FIGURE 2.1 – Chaîne de communication numérique

Dans la partie émettrice, on distingue les blocs suivants :

**La source du message :** C'est le premier maillon de la chaîne de transmission qui fournit le message porteur de l'information sous forme de symboles, qui peut être soit numérique ou analogique.

**Le codeur de source :** Il effectue une mise en correspondance entre l'alphabet du message et celui du canal. L'objectif principal du codage source est de représenter les données de façon compacte (compression de données en éliminant les éléments binaires non significatifs) et de réduire la redondance de la source pour diminuer le débit binaire à transmettre. Les symboles qui sont groupés en blocs à la sortie de ce codeur, sont appelés des mots d'information.

**Le codeur canal :** Son objectif est de protéger les mots d'information en ajoutant de la redondance à chacun d'eux. Le message rallongé est alors appelé mot de code. Contrairement au codage de source, le codage de canal augmente le débit. Optionnellement, les mots de code obtenus peuvent être adaptés aux caractéristiques du canal en leur appliquant un codage de ligne ou un codage de ligne suivi d'une modulation.

**Le canal de transmission :** un canal de communications est le milieu physique qui sert à lier l'émetteur et le récepteur et à transmettre la suite des signaux analogiques, il se caractérise par sa capacité et par sa bande passante. Un canal de transmission peut être matérialisé par une ligne téléphonique, une liaison radio, un support magnétique ou optique ; sa nature diffère selon le type de grandeur qu'il permet de véhiculer.

Le milieu sera généralement perturbé par un bruit qui dépendra de l'environnement et de la nature du milieu. Par exemple, des perturbations électriques ou électromagnétiques [9].

Du côté récepteur, les fonctions de démodulation et de décodage sont les inverses respectifs des fonctions de modulation et de codage situées dans la partie émettrice. Globalement, le récepteur effectue les opérations de détection en transformant les signaux reçus en une suite de symboles. Il commet parfois des erreurs d'estimations puisque le signal reçu peut être entaché de bruit. Ainsi, à la sortie du canal, nous obtenons ce que l'on appelle les mots reçus. Notons que les mots reçus ne ressemblent pas toujours à ceux du mot de code [9]. Le récepteur connaît la loi de codage utilisée et donc il est capable de détecter et corriger les données binaires erronées [17].

Le décodage de canal consiste à détecter la présence d'erreurs et de les corriger en utilisant la redondance contenue dans les mots de code, si la capacité du code n'est pas dépassée, et il fournit aussi une estimation du mot d'information. Cette estimation

consiste à reconstituer l'information originale de la séquence de substitution et la délivrer à l'utilisateur en inversant l'opération du codeur de source.

## 2.3 Codage de canal

Dans cette section, nous allons considérer plus en détails le codage de canal. En effet, après le codage de source, on obtient une série de bits vulnérable aux erreurs de transmission qui peuvent être introduites par le canal. Il est donc primordial de protéger les messages transmis. Le codage de canal, ou codage correcteur d'erreurs, est un composant essentiel des systèmes de transmission de l'information, il peut minimiser l'effet du bruit du canal en utilisant un codeur, qui code les bits d'information en leurs ajoutant de la redondance contrôlée. Cette dernière est appliquée en utilisant des méthodes algébriques de sorte à pouvoir assurer la fiabilité de la transmission de données numériques et permet d'assurer une certaine diversité temporelle qui va être exploitée par le décodeur pour récupérer l'information originale de l'émetteur. Le décodeur de canal exploite les bits redondants et récupère les bits d'information en utilisant un algorithme de décodage.

Il existe de nombreux schémas de codage de canal. Nous consacrerons la prochaine section à l'étude des codes correcteurs d'erreurs.

## 2.4 Codes correcteurs d'erreurs

Les codes correcteurs d'erreurs peuvent être classés comme montrés par la figure 2.2. Les deux classes les plus importantes sont les codes convolutifs et les codes linéaires en bloc. Dans ce mémoire, nous nous intéressons à la classe des codes en blocs, et en particuliers aux codes LDPC. Ainsi, nous passerons en revue les deux classes de codes mais nous réserverons plus de détails aux codes LDPC.

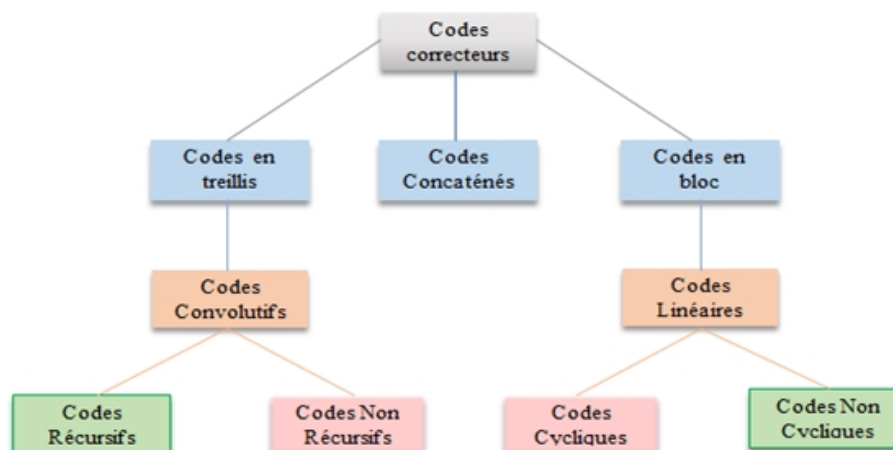


FIGURE 2.2 – Familles des codes correcteurs d'erreurs

### 2.4.1 Codes convolutifs

Un code convolutionnel est un type de code correcteur d'erreurs formant une classe extrêmement souple et efficace. Ce sont les codes les plus utilisés dans la communication fixe et mobile. Les codes convolutifs ont été introduits comme une alternative aux codes en bloc en 1955 par P. Elias [18]. Ceux-ci sont des codes à mémoire ce qui veut dire que



la sortie du codeur dépend de l'entrée présente et de l'entrées précédentes. Un codeur convolutif est défini par trois paramètres,  $n$  le nombre de sorties,  $k$  le nombre d'entrées et  $v$  la taille de la mémoire. Chaque code convolutif est caractérisé par le rendement  $R$  et par sa longueur de contrainte  $K$ , données respectivement par les équations (2.1) et (2.2).

$$R = k/n \quad (2.1)$$

$$K = v + 1 \quad (2.2)$$

Les bits d'informations sont introduits dans des registres à décalage à états finis. La combinaison linéaire du contenu des registres à états va générer les bits codés comme le montre la figure 2.3. Un registre à décalage est un outil électronique ou informatique de taille fixe dans lequel les bits sont décalés à chaque coup d'horloge  $t$  du système. Chaque registre est formé de bascule qui permettent de stocker un bit et qui a chaque temps d'horloge fait sortir l'information qu'elle contient. A chaque temps d'horloge un bit rentre et un bit modifier sort du registre. Au cours du registre à décalage, il est possible d'effectuer une série d'opérations linéaires et de transformations comme un « XOR » sur les bits ou toutes autre opérations électroniques, ainsi chaque bit de sortie dépend des  $v$  autres bits du registre, on parle de mémoire du registre ou complexité. Les codes convolutifs sont très intéressants dû à l'efficacité du très connus algorithme de Viterbi.

Parmi les codes convolutifs, nous pouvons distinguer les turbo codes qui sont les premiers codes pratiques à se rapprocher de près de la capacité maximale ou limite de Shannon. Ils représentent une percée majeure dans le domaine de la communication numérique. Ces codes, inventés et présentés par Claude-Berrou à l'ENST de Bretagne [19], sont obtenus par la concaténation parallèle/série [20] ou hybride de deux ou plusieurs codes correcteurs d'erreurs de faible complexité.

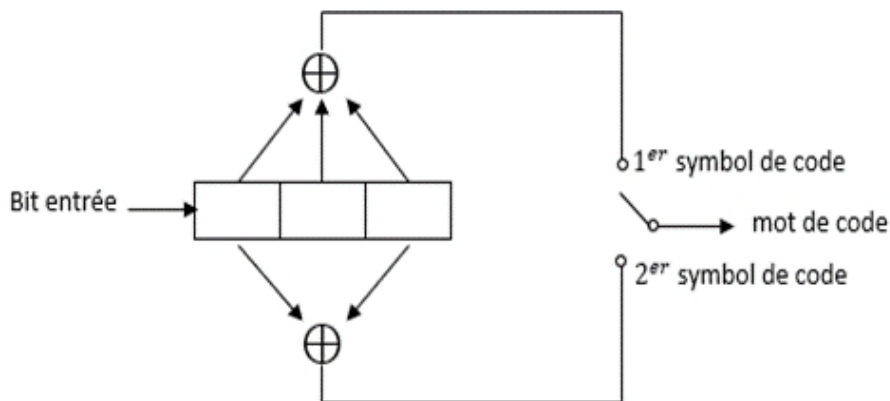


FIGURE 2.3 – Schéma d'un code convolutif

## 2.4.2 Codes linéaires en bloc

Le codage en bloc s'effectue par bloc de symboles. Considérons la série de bits à la sortie du codeur de source et désignons la par  $\{d_n\}$ . Le codage de canal par codage en blocs consiste à diviser la séquence  $\{d_n\}$  en groupe de  $k$  bits. Chaque groupe ainsi constitué est appelé un mot d'information.

A chaque groupe, le codeur de canal ajoutera  $(n - k)$  bits de redondance (ou de parité) pour former des groupes de  $n$  ( $n > k$ ) bits, appelés mots de codes. Ce codeur canal travaillant sur  $k$  bits d'informations pour constituer des mots de codes de  $n$  bits est désigné par  $\mathbf{C}(n, k)$ . Avec  $k$  bits d'entrée, on peut former  $2^k$  mots d'information qui ne diffèrent l'un de l'autre que par un bit. En revanche, le codeur va utiliser  $2^k$  mots de codes choisis parmi  $2^n$  mots de codes possibles. En effet, un code linéaire en bloc est une application biunivoque entre des mots d'informations de  $k$  bits et des mots de code de  $n$  bits et l'objectif principal du codage de canal est ainsi de rajouter  $(n-k)$  bits de parité de façon à avoir des mots de codes qui diffèrent l'un de l'autre par un aussi grand nombre de bits que possible. Ainsi, le bruit aura un effet minime sur les données. La figure 2.4 montre le principe du codage linéaire en blocs.

Considérons deux mots de code  $\mathbf{C}_i$  et  $\mathbf{C}_j$  et deux constantes binaires quelconques  $\alpha_i$  et  $\alpha_j$ , on dit que le code est linéaire si seulement si  $(\alpha_i \mathbf{C}_i + \alpha_j \mathbf{C}_j)$  est aussi un mot de code. Un code linéaire contient le vecteur nul, donc les codes à poids fixe ne sont pas linéaires. Les bits de contrôle sont une combinaison linéaire des bits d'information.

Chaque code en bloc est défini par son rendement qui est donné par l'équation 2.3.

$$R = \frac{\text{nombre de symboles d'entrées}}{\text{nombre de symboles de sorties}} = k/n \quad (2.3)$$

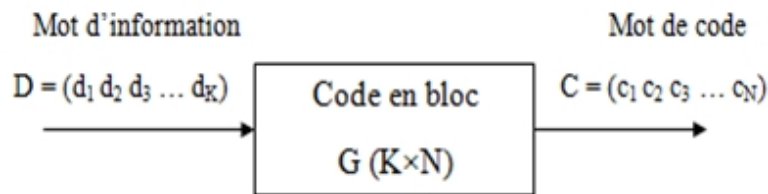


FIGURE 2.4 – Codage en bloc

Chaque code en bloc est ainsi défini par une matrice génératrice  $G$  de taille  $(K \times N)$ , où  $K$  est le nombre de lignes et  $N$  est le nombre de colonnes. En effet, puisque le code est linéaire (la correspondance entre la sortie et l'entrée est une fonction linéaire), chaque bit d'un mot de code donné est obtenu par une combinaison linéaire des bits du mot d'information. Dans ce qui suit, nous exposons ce principe avec plus de détails.

### 2.4.3 La matrice génératrice d'un code linéaire

Les composantes du mot de code  $\mathbf{C}$  sont obtenues par combinaison linéaire des composantes du mot de données original  $d$ . Si les  $K$  premiers bits d'un mot codé sont des bits de données, on dit que le code est systématique. C'est-à-dire, le mot de code est représenté sous la forme suivante :

$$\mathbf{C} = (c_{K+1} \dots c_N, d_1 \dots d_K)$$

Considérons un code systématique. Soit  $d = [d_1 d_2 \dots d_K]$  le mot de données et soit  $\mathbf{C} = [c_1 c_2 \dots c_N]$  le mot codé qui lui est associé.

Nous pouvons écrire donc :

$$\begin{aligned} c_1 &= d_1 \\ c_2 &= d_2 \\ &\vdots \\ c_K &= d_K \\ c_{K+1} &= p_{11}d_1 \oplus p_{12}d_2 \oplus \dots \oplus p_{1K}d_K \\ &\vdots \\ c_N &= p_{M1}d_1 \oplus p_{M2}d_2 \oplus \dots \oplus p_{MK}d_K \end{aligned}$$

Où  $M = N - K$

Sous forme matricielle, cette relation s'écrit :  $[\mathbf{C}] = [\mathbf{d}]\mathbf{G}$

Où :

$$\mathbf{G} = \begin{bmatrix} 1 & \dots & 0 & p_{11} & \dots & p_{1K} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 1 & p_{1K} & \dots & p_{MK} \end{bmatrix} \quad (2.4)$$

est la matrice génératrice du code.

Notons que  $\mathbf{G}$  est sous forme :

$$\mathbf{G} = [I_K \quad P^t]$$

Où  $I_K$  est la matrice identité de dimension  $(K \times K)$  et  $\mathbf{P}$  est de dimension  $(N - K) \times K$  et  $P^T$  est de dimension  $K \times (N - K)$  comme attendu.

$$\mathbf{P} = \begin{bmatrix} p_{11} & \dots & p_{1M} \\ \vdots & & \vdots \\ p_{1M} & \dots & p_{MK} \end{bmatrix}$$

### Exemple :

Un code (5,3) est défini par sa matrice génératrice  $\mathbf{G}$  qui est donnée par :

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} = [I_3 P^t]$$

A partir de la matrice génératrice on peut déterminer une matrice définissant le décodeur d'un code en bloc et elle possède encore une forme particulièrement simple. Cette matrice est appelée matrice de contrôle de parité. Elle permet de détecter les erreurs de transmission.

#### 2.4.4 Matrice de contrôle de parité

Pour chaque matrice  $\mathbf{G}$  de taille  $K \times N$ , il existe une matrice  $\mathbf{H}$  de dimension  $M \times N$  ( $M = N - K$ ), définie par

$$\mathbf{H} = [P \ I_M] = [P \ I_{N-K}]$$

On écrit donc la transposée de  $\mathbf{H}$  comme suit

$$\mathbf{H} = \begin{bmatrix} P^t \\ I_m \end{bmatrix}$$

Tel que la relation suivante est vérifiée :  $\mathbf{G} \cdot \mathbf{H}^T = 0$

#### 2.4.5 Syndrome

Considérons le cas où l'émetteur transmet le mot de code  $\mathbf{C}$  et le récepteur reçoit le mot  $\mathbf{r}$ . On appelle syndrome, le vecteur  $s = [\mathbf{r}] \mathbf{H}^T$ . Si le vecteur  $\mathbf{s}$  est nul, donc le mot  $\mathbf{r}$  est un mot de code valide, c'est-à-dire qu'il n'a pas été affecté par le bruit.

#### 2.4.6 Distance de Hamming

Une mesure qui convient pour estimer la fiabilité d'un code est la distance de Hamming. La distance de Hamming entre un mot codé  $\mathbf{C}_1$  et un mot  $\mathbf{C}_2$  est notée  $d_{\mathbf{H}}(\mathbf{C}_1, \mathbf{C}_2)$  et qui est le nombre de bits qui sont différents entre ces deux mots de code.

Pour avoir un code efficace, il faut donc maximiser la distance minimale de Hamming  $d_{min}$ . Celle-ci est la distance de Hamming entre les deux mots de codes les plus proches du code  $\mathbf{C}$ .

**Remarque :**

Si un code est de distance  $d_{min}$  on pourra détecter toutes les erreurs de poids inférieur à  $d_{min}-1$  et on pourra corriger  $t$  erreurs si  $t < d_{min}/2$ .

### 2.5 Familles des codes linéaires en bloc

#### 2.5.1 Les codes cycliques

Les codes cycliques forment une sous-classe des codes linéaires, et sont les plus utilisés en pratique [21]. Un code linéaire en bloc  $\mathbf{C}$  est dit cyclique s'il est stable par permutation circulaire de mots, c'est-à-dire pour tout mot de code  $\mathbf{C} = (c_0, c_1, \dots, c_{n-1})$  son décalé  $\mathbf{C}_1 = (c_{n-1}, c_0, \dots, c_{n-2})$  est aussi un mot de code.

#### 2.5.2 Codes de Hamming

Un code de Hamming est défini comme un code correcteur linéaire qui permet la correction et la détection des erreurs sur un seul bit et indique également le bit qui contient l'erreur à corriger. Le code de Hamming est donc un code binaire défini par sa matrice de parité plutôt que par sa matrice génératrice [22].

### 2.5.3 Les codes BCH (Bose-Chaudhuri-Hocquenghem)

Les codes BCH forment une grande classe de codes correcteurs d'erreurs cycliques aléatoires puissants. La caractéristique principale de ces codes est la possibilité de contrôle précis du nombre d'erreurs de symbole que le code peut corriger pendant la conception du code. Le décodage des codes BCH n'est pas compliqué en suivant une méthode algébrique nommé le décodage de syndrome.

Pour construire un code BCH, on se fixe une distance construite  $d$  du code, et on détermine son polynôme générateur  $g(x)$ . La distance minimale  $d_{min}$  du code obtenu est toujours supérieure ou égale à la distance construite

### 2.5.4 Les codes de Reed-Solomon (RS)

Parmi les codes cycliques les plus utilisés on retrouve les codes de Reed-Solomon, qui ont été inventés par Irving S.Reed et Gustave Solomon en 1960. Ils appartiennent à la famille des codes parfaits, ou codes MDS. Ils sont connus par leur utilisation dans les communications spatiales, notamment avec les sondes Voyager. Les codes RS sont des codes optimaux. Pour tous paramètres de code  $(n,k,d=n-k+1)$ , il existe un code RS  $(n,k,d)$  sur le corps fini  $F_Q$  pour  $n \leq q+1$ , Où  $n = q-1$ ,  $q = 2^s$  et  $s$  est le nombre de bits par symbole.

Pour une distance minimale  $d$  et une dimension  $(k = nd+1)$ , le polynôme générateur d'un code RS est sous la forme suivante

$$g(x) = \prod_{i=1}^{d-1} (x - \alpha^i)$$

Où  $\alpha$  est un élément primitif du corps fini à  $q$  éléments.

### 2.5.5 Les codes LDPC

Un code LDPC est un code dont la matrice de contrôle de parité  $H$  de taille  $(M \times N)$  est de faible densité [23]. Ainsi le nombre de 1 dans la matrice est faible devant le nombre de 0. Cette matrice  $H$  définit un code en bloc, où le nombre de bits d'information est  $K = N - M$ . Ces codes vont être détaillé dans le prochain chapitre.

## 2.6 Métriques de mesure et évaluation des performances des codes

### 2.6.1 SNR et $E_b/N_0$

Le rapport signal bruit (SNR), en anglais (signal to noise ratio), est un indicateur de la qualité de transmission d'une information. C'est le rapport entre la puissance du signal utile et celle du bruit de fond indésirable.

$$SNR = \frac{\text{puissance (signal)}}{\text{puissance (bruit)}} \quad (2.5)$$

Il est exprimé généralement en décibels sous forme logarithmique :

$$SNR(dB) = 10 \log_{10} \frac{P(\text{signal})}{P(\text{bruit})} \quad (2.6)$$

Plus la valeur de SNR ou  $E_b/N_0$  est grande, plus le signal est puissant.  $E_b/N_0$  est également exprimé en décibels (dB). Pour un code de rendement  $R$ , la relation entre le SNR et  $E_b/N_0$  est exprimé par l'équation suivante :

$$SNR = 2R \frac{E_b}{N_0} \quad (2.7)$$

Où  $E_b$  est l'énergie reçue par symbole binaire d'information transmis et  $N_0$  la densité spectrale de puissance du bruit contenue dans la bande.

### 2.6.2 BER

BER (Bit Error Rate) ou bien TEB (Taux d'Erreur Binaire), est le rapport entre le nombre de bits erronés et le nombre total de bits transmis, qui est un paramètre clé permet de quantifier un canal de transport de données en calculant le taux d'erreurs dans la chaîne de données. Cette définition peut être traduite en une formule simple :

$$BER = TEB = \frac{\text{nombre d'erreurs}}{\text{nombre total de bits envoyés}} \quad (2.8)$$

### 2.6.3 Taux d'erreur par bloc

Le taux d'erreur par bloc (BLER) nommé aussi FER (Frame Error Rate), est le rapport entre le nombre de bloc erronés et le nombre total de blocs transmis, exprimé en pourcentage. Le BLER est utilisé pour mesurer les performances au niveau de la couche physique en évaluant le contrôle de redondance cyclique (CRC) sur chaque bloc de transport.

$$FER = \frac{\text{nombre de trames erronées}}{\text{nombre total de trames transmises}} \quad (2.9)$$

## 2.7 Conclusion

Dans ce chapitre, nous avons donné un aperçu général sur la chaîne de communication numérique. Nous avons également décrit les notions de base sur les codes correcteurs d'erreur où nous regroupons ces codes selon deux classes : linéaires ou convolutifs. Chaque famille se distingue des autres codes par leurs propriétés structurelles, notamment les codes LDPC qui vont être détaillés dans le chapitre suivant.

# Chapitre 3

## Les codes LDPC

### 3.1 Introduction

Les codes de contrôle de parité à faible densité (LDPC), ont reçu beaucoup d'attention pour la correction d'erreur directe dans les systèmes de communication sans fil en raison de leurs excellentes performances et efficacité. Au cours des dernières décennies, ils ont été utilisés dans diverses applications telles que WiMAX, Digital Video Broadcast (DVB) et Advanced Television System Committee (ATSC). En particulier, ils ont été sélectionnés par 3GPP dans le scénario Enhanced Mobile BroadBand (eMBB) comme codes de correction d'erreurs pour la transmission de données dans les communications mobiles de la 5ème génération.

Ce chapitre fournit les bases pour l'étude des codes LDPC. Nous commencerons par un bref historique et une simple description. Les représentations fondamentales et la construction des codes LDPC seront introduites. Nous passerons également en revue les codes LDPC 5G NR, leur codage et décodage. Enfin, la dernière partie de ce chapitre présentera quelques algorithmes utilisés pour décoder les codes LDPC.

### 3.2 Bref historique

En 1948, Shannon a prouvé que tout canal bruité possède un débit maximal auquel l'information peut être transférée et qu'il est possible de concevoir des codes correcteurs d'erreurs qui s'approchent de cette capacité, ou limite de Shannon à condition que les codes puissent être d'une longueur illimitée. Au cours de ces dernières décennies, les théoriciens du codage ont cherché des codes pratiques capables de se rapprocher de la limite de Shannon. Parmi les familles de bons candidats, les codes LDPC [24], qui sont une classe de codes linéaires en blocs qui offrent des performances proches de la capacité sur un grand nombre de canaux de transmission tout en admettant des décodeurs implémentables.

Les codes LDPC (Low Density Parity Check) ont été proposés pour la première fois par Robert Gallager dans sa thèse de doctorat de 1960 [25], mais à cette époque, ces codes ont été largement négligés en raison de la complexité des calculs. Une exception notable est le travail important de Tanner en 1981 [26] dans lequel il a généralisé les codes LDPC et a introduit une représentation graphique de ces codes, maintenant appelés graphes de Tanner. Finalement, Après la découverte des Turbo codes, les codes LDPC ont été revisités, et leur étude a été relancée au milieu des années 1990 par MacKay, Luby et d'autres qui ont remarqué, indépendamment des travaux de Gallager, les avantages de ces codes qui possèdent des matrices de contrôle de parité éparses.

Les codes LDPC se sont développés rapidement au fil du temps et ont été adoptés dans de nombreuses applications telles que la diffusion numérique par satellite et les normes de communication optique longue distance et sont très susceptibles d'être adoptés dans la norme IEEE pour les réseaux locaux sans fil.

## 3.3 Description des codes LDPC

### 3.3.1 Codes à matrices creuses

Les codes LDPC, pour Low Density Parity Check, forment une famille populaire de codes correcteurs linéaires en bloc, caractérisés par une matrice de contrôle de parité creuse, c'est-à-dire une matrice dont le nombre d'éléments non nuls est très faible devant le nombre de 0. Grâce à cette particularité, il est possible d'utiliser une classe particulière d'algorithmes de décodage en se basant sur le principe de propagation de croyance (Belief propagation). Ces algorithmes ont l'avantage d'avoir une complexité linéaire à la longueur du code. Ainsi les codes LDPC peuvent protéger efficacement des messages de grande longueur [27] [28].

Dans ce projet, nous nous sommes intéressés aux codes binaires. Un code LDPC noté  $\mathbf{C}(n, k)$  est dit binaire, si tous les éléments non nuls dans sa matrice de parité égalent à 1, où  $n$  est la longueur des mots codes et  $k$  représente le nombre de bits d'information. La matrice de parité est définie par  $M$  contraintes (équations) de parité et  $n$  symboles du mot de code, donc le nombre d'informations  $k$  est donné par :

$$k = n - M$$

Avec un rendement :  $R = k/n$

Nous allons maintenant présenter les codes LDPC, en présentant dans un premier temps les moyens dont nous disposons pour les représenter : leur matrice de parité ou leur graphe de Tanner.

### 3.3.2 Représentation des codes LDPC

Comme tous les codes linéaires en bloc, il existe deux possibilités différentes pour représenter les codes LDPC, ils peuvent être représentés sous forme d'un graphe bipartite (dit aussi graphe de Tanner), comme ils peuvent être décrits sous une forme matricielle.

### 3.3.3 Représentation matricielle

Un code LDPC est représenté par sa matrice de parité creuse  $\mathbf{H}$  de dimension  $(M \times N)$ , où  $M \leq (N - K)$  est le nombre de lignes ou le nombre des équations de contrôle de parité (noeuds de contrôle) et  $N$  est le nombre de colonnes (la longueur du mot de code). Nous pouvons donc définir deux nombres décrivant cette matrice :  $W_r$  pour le nombre de 1 dans chaque ligne et  $W_c$  pour les colonnes. Pour une matrice à faible densité, les deux conditions  $W_c \ll N$  et  $W_r \ll M$  sont vérifiées.

**Exemple :**

On considère la matrice de contrôle de parité suivante d'une taille  $(4 \times 6)$  et de rendement  $R = 2/3$ .



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

Les équations de parité associées à la matrice  $H$  sont :

$$x_2 + x_4 + x_5 = 0 \quad (3.1)$$

$$x_0 + x_3 + x_5 = 0 \quad (3.2)$$

$$x_1 + x_3 + x_4 = 0 \quad (3.3)$$

$$x_0 + x_1 + x_2 = 0 \quad (3.4)$$

### 3.3.4 Graphe de Tanner

[32]

La deuxième représentation des codes LDPC est la représentation graphique, connue sous le nom de graphe de Tanner, qui a été introduit par R. Michael Tanner en 1981. Le graphe de Tanner est un graphe bipartite, ce qui signifie qu'il est composé de deux ensembles de noeuds : l'ensemble des noeuds variables  $V = V_1, \dots, V_N$  et l'ensemble des noeuds de contrôle  $\mathbf{C} = c_1, \dots, c_M$ , et un ensemble d'arrêtes, tels que chaque arrête joint un noeud de  $V$  à un noeud de  $\mathbf{C}$ , sans que deux noeuds dans  $V$  ou dans  $\mathbf{C}$  soient connectés.

Dans le graphe de Tanner, les lignes de la matrice de parité (noeuds de contrôle) sont représentées par des carrés et les colonnes (noeuds de variable) sont représentées par des cercles et les 1 représentent les arrêtes du graphe. La figure 3.1 [29] montre le graphe de la matrice de contrôle de parité défini dans l'exemple que nous avons donné ci-haut.

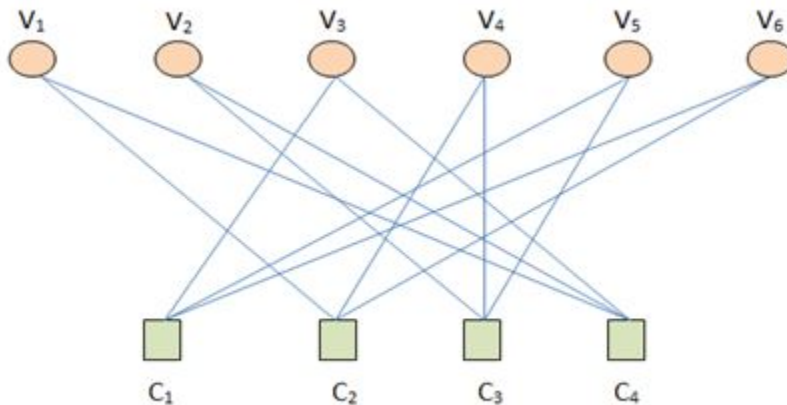


FIGURE 3.1 – Graphe de Tanner

Cette représentation sous forme de graphe permet d'utiliser la théorie des graphes pour construire et étudier les codes LDPC, elle est utile lorsqu'on souhaite se représenter

le principe de passage de messages (message passing), sur lequel est basé le décodage itératif [29].

**Remarque :**

Un mot du code est valide si pour chaque noeud de contrainte, tous les symboles qui sont reliés somment à 0 :  $\mathbf{H}_c^T = \mathbf{s} = 0$

Où  $\mathbf{s}$  : Syndrome checking.

### 3.3.5 Codes LDPC réguliers et irréguliers

Les codes LDPC sont classés en deux familles : les codes réguliers et les codes irréguliers. On parle d'un code LDPC régulier quand le nombre de 1 par ligne est égal au nombre de 1 par colonne, on appelle ces quantités : le degré de chaque ligne  $W_r$  et de chaque colonne  $W_c$ . Le nombre total de positions non nulles dans la matrice est égal au nombre d'arêtes du graphe [30][31][32].

Dans le cas d'un code LDPC régulier, on a la relation suivante : [29]

$$W_c \times N = W_r \times (N - K)$$

Une deuxième classe des codes LDPC, dite irrégulière, a été introduite par Luby et all. Ce sont des codes définis par des matrices de contrôle de parité dont les degrés de connexion  $W_r$  et  $W_c$  sont différents.

L'irrégularité de ces codes se spécifie à l'aide des deux polynômes :

— Polynôme associé aux noeuds de variables  $\lambda(x)$  :

$$\lambda(x) = \sum_{i=2}^{W_c} \lambda_i \cdot x^{i-1} \quad 0 \leq \lambda_i \leq 1 \quad \sum_{i=2}^{W_c} \lambda_i = 1$$

Où :  $\lambda_i$  caractérise la propagation du nombre de branches connectées à un noeud de variable de degré de connexion  $i$ .

— Polynôme associé aux noeuds de contrainte de parité  $p(x)$  :

$$p(x) = \sum_{i=2}^{W_c} p_i \cdot x^{i-1} \quad 0 \leq p_i \leq 1 \quad \sum_{i=2}^{W_c} p_i = 1$$

Où :  $p_i$  est la proportion de branches du graphe qui sont connectées à un noeud de contrainte de parité de degré  $i$ .

Ces quantités sont reliées au rendement du code et qui est donné par :

$$R = 1 - \frac{\sum_{i=2}^{W_c} \frac{p_i}{i}}{\sum_{i=2}^{W_c} \frac{\lambda_i}{i}}$$

## 3.4 Construction des codes LDPC

La construction d'un code LDPC doit être effectuée de manière à optimiser les performances de correction d'erreurs et la complexité d'implémentation du décodeur [33]. Pour cela, il existe plusieurs techniques pour les concevoir, citons certaines constructions aléatoires (de Gallager et de Mackay), ainsi que certaines constructions déterministes (celles basées sur les protographes, quasi-cyclique, etc.). La construction des codes LDPC consiste à définir les positions de tous les éléments non nuls dans la matrice de contrôle de parité  $\mathbf{H}$ . La difficulté principale dans la construction d'un code LDPC est le choix des paramètres qui le définissent.

## 3.5 Les codes LDPC 5G NR

La technologie d'accès NR marque une grande transition dans le codage de canal pour le 3GPP des réseaux cellulaires. Selon l'UIT-R, il existe trois principaux cas d'utilisation de la 5G NR définis par le 3GPP dans le cadre de son projet SMARTER (Study on New Services and Markets Technology Enablers) qui sont : eMBB (Enhanced Mobile Broadband), URLLC (Ultra-Reliable Low Latency Communications) et mMTC (Massive Machine Type Communications).

La phase initiale des déploiements de la 5ème génération se concentre sur le cas d'utilisation eMBB, qui peut être considéré comme une extension directe du service haut débit de la 4ème génération. Le scénario eMBB doit prendre en charge une gamme de débits plus large par rapport à la 4G [electronics]. Récemment, les codes LDPC ont été sélectionnés comme schéma de codage pour le canal de données (PSCH) 5G eMBB [33].

Ces dernières années, la recherche sur les codes LDPC s'est concentrée sur les codes LDPC connus sous le nom de codes de contrôle de parité quasi-cycliques à faible densité (QC-LDPC), qui ont de grands avantages par rapport aux autres types de codes LDPC et qui sont largement utilisés dans de nombreuses applications pratiques. Dans cette section, les codes QC-LDPC sont passés en revue, et les caractéristiques des codes 5G QC-LDPC sont résumées.

## 3.6 Les codes LDPC Quasi-cycliques

### 3.6.1 Préliminaires

Soit  $(Z \times Z)$  la taille d'une matrice identité de permutation circulante et  $\mathbf{P}_{(i,j)}$  la valeur de son décalage où  $i$  et  $j$  désignent la position de cette matrice identité dans la matrice  $\mathbf{H}$  à construire.

Pour toute valeur entière  $P_{(i,j)}$ , tel que  $0 \leq P_{(i,j)} \leq Z$ , une matrice de permutation circulante de taille  $(Z \times Z)$  est considérée comme un décalage circulaire de la matrice identité  $\mathbf{I}$  de taille  $(Z \times Z)$ , à droite de  $P_{(i,j)}$  positions. Cette matrice binaire de permutation circulaire est notée  $\mathbf{Q}(P_{(i,j)})$ .

Considérons  $\mathbf{Q}(\mathbf{1})$  comme exemple,  $\mathbf{Q}(\mathbf{1})$  est alors donné par :

$$\mathbf{Q}(\mathbf{1}) = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}$$

Pour une notation simple,  $\mathbf{Q}(-\mathbf{1})$  désigne la matrice nulle de même taille.

### 3.6.2 Introduction aux codes quasi-cycliques

Un code LDPC quasi-cycliques (QC-LDPC) est caractérisé par sa matrice de contrôle de parité  $\mathbf{H}$  qui est composée de sous-matrices  $Q(P_{(i,j)})$ , caractérisée par son facteur d'expansion  $Z$ . Les sous-matrices  $Q(P_{(i,j)})$ , appelées circulantes sont obtenues soit par décalage de la matrice identité de taille  $(Z \times Z)$ , soit égales à des matrices nulles [34].

Pour deux entiers positifs  $m_b$  et  $n_b$ , avec  $m_b \leq n_b$ , considérons le code QC-LDPC exprimé par la forme suivante :

$$\mathbf{H} = \begin{bmatrix} Q(P_{1,1}) & Q(P_{1,2}) & \dots & Q(P_{1,n_b}) \\ Q(P_{2,1}) & Q(P_{2,2}) & \dots & Q(P_{2,n_b}) \\ \vdots & \vdots & \ddots & \vdots \\ Q(P_{m_b,1}) & Q(P_{m_b,2}) & \dots & Q(P_{m_b,n_b}) \end{bmatrix}$$

La matrice des exposants de  $\mathbf{H}$ , qui est  $\mathbf{E}(\mathbf{H})$ , a la forme suivante :

$$\mathbf{E}(\mathbf{H}) = \begin{bmatrix} P_{1,1} & P_{1,2} & \dots & P_{1,n_b} \\ P_{2,1} & P_{2,2} & \dots & P_{2,n_b} \\ \vdots & \vdots & \ddots & \vdots \\ P_{m_b,1} & P_{m_b,2} & \dots & P_{m_b,n_b} \end{bmatrix}$$

Chaque entrée de la matrice  $\mathbf{E}$  est désignée par une valeur de décalage. Il convient de noter que la matrice de contrôle de parité  $\mathbf{H}$  peut être construite en développant la matrice d'exposant  $m_b \times n_b$   $\mathbf{E}(\mathbf{H})$ . Cette procédure est appelée construction de protographe [35].

La figure 3.2 montre un exemple de code QC-LDPC de taille  $448 \times 160$  et de facteur d'expansion  $Z = 32$ , dont les sous-matrices sont représentées par des nombres compris entre -1 et 31 qui ont la signification suivante, en représentant une description compressée de la matrice  $\mathbf{H}$  et les sous-matrices sont représentées par des nombres compris entre -1 et 31 qui ont la signification suivante [36] :

- -1 : la sous-matrice est nulle.
- 0 : la sous-matrice est une matrice identité.
- $P_{(i;j)=1}$  à 31 : la sous-matrice est une matrice identité décalée  $P_{(i;j)}$  fois vers la droite.

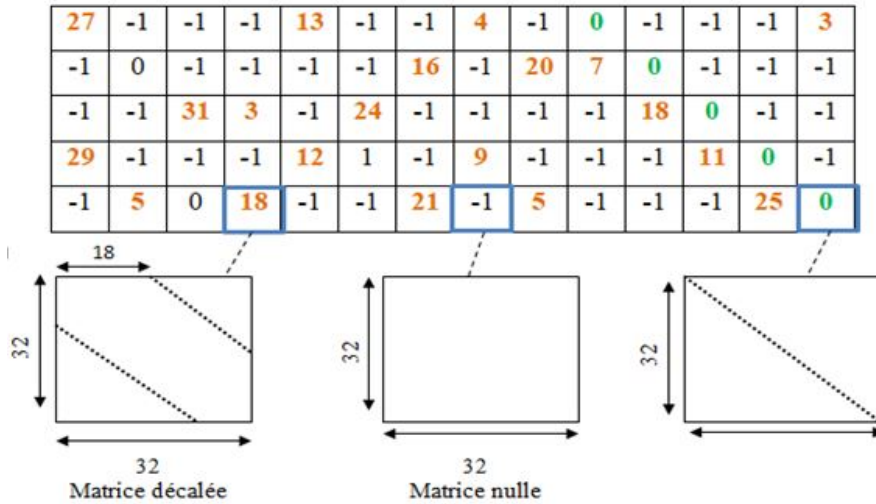


FIGURE 3.2 – Un exemple de code QC-LDPC

### 3.6.3 Structure des codes QC-LDPC 5G NR

Comme c'est mentionné ci-dessus, les codes QC-LDPC jouent un rôle très important dans les communications 5G NR et ont été acceptés comme schéma de codage de canal de données eMBB 5G dans la réunion standard 3GPP.

La figure 3.4 illustre la structure générale du graphe de base des codes QC-LDPC NR, dont les colonnes sont divisées en trois parties : colonnes d’informations, colonnes de parité de base et les colonnes de parité d’extension. Les lignes de ce graphe de base, sont divisées en deux parties : les lignes de contrôle de base et les lignes de contrôle d’extension.

## 3.7 Graphes de base et matrice de contrôle de parité

### 3.7.1 Analyse de graphes de base

Pour prendre en charge un taux compatible et une transmission évolutive des données, le 3GPP a accepté d’envisager deux graphes de base (BG) compatibles adopté pour les codes LDPC 5G [37] pour le codage de canal. Le graphe de base est une matrice où chacune des entrées peut être développée en fonction du facteur d’expansion  $Z_c$ , et leur utilisation est déterminée par le rendement du code et la taille des bits d’informations. Les deux types de BG envisagés consistent en le graphe de base 1 ( $BG_1$ ) qui est de taille ( $46 \times 68$ ) et le graphe de base 2 ( $BG_2$ ) de taille ( $42 \times 52$ ). Ces deux matrices ont une structure similaire présentée par la figure 3.5, la différence entre eux est que ( $BG_1$ ) est conçu pour des longueurs d’informations ( $500 \leq k \leq 8448$ ) et des taux de codage de  $\frac{1}{3}$  à  $\frac{8}{9}$ , tandis que ( $BG_2$ ) est conçu pour des longueurs d’informations plus petites ne dépassant pas 3840 et des taux de codage de  $\frac{1}{5}$  à  $\frac{2}{3}$  [38].

Comme le montre la figure 3.4 la matrices de base se compose de cinq sous-matrices, nommées **A**, **B**, **O**, **C** et **I** : la sous-matrice **A** est réservée aux bits systématiques ; de plus, **B** correspond au premier ensemble de bits de parité, qui est aussi une matrice carrée avec une structure bi-diagonale et sa première colonne et de poids trois ; **O** est une matrice entièrement nulle ; **C** correspond aux lignes de contrôle de parité unique et **I** est une sous-matrice identité [39]. La combinaison de **A** et **B** est définie comme le noyau et les trois sous-matrices **O**, **I** et **C** sont appelées extension. Il est clair que cette structure de code est similaire à l’extension de type Raptor décrit dans [40].

Pour ( $BG_1$ ) et ( $BG_2$ ), et pour tous les taux de codage les bits de code correspondant aux deux premières colonnes circulantes sont perforés avant la transmission. Cela est dû au poids de colonnes qui est élevé dans ces deux blocs perforés par rapport à d’autres colonnes de la matrice de base.

$Z_c$	j								
	0	1	2	3	4	5	6	7	
2	2	4	8	16	32	64	128	256	
3	3	6	12	24	48	96	192	384	
5	5	10	20	40	80	160	320		
7	7	14	28	56	112	224			
9	9	18	36	72	144	288			
11	11	22	44	88	176	352			
13	13	26	52	104	208				
15	15	30	60	120	240				

TABLE 3.1 – Table de levage

Le nombre de conceptions de coefficient de décalage est de 8. Toutes les tailles de levage sont divisées en huit ensembles basés sur le paramètre  $a$ , où  $a$  est utilisé pour la définition de la taille de levage  $Z_c = a2^j$  [34]. Les facteurs pris en compte pour déterminer les tailles de levage sont le degré de parallélisme possible et la complexité du réseau de commutation associé.

La taille de levage est sélectionnée de telle sorte qu'elle soit proche du nombre de bits d'information divisé par le nombre de colonnes de graphe de base correspondant aux bits d'information. Le choix de la longueur de code variable est effectué en choisissant la taille de levage appropriée en fonction de la longueur de bit d'information à coder (voir tableau 3.1) [35] Pour  $BG_1 : k = 22 \times Z$  Pour  $BG_2 : k = 10 \times Z$

### 3.7.2 Calcul de la matrice de contrôle de parité

A partir du graphe de base nous obtiendrons la matrice de contrôle de parité  $\mathbf{H}$ , et cela en remplaçant chaque élément du graphe de base par une matrice carrée de taille  $(Z \times Z)$ . Les éléments contenant dans le graphe de base sont des nombre compris entre la valeur -1 à  $Z-1$ . Chaque élément de valeur -1 dans le graphe de base est remplacé par une matrice entièrement nulle de taille  $(Z \times Z)$ , la valeur 0 est remplacée par une matrice identité, tandis que les autres éléments sont remplacés par une matrice identité de permutation circulaire  $\mathbf{I}(p_{(i,j)})$ .

L'utilité de la méthode de permutation circulaire consiste à réduire la mémoire nécessaire à la mise en oeuvre et faciliter l'utilisation d'un simple réseau de commutation pour le codage et le décodage.

#### Exemple :

Considérons un simple exemple explicatif du principe d'obtention de la matrice de contrôle de parité  $\mathbf{H}$  donnée par l'équation.  $\mathbf{H}$  est alors être obtenu par la matrice génératrice  $\mathbf{L}$ , de facteur d'expansion  $Z=5$ , et qui est donnée par : [43]

$$\mathbf{L} = \begin{bmatrix} 2 & 3 & -1 & 0 & 4 \\ 1 & -1 & 1 & 2 & 0 \\ 1 & 1 & 4 & 0 & 0 \\ 4 & 3 & 0 & 1 & 2 \end{bmatrix}$$

Graphe de base  $\mathbf{L}$

## 3.8 Encodage des codes LDPC

Comme tout code linéaire, l'encodage des codes LDPC consiste à multiplier le mot d'information  $\mathbf{m}$  par sa matrice génératrice  $\mathbf{G}$  afin d'obtenir le mot de code désiré  $\mathbf{C}$ .

$$\mathbf{C} = \mathbf{m} \times \dots (a)$$

Cependant, il est plus efficace d'utiliser plutôt la matrice de contrôle de parité [44].

Les travaux de T.J. Richardson et R.L Urbanke [46] ont prouvés que la matrice de contrôle doit subir un prétraitement avant l'opération d'encodage afin de mettre la matrice  $\mathbf{H}$  sous une forme presque triangulaire inférieure, en utilisant uniquement des permutations de lignes ou de colonnes, comme illustré sur la figure 3.4 [44], cette matrice est composée de 6 sous-matrices creuses, notées  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ ,  $\mathbf{E}$  et d'une sous-matrice triangulaire inférieure  $\mathbf{T}$  de taille  $(M - G \times M - G)$ .

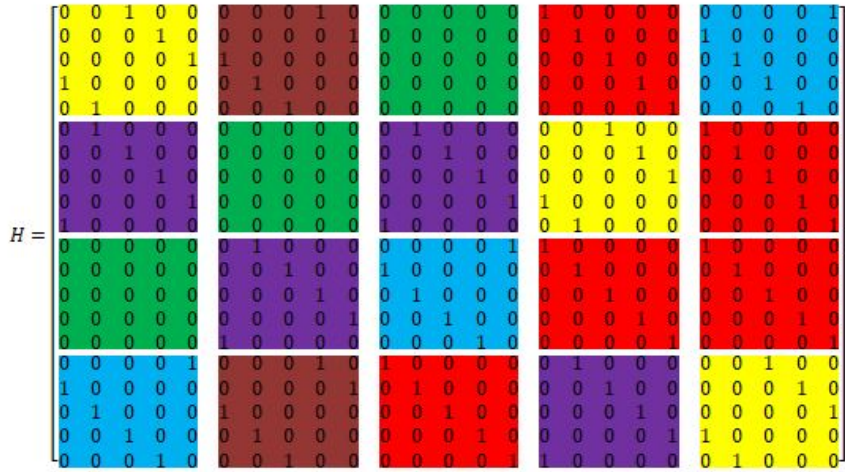


FIGURE 3.3 – La matrice de contrôle de parité  $H$

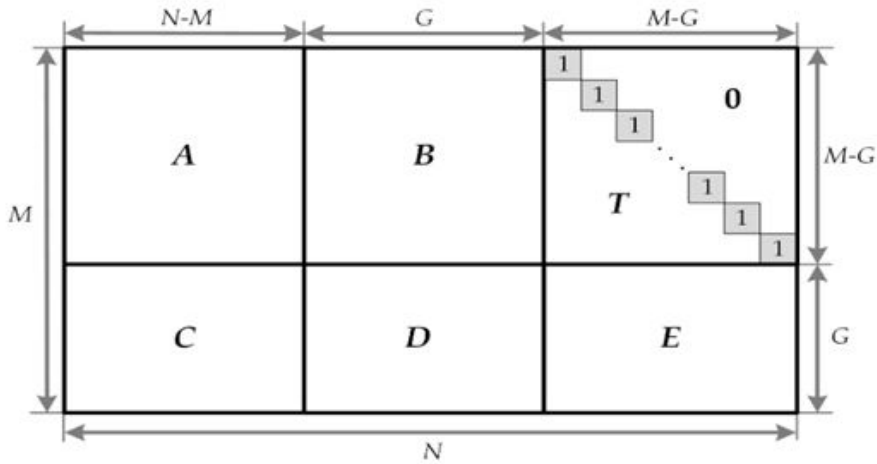


FIGURE 3.4 – Les matrices utilisées pour l'encodage LDPC

Par conséquent, l'algorithme d'encodage des codes LDPC se compose de deux étapes : une étape de prétraitement et une étape d'encodage proprement dite. Tout d'abord, dans la 1ère étape de prétraitement), comme mentionné ci-dessus, la matrice de contrôle de parité  $H$  de taille  $(M \times N)$  est transformée autant que possible en une matrice de forme triangulaire inférieure, où  $N$  est la longueur de bloc du code et  $M$  est le nombre d'équations de contrôle de parité, c'est ce qu'on appelle (Triangulation). Étant donné que la transformation matricielle est réalisée uniquement par permutations lignes et colonnes, la matrice  $H$  reste une matrice creuse :

$$H_T = \begin{bmatrix} A & B & T \\ C & D & E \end{bmatrix}$$

Ici, la matrice  $T$  a une forme triangulaire inférieure avec des 1 le long de la diagonale, et toutes les entrées au-dessus de la diagonale sont des 0. En multipliant  $H$  de gauche par [34] :

$$\begin{bmatrix} I & 0 \\ -ET^{-1} & I \end{bmatrix}$$

On obtient :

$$H' = \begin{bmatrix} A & B & T \\ C' & D' & 0 \end{bmatrix}$$

Où :

$$\begin{aligned} C' &= -ET^{-1}A + C \\ D' &= -ET^{-1}B + D \\ E' &= -ET^{-1}T + E = 0 \end{aligned}$$

Soit le mot de code  $\mathbf{C} = (s, p_1, p_2)$ , où  $s$  est les bits d'information,  $p_1$  désigne les  $G$  premiers bits de contrôle de parité et  $p_2$  contient les  $(M - G)$  bits de contrôle de parité restants. Ce mot de code  $\mathbf{C}$  doit satisfaire l'équation de contrôle de parité :

$$\mathbf{H}\mathbf{C}^T = 0$$

En effectuant cette multiplication, on obtient les deux équations suivantes :

$$As^T + Bp_1^T + Tp_2^T = 0 \quad (3.5)$$

$$C's^T + D'p_1^T + 0p_2^T = 0 \quad (3.6)$$

Où  $p_1^T$  et  $p_2^T$  peuvent être calculés à partir des équations suivantes :

$$p_1^T = D'^{-1} + C's^T \quad (3.7)$$

$$p_2^T = -T^{-1}(As^T + Bp_1^T) \quad (3.8)$$

Le codage par blocs permet de réduire fortement la complexité de l'opération de multiplication matricielle qui est exprimée par l'équation (a), lorsque les blocs d'information à coder sont de grande taille.

### 3.9 Décodage des codes LDPC

Par rapport aux autres types de codes, le travail le plus difficile est de trouver les meilleures méthodes pour construire des codes LDPC efficaces, par contre, leur décodage ne pose pas autant de problèmes pour les chercheurs que leur construction [47].

Un code LDPC peut être décodé par plusieurs méthodes appelées (algorithmes de décodage). Ces algorithmes de décodage présentent une grande diversité au niveau de la complexité calculatoire et des performances en termes de pouvoir de correction d'erreurs [36]. Dans le cadre du décodage canal, il est déjà possible de différencier globalement deux principaux types de décodage décrits dans la figure 3.5 [36].



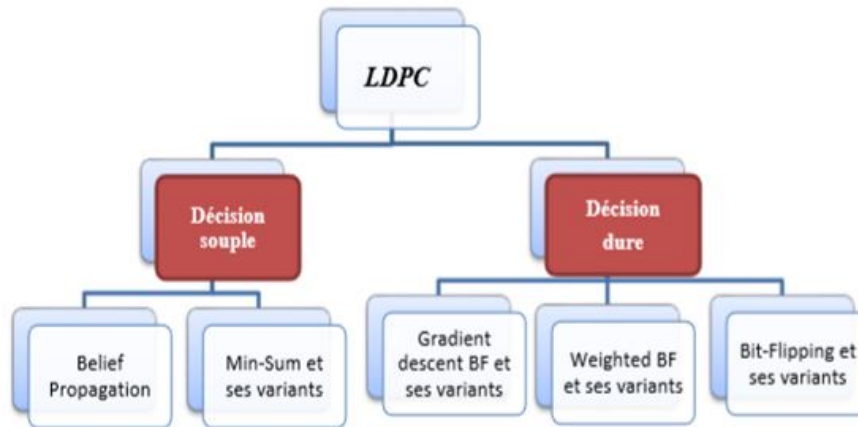


FIGURE 3.5 – Types d’algorithmes de décodage

- **Le décodage à entrées dures (Hard-decision)** : dans ce type de décodage le message transmis contient la valeur réelle des bits, c’est-à-dire le message est la valeur que doit prendre le noeud destinataire selon le noeud émetteur. Cela signifie qu’il doit se baser sur les bits estimés par le démodulateur pour corriger les éventuelles erreurs de transmission. Gallager proposait déjà trois algorithmes pour décoder des informations dures ils se nomment Gallager algorithm A (Gallager A), Gallager algorithm B (Gallager B), et Bit-Flipping algorithm (BF) [48]. L’algorithme Bit-Flipping est considéré comme le meilleur exemple de décodage à décision dure qui sera détaillé prochainement.
- **Le décodage à entrées souples (Soft-decision)** : Pour des décodeurs de type soft, le message transmis est la valeur de probabilité sur le noeud destinataire conditionnellement au noeud émetteur (principe d’échange d’informations probabilistes). Elle tente d’approximer au mieux la solution de décodage à maximum de vraisemblance. Cette méthode est basée sur l’idée de propagation de croyance, pour cela elle porte plusieurs noms dans la littérature comme : algorithme de propagation de croyance (BP : belief propagation) ou algorithme MPA (Message Passing Algorithm)[49].

Les algorithmes de décodage à décision souple dans le domaine logarithmique offrent les meilleures performances que les algorithmes de décodage à décision dure, quel que soit le niveau de SNR [51].

### 3.10 Propagation de croyance

L’algorithme par propagation de croyances, aussi appelé Belief Propagation (BP), ou Sum Product Algorithm (SPA), permet d’effectuer le décodage des codes LDPC. Le décodage LDPC illustré dans la figure 3.6, tente de corriger les erreurs à l’aide d’algorithmes itératifs de message qui utilisent un certain nombre d’itérations pour estimer la vraisemblance d’un symbole.



FIGURE 3.6 – Décodeur LDPC

Le graphe de Tanner est un excellent moyen pour illustrer le passage itératif des messages, selon lequel à chaque itération, les messages sont transmis des noeuds variables aux noeuds de contrôle et vice versa. Ce type de décodage peut être divisé en deux parties : l'opération des noeuds de variables (opérations lignes) et l'opération de noeuds de contrôle (opérations colonnes). Chaque itération de l'algorithme de propagation de croyance est composée de deux étapes [44][50] :

- Etape de mise à jour des messages lorsqu'ils vont des noeuds variables aux noeuds de contrôle comme le montre la figure 3.7, c'est ce qu'on appelle Data pass, dans laquelle toutes les informations que possède  $v_1$  sont envoyées au noeud de vérification  $c_3$  à l'exception des informations que le noeud de vérification  $c_3$  possède déjà.
- La deuxième étape de mise à jour des messages lorsqu'ils vont des noeuds de contrôle aux noeuds de variables comme le montre la figure 3.7, appelée Check pass, dans lequel le noeud de contrôle  $c_1$  transmet toutes les informations dont il dispose à chacun des noeuds variables  $v_i$  à l'exclusion des informations que le noeud récepteur possède déjà.

On trouve plusieurs algorithmes itératifs de passage de message dû à la diversité de types des messages passés et aux différents calculs aux noeuds, tels que SPA et MSA. On appelle croyance, la probabilité du message transmis entre les noeuds de variables et les noeuds de contrôle. Comme  $q_{13}(b)$  et  $r_{15}(b)$ .

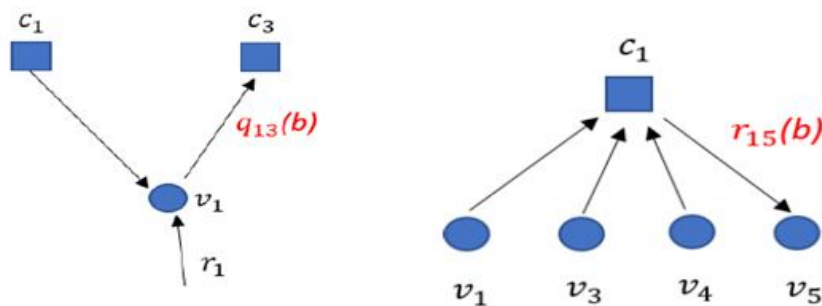


FIGURE 3.7 – Itérations entre les noeuds de variables et les noeuds de contrôle

Les messages utilisés dans ce mémoire seront des log-rapports de vraisemblance. On note que :  $v = \log \frac{p(y/c=0)}{p(y/c=1)} = \log \frac{v[0]}{v[1]}$  le message de sortie d'un noeud de variable.

Et :  $u = \log \frac{p(y'/c'=0)}{p(y'/c'=1)} = \log \frac{u[0]}{u[1]}$  le message de sortie d'un noeud de contrôle.

#### 1. Mise à jour des messages sur un noeud de variable : Data pass

Lors de la mise à jour sur le noeud de variable  $k$  de degré  $i$  à l'itération  $l$ , le message envoyé sur la  $q^{ieme}$  branche est donné par la relation suivante [44][50] :

$$v_q^{(l)} = u_0 + \sum_{i=1; i \neq q}^i u_n^{l-1} \quad \forall q = 1, \dots, i$$

Où  $u_n (n = 1, \dots, i)$  représentent les messages venant des noeuds de contrôle voisins et  $u_0 = \log \frac{p(y_k/c_k=0)}{p(y_k/c_k=1)}$  représente le log-rapport de vraisemblance observé en sortie du canal.

## 2. Mise à jour des messages sur un noeud de contrôle : check pass

Lors de la mise à jour sur un noeud de contrôle de degré  $j$ , le message de sortie sur la  $p^{\text{ieme}}$  branche est donné par l'expression [44][50].

$$\tanh \frac{u_p^l}{2} = \prod_{m=1; m \neq p}^j \tanh \frac{v_m^l}{2} \quad \forall p = 1, \dots, j \quad (3.9)$$

Où  $v_m, m = 1, \dots, j$ , sont les messages venant des noeuds de données voisins.

L'algorithme de passage de messages peut être représenté dans le domaine des probabilités et dans le domaine logarithmique, les probabilités étant représentées par des rapports logarithmiques de vraisemblance (LLR) [50] donnée par la formule suivante :

$$LLR(y_i/v_i) = \ln \left( \frac{p(y_i/v_i = 0)}{p(y_i/v_i = 1)} \right) \quad (3.10)$$

L'algorithme de décodage dans le domaine LLR peut réduire la complexité de mise en oeuvre car les multiplications dans le domaine de probabilité peuvent être exprimées comme des additions dans le domaine logarithmique. En outre, de nombreuses multiplications des probabilités impliquées pourraient devenir numériquement instables, c'est pourquoi l'algorithme de domaine logarithmique est préféré dans le décodage LDPC [50]. Dans ce mémoire, nous allons travailler avec les algorithmes de domaine logarithmique.

Les algorithmes de décodage les plus importants abordés dans ce projet sont : Sum-Product et Minimum-Sum de décodage à décision souple et l'algorithme Bit-Flipping de décodage à décision dure.

## 3.11 Algorithmes de décodage LDPC

### 3.11.1 Algorithme décodage Bit-Flipping

Bit-Flipping est un algorithme qui a été proposé par Gallager dans sa thèse de doctorat qui travaille en décisions dures [53]. Cet algorithme de décodage consiste à inverser les valeurs des bits estimés comme erronées de manière à corriger les erreurs de transmissions. La mise en oeuvre de Bit-Flipping est décrite selon l'organigramme illustré au début de chapitre. Les messages transmis au long des bords (arêtes) du graphe de Tanner sont des séquences binaires.

Pour commencer, un noeud de bit envoie un message indiquant sa valeur (0 ou 1), et chaque noeud de contrôle envoie un message à chaque noeud de bit connecté en indiquant la valeur du bit sur la base des informations disponibles pour le noeud de contrôle. Les noeuds de contrôle déterminent que leur équation de contrôle de parité est satisfaite en calculant le vecteur de syndrome  $\mathbf{s}$  donné par l'équation :

$$\mathbf{s} = \mathbf{r} \times \mathbf{H}^t \quad (3.11)$$

Si l'équation (3.11) égale à un vecteur nul donc le mot reçu  $r$  par le récepteur est un mot de code valide, alors le processus de décodage est terminé et le mot de code de sortie sera le même que le mot de code reçu. Si les bits du syndrome ne sont pas des zéros, le décodeur calculera l'équation de vérification de parité insatisfaite  $y$  représentée par la formule suivante :

$$y = s \times H \quad (3.12)$$

Où la sortie  $y$  sera utilisée pour identifier les positions de bits inversés dans le vecteur reçu.

Pour chaque bit dans  $y$ , si le bit est supérieur ou égal au bit précédent, le bit correspondant dans le mot de code reçu est inversé pour obtenir le mot de code décodé [52]. en calculant la fonction d'inversion noté  $\Delta k$  exprimée par :

$$\Delta k^{BF}(x) = \sum_{i \in M(n)} \prod_{j \in N(i)} x_j \quad (3.13)$$

Si le nombre maximum d'itérations est atteint alors le décodage s'arrête, sinon le processus de décodage recommence jusqu'à ce que toutes les sommes de contrôle de parité soient satisfaites.

Le décodage Bit-Flipping corrige beaucoup de séquence d'erreur qui possède plus de bits erronés que la capacité du code pour corriger les erreurs. Nous avons vu que Le décodeur de Bit-Flipping peut être immédiatement arrêté chaque fois qu'un mot de code valide a été trouvé en vérifiant l'équation (3.11). Cela a l'avantage d'une part d'éviter les itérations supplémentaires une fois qu'une solution a été trouvée, et d'autre part, un échec de convergence vers un mot de code est toujours détecté.

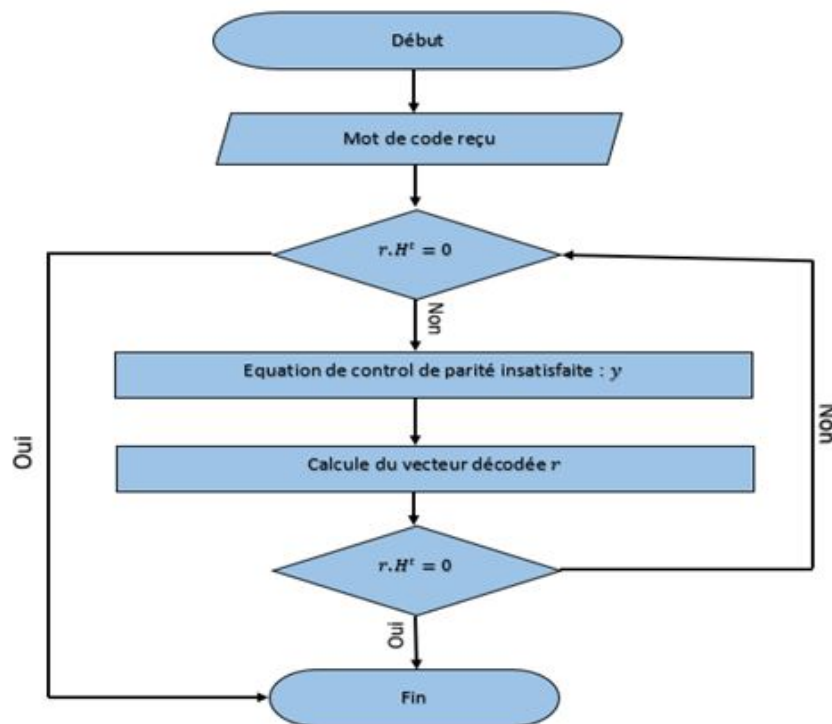


FIGURE 3.8 – Logigramme de décodage Bit-Flipping

### 3.11.2 Algorithme Sum-Product

L'algorithme Sum-Produit (SPA) est la méthode de décodage la plus performante, connu aussi sous le nom d'algorithme Belief propagation (BP), a été développé par Mackay et Neal [53]. SPA est un algorithme de passage de messages à décision souple permettant d'effectuer le décodage des codes LDPC qui est similaire à l'algorithme Bit-Flipping décrit dans la section précédente, la principale différence entre eux est que les messages de SPA sont transmis sous forme de probabilités. Les probabilités sont connues avant de lancer le processus de décodage LDPC pour décider à partir de quel bit démarrer, pour cela les probabilités de bits d'entrée sont appelées probabilités a priori pour les bits reçus, et les probabilités de bits renvoyées par le décodeur sont appelées probabilités a posteriori.

L'objectif du SPA est de calculer la probabilité maximale a posteriori pour chacun des bits codés en entrée. Les informations obtenues par les contrôles de parité effectués sur les LLR dans le processus de décodage sont connues sous les messages extrinsèques de ce bit codé particulier. Avant d'aller plus loin, il est nécessaire de rappeler l'expression du calcul qui permet d'estimer les messages sortants des  $CN(c2v_{ij})$  dans l'algorithme BP-SPA [36].

$$c2v_{ji} = \left( \prod_{k \in N(j)-i} \text{sign}(v2c_{kj}) \times f\left( \sum_{k \in N(j)} f(|v2c_{kj}|) \right) \right) \quad (3.14)$$

Tels que  $(1 \leq i \leq N)$  et  $(1 \leq j \leq M)$ , où  $M, N$  sont respectivement le nombre de lignes et de colonnes de la matrice de contrôle de parité  $\mathbf{H}$ . Ce calcul peut se décomposer en deux étapes, la première est le calcul du signe du message sortant et la seconde correspond au calcul de son amplitude.

Pour un décodage à décision souples, ces probabilités sont représentées sous la forme de rapports de vraisemblance log (LLR) comme indiqué par l'équation : [52]

$$L(x) = \log\left(\frac{p(x=0)}{p(x=1)}\right) \quad (3.15)$$

étapes impliquées dans l'algorithme du SPA sont [49] :

— **Étape 1** : Initialisation.

Dans cette étape le seuil de la décision des messages reçus  $y(y_1 y_n)$  vaut toujours 0, ces messages peuvent être traduits en binaire  $x(x_1 \dots x_n)$ . On calcule les syndromes, s'ils sont égaux à 1, il n'y a pas d'erreurs dans les bits reçus, on arrête le décodage et on obtient le mot de code  $\mathbf{C}(c_1 \dots c_n)$ , sinon on commence l'étape 2.

— **Étape 2** : Mise à jour des noeuds de variable.

Pour chaque bit  $x_i$ , les noeuds de variable sont initialisés avec les LLRs du canal.

$$V_{n \rightarrow m} = L(xy)_{AWGN} = \log \frac{\text{Pr}(x=1/y)}{\text{Pr}(x=0/y)} = \frac{2y_n}{\sigma^2}$$

Les noeuds de parité sont initialisés avec 0 :

$$c_n = 0$$

— **Étape 3** : Mise à jour des noeuds de parité.

Les noeuds de parité sont renouvelés en prenant en compte les probabilités issues des noeuds de variable :

$$E_{n \rightarrow m}^{BP}(x_n) = 2 \tanh^{-1} \left\{ \prod_{n' \in N(m) \setminus n} \tanh \left[ \frac{V_{n' \rightarrow m}(x_{n'})}{2} \right] \right\} \quad (3.16)$$

Où  $n' \in N(m)$   $n$  qui représente l'ensemble des bits (noeuds de variable) liés au noeud de parité  $m$  sauf le  $n$ me bit.

Les noeuds de variables sont mis à jour selon la formule :

$$V_{n \rightarrow m}(x_n) = \frac{2y_n}{\sigma^2} + \sum_{m' \in M(n) \setminus m} E_{m' \rightarrow n}^{BP}(x_n) \quad (3.17)$$

Où  $m' \in M(n)$   $m$  représente l'ensemble des noeuds de parité qui sont liés au noeud de variable  $n$  sauf le meme noeud de parité.

#### Etape 4 : Décision des bits

On prend une décision dure sur les bits  $\hat{x}_n(\hat{x}_1, \dots, \hat{x}_n)$  selon la règle :

$$\hat{x}_i = \begin{cases} 0, & \text{si } V_i(x_i) \leq 0 \text{ avec } i \in \{1, \dots, n\} \\ 1, & \text{si } V_i(x_i) \geq 0 \text{ avec } i \in \{1, \dots, n\} \end{cases}$$

D'où

$$V_i(x_i) = \frac{2y_n}{\sigma^2} + \sum_{m' \in M(n)} E_{m' \rightarrow n}^{BP}(x_n) \quad (3.18)$$

#### Etape 5 : Re-vérification

On recalcule les syndromes par l'équation  $\hat{x}_n \times \mathbf{H}^T$ , s'ils sont tous égaux à 1, on arrête le décodage et on sort le mot de code  $\hat{x}_n$ , sinon on continue l'étape 2. Le décodage va s'arrêter lorsque les syndromes sont tous égaux à 1 ou lorsque le nombre maximum d'itérations est atteint.

### 3.11.3 Algorithme Min-Sum

L'algorithme BP atteint les performances de décodage optimales très proches de la limite de Shannon [53] mais nécessite une complexité de calcul élevée. Pour cette raison, l'algorithme Min-Sum (MS) a été proposé pour diminuer la complexité de calcul en utilisant uniquement les opérations de comparaison et d'addition [46].

L'opération sur les noeuds de contrôle pour l'algorithme Min-Sum est similaire à celle pour Sum-Product, la différence réside dans l'opération sur les noeuds de variables. Ainsi que la procédure de mise en oeuvre de ces deux algorithmes est la même à l'exception de l'étape 3 où les tangentes hyperboliques sont remplacées par des calculs de minimums, c'est-à-dire au lieu de calculer  $\tanh^{-1}$  et  $\tanh$ , l'algorithme Min-Sum permet d'approximer la valeur absolue des messages produits par les noeuds de parité. Donc on obtient l'équation (3.19) : [53]

$$E_{m \rightarrow n}^{MS}(x_n) = \gamma_m * \mu_{m, \min} \quad (3.19)$$

Où

$$\begin{aligned} \gamma_m &= XOR\{signe(V_{n \rightarrow m} \mid n' \in N(m) \setminus n)\} \\ \mu_{m, \min} &= \min\{abs(V_{n \rightarrow m}) \mid n' \in N(m) \setminus n\} \end{aligned}$$

Et  $E_{m \rightarrow n}^{MS}(x_n)$  est la vitesse de convergence de l'algorithme Min-Sum.

L'algorithme Min-Sum est l'approximation la plus connue des algorithmes à décision souple, qui surévalue les valeurs absolues des messages des noeuds de contrôle vers les noeuds de variables ( $c2v$ ) selon la formule suivante [53] :

$$|c2v_{ji}| = \max(\min_{i' \in N(j)_i}(|c2v_{i'j}|))$$

Afin de compenser cette surévaluation de la valeur des messages, les algorithmes Offset Min-Sum et Normalized Min-Sum ont été introduits.

#### A- L'algorithme Offset Min-Sum (OMS) [48]

Cet algorithme ajoute un terme de compensation noté  $\alpha$  dans le calcul des informations extrinsèques. Ce terme constant durant le décodage minimise la surévaluation des message ( $c2v$ ), tels que :

$$|c2v_{ji}| = \max(\min_{i' \in N(j)_i}(|c2v_{i'j}|) - \alpha, 0)$$

Où  $\alpha$  est une constante positive. Une saturation est effectuée sur le résultat pour éviter d'inverser le signe quand  $\min(|c2v_{i'j}|) < \alpha$ .

#### B- L'algorithme Normalized Min-Sum (NMS) [48]

L'algorithme Normalized Min-Sum (NMS) utilise, quant à lui, un facteur de normalisation  $\gamma$  afin de minimiser la surévaluation des messages ( $c2v$ ). Le calcul de ces messages se fait à partir de l'expression suivante :

$$|c2v_{ji}| = \gamma(\min_{i' \in N(j)_i}(|c2v_{i'j}|))$$

La valeur du facteur de normalisation  $\gamma$  est comprise entre 0 et 1.

## 3.12 Conclusion

Dans ce chapitre, nous avons décrit les notions de base sur les codes LDPC et la représentation matricielle et graphique par le graphe de Tanner qui est une représentation utile des codes blocs linéaires et en particulier des codes LDPC. Nous avons parlé sur la différence entre les codes LDPC réguliers et irréguliers, ainsi que leurs l'utilisation dans la 5G NR en se basant sur les codes Quasi-cyclique, la construction des deux graphes de base adoptés aux codes LDPC 5G NR et le calcul de leurs la matrice de contrôle de parité. A la fin, nous avons expliqué la technique principale de codage et décodage LDPC en détaillant quelques algorithmes de décodage.

# Chapitre 4

## Simulation et résultats

### 4.1 Introduction

Après avoir finalisé la partie théorique qui nous a servi comme introduction pour notre étude pratique, il est temps alors de passer à la description pratique de notre travail. Dans le cadre de ce chapitre, nous proposons de présenter les résultats obtenus lors de nos simulations dans un environnement Matlab. Nous allons établir la simulation de décodage avec les deux différents algorithmes (BF et MSA) dans un canal bruité en calculant les performances en termes de BER en fonction du rapport signal sur bruit en montrant l'influence du nombre d'itérations de décodage et du rendement  $R$ .

Ce chapitre va premièrement commencer par une brève présentation du logiciel utilisé. Nous exposons ensuite la procédure que nous avons suivie pour réaliser ce travail où nous allons expliquer les étapes du programme utilisé afin d'effectuer la simulation. Puis nous discutons les résultats de décodage LDPC obtenus en appliquant les deux algorithmes les plus fréquemment utilisés.

### 4.2 Présentation du logiciel

Nous avons utilisé comme langage de programmation de haut niveau « *Matlab* » version (R2021b). Le choix de ce logiciel est lié à la facilité et la souplesse d'utilisation et qui est plus adapté à la 5G. En outre, Matlab (Matrix Laboratory) est un langage informatique très puissant particulièrement dédié aux applications scientifiques. Il est un outil de simulation et de programmation qui facilite la résolution des problèmes nécessitant une très grande puissance de calcul et de visualisation de données. Matlab est devenu aujourd'hui une bibliothèque très riche et simple qui permet :

- Le développement des algorithmes.
- La simulation et l'acquisition des données.
- Le calcul scientifique et l'exploration des résultats.





FIGURE 4.1 – Logo Matlab

## 4.3 Procédure de travail

L'objectif principal de notre travail est d'étudier les codes LDPC et leurs applications dans la cinquième génération. Les performances de ces codes sont étudiées en termes de taux d'erreur binaire (BER) en fonction du rapport signal sur bruit (SNR). Les algorithmes de décodage les plus pertinents ont été utilisés. Pour atteindre cet objectif, nous utilisons un script sous Matlab que nous avons adapté à notre étude.

### 4.3.1 Présentation du système :

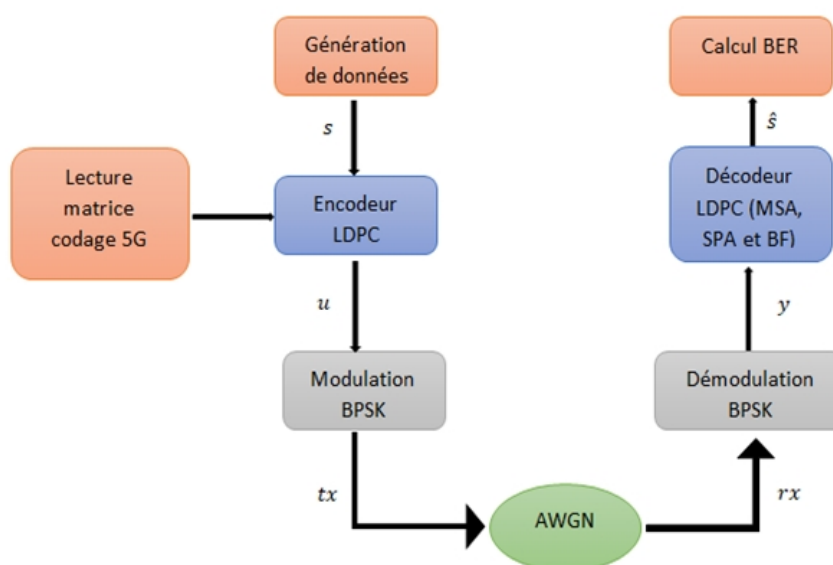


FIGURE 4.2 – Organigramme de la simulation

La figure 4.2 représente l'organigramme de la simulation, un bloc de bits de message  $s$  va être entré dans le codeur LDPC, et on obtient un mot codé  $u$  à la sortie de ce codeur, par la suite, en appliquant une modulation BPSK on obtient un nouveau mot de code  $tx$  qu'on va lui ajouter un bruit gaussien aditif (sigma). Ce mot de code  $tx$  est transmis par un canal de communication qui est modélisé comme AWGN avec une densité de puissance de bruit  $N_0/2$ .

Cet effet provoque des erreurs dans les données transmises, pour les corriger. Le mot de code reçu  $rx$  sera introduit dans un démodulateur BPSK pour inverser la modulation effectuée dans la partie émettrice, puis le mot de code démodulé  $y$  va être passé dans le décodeur LDPC et l'estimation du message d'origine  $\hat{s}$  est récupérée à la sortie de décodeur.

Ce schéma bloc représente un système de communication qui utilise un décodage à

décision dure. S'il s'agit d'une décision souple, le démodulateur BPSK va être placé après le décodeur LDPC, c'est-à-dire le mot de code reçu  $\mathbf{rx}$  est d'abord donné au décodeur LDPC puis au démodulateur BPSK.

Dans cette partie pratique, les performances BER des algorithmes de décodage à décision dure et à décision douce vont être analysées.

Avant de passer à l'interprétation des résultats obtenus, nous expliquons d'abord le déroulement de la simulation.

### 4.3.2 Déroulement de la simulation

Nous avons commencé par générer une matrice de contrôle de parité  $\mathbf{H}$  pour coder 100 mots de code avec une modulation BPSK pour des valeurs de SNR allant de 0 jusqu'à 20 dB. Ensuite, pour coder le mot de code  $\mathbf{s}$  généré aléatoirement par  $\mathbf{H}$  nous calculons tout d'abord sa matrice génératrice  $\mathbf{G}$  puis nous multiplions cette matrice  $\mathbf{G}$  par le message  $\mathbf{s}$ .

L'étape qui suit le codage est la modulation, dans notre cas nous avons effectué une modulation BPSK sur le mot de code  $\mathbf{u}$  pour qu'il soit adapté à une transmission sur un canal. Après avoir contrôlé la taille du bruit (sigma) en fonction des valeurs de SNR, on fait passer le mot obtenu  $\mathbf{tx}$  par un canal à bruit additif blanc gaussien (AWGN).

Pour arriver à l'étape finale, nous allons décoder premièrement le mot modulé  $\mathbf{rx}$  par l'algorithme Bit-Flipping et puis par l'algorithme Min-Sum en calculant la valeur de BER dans chaque cas.

Les étapes de ces deux algorithmes sont bien détaillées dans les fonctions Matlab utilisées dans le script principal.

Les résultats de cette simulation seront discutés dans la section suivante.

## 4.4 Implémentation et résultats

Dans cette section, la simulation de décodage des codes LDPC 5G NR est présentée. Nous considérons le nombre de mot de code, les différents rendements, le nombre d'itérations et les schémas de décodage comme des paramètres utilisés pour l'analyse des codes LDPC.

Les itérations définissent le nombre de fois que les informations sont évaluées avant de prendre une décision finale sur le mot de code reçu.

— **Analyse des codes LDPC avec des différents rendements :** Dans cette section nous exposons les performances des codes LDPC avec des différents rendements, les paramètres utilisés pour cette simulation sont illustrés dans le tableau suivant :

Nombre de mots de code	512
Valeurs de SNR	0 à 20
Valeurs de R	1/2, 1/4, 1/6
Nombre d'itérations	10

TABLE 4.1 – Table de paramètres utilisés

— Pour  $R=1/2$  :

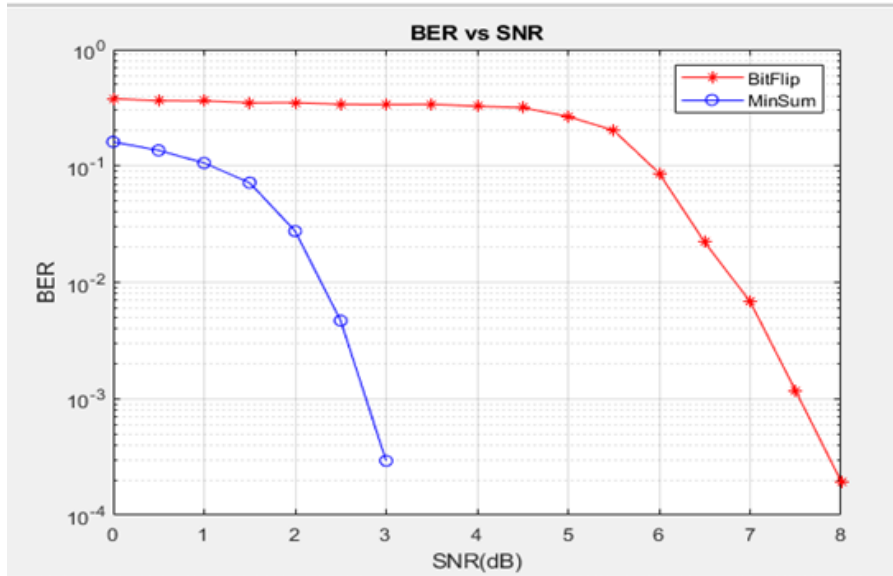


FIGURE 4.3 – BER vs SNR pour BF et pour MSA avec  $R=1/2$

— Pour  $R=1/4$  :

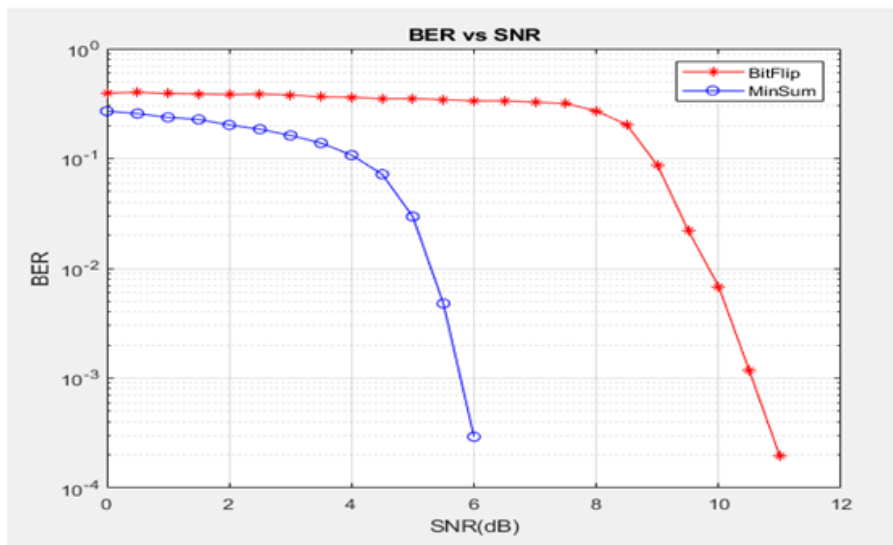


FIGURE 4.4 – BER vs SNR pour BF et pour MSA avec  $R=1/4$

— Pour  $R=1/6$  :

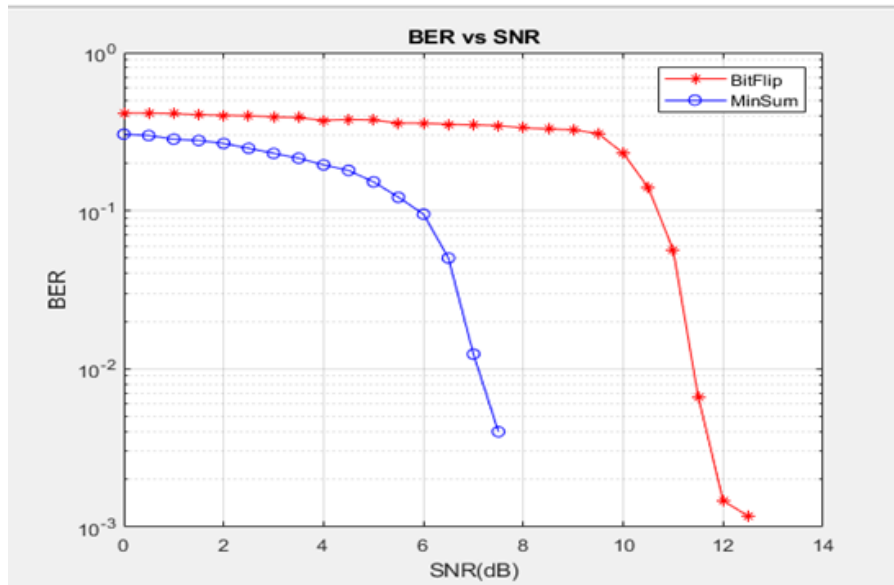


FIGURE 4.5 – BER vs SNR pour BF et pour MSA avec  $R=1/6$

A partir de la simulation nous pouvons observer qu'un faible rendement  $R$  donne de meilleures performances de BER et de SNR. Dans notre cas le BER à  $R=1/2$  est plus performant que d'autres rendements.

#### 4.4.1 Comparaison entre les deux algorithmes utilisés

Comme c'est mentionné précédemment nous avons utilisé deux algorithmes (BF et MSA), dans cette partie nous allons faire une étude comparative de ces derniers.

Après l'exécution du script nous obtiendrons la figure 4.6, qui montre les résultats comparatifs de cette comparaison basée sur BER.

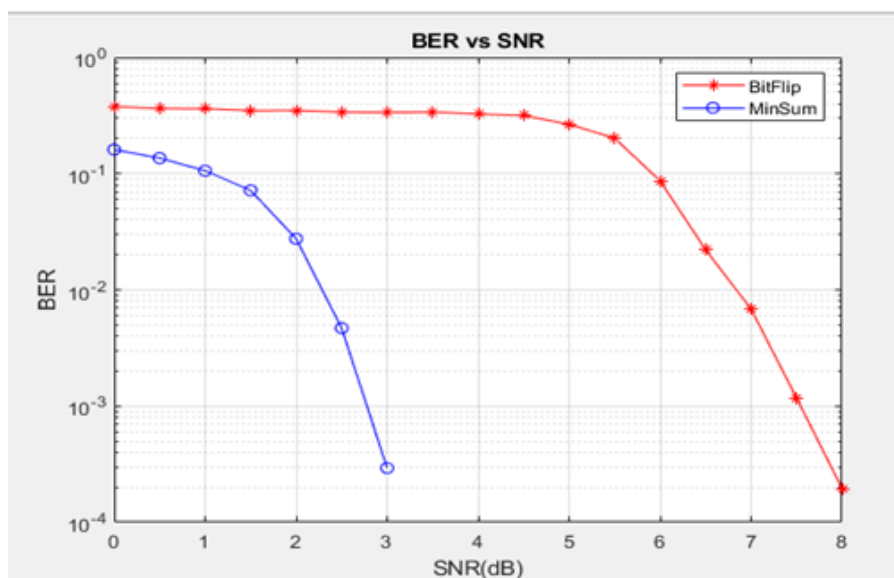


FIGURE 4.6 – Résultat de comparaison entre BF et MSA

Pour l'étude comparative, prenons la valeur de SNR à 3dB, pour décodage dur le nombre d'erreur est supérieur au nombre d'erreur pour un décodage souple, à titre d'exemple comme nous observons sur la figure 4.6 dans le décodage à décision souple la valeur  $10^{-1}$  de BER est obtenu pour un SNR de 1dB, mais dans le cas de décision dure la meme valeur de BER est obtenu à 6.8 dB.

D'après cette comparaison, nous concluons que le décodage à décision douce donne de meilleures performances.

#### 4.4.2 Influence du nombre d'itérations sur le décodage

Les figures montrent les résultats de différents algorithmes de décodage LDPC avec différents itérations de décodage.

— Pour itr=5 :

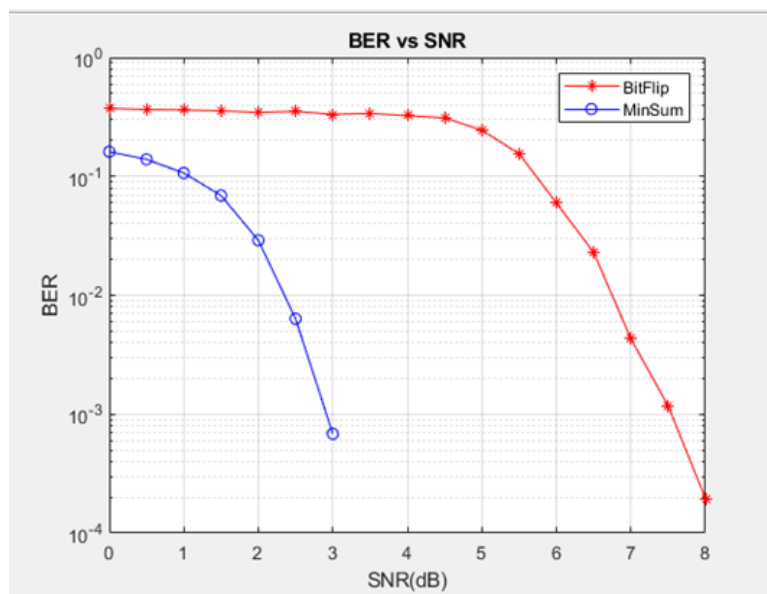


FIGURE 4.7 – BER vs SNR pour 5 itérations

— Pour  $\text{itr}=8$  :

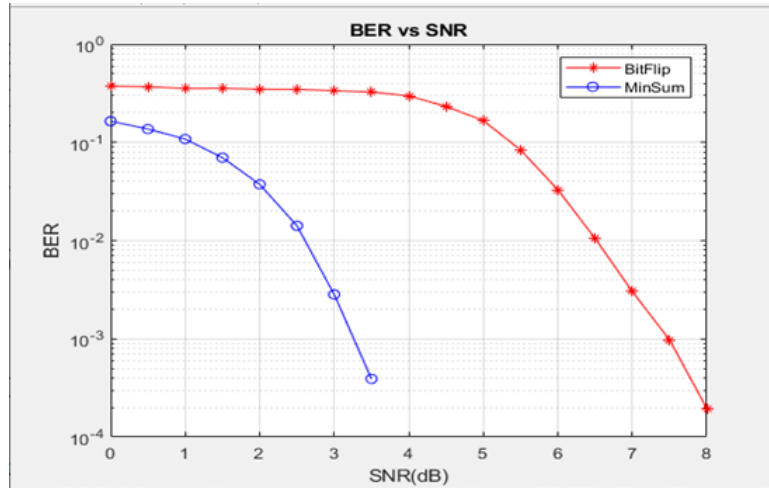


FIGURE 4.8 – BER vs SNR pour 8 itérations

— Pour  $\text{itr}=20$  :

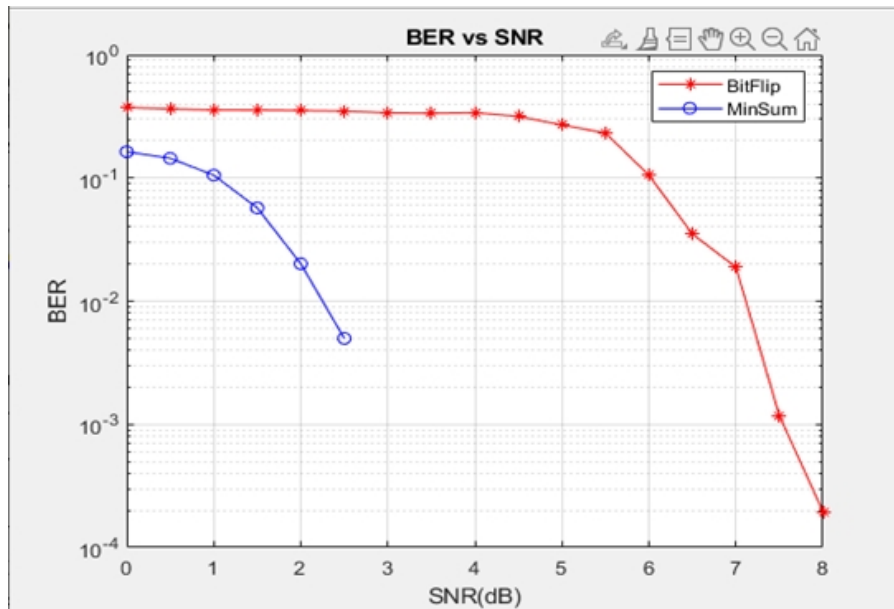


FIGURE 4.9 – BER vs SNR pour 20 itérations

D'après ces trois figures, nous observons que lorsqu'il s'agit d'un décodage Min-Sum, l'augmentation de nombre d'itérations améliore considérablement le gain de décodage. Le nombre d'itérations n'a aucune influence sur les performances de décodage à décision dure.

## 4.5 Conclusion

Dans ce chapitre, nous avons présenté les résultats de simulation des performances de décodage LDPC. Les performances du LDPC dépendent de sa conception et de la sélection des algorithmes de décodage. Les valeurs de BER de l'algorithme de décodage à décision souple ont été comparés au décodage de décision dure, cette comparaison a été effectuée à différents niveaux SNR.

Après cette étude, on conclut que l'algorithme de décodage de décision souple (MS) offre de meilleures performances BER que la décision dure quel que soit le niveau de SNR.

## Conclusion générale

Dans ces dernières années, l'utilisation de la communication numérique est devenue de plus en plus nombreuse. Par conséquent il est nécessaire d'avoir une chaîne de transmission plus fiable afin d'éviter les erreurs de transmission, et d'avoir des débits de transmission plus élevés et une meilleure performance. Pour cette raison une nouvelle génération de communication à vue le jour qui a choisi d'employer essentiellement les codes LDPC comme code de correction d'erreurs.

Notre mémoire a été structurée en quatre chapitres, le premier chapitre donne un aperçu général de la cinquième génération qui sert à augmenter le débit et offrir de meilleures performances.

Ensuite dans le second chapitre nous avons concentré sur le codage canal et les codes correcteurs d'erreurs les plus employés pour corriger les erreurs de la chaîne de transmission et minimiser le taux d'erreurs binaire (BER) et qui permet d'assurer de très grandes vitesses de transmission.

Le troisième chapitre nous l'avons consacré pour l'étude théorique détaillée des codes LDPC, dans lequel nous avons décrit ces codes à faible densité en donnant la spécification qui les distingue du reste des codes. Nous avons parlé des codes LDPC Quasi-cyclique qui sont utilisés dans la 5G. Nous avons ensuite expliqué la technique du codage de ces codes. Enfin, nous avons élaboré le décodage des codes LDPC en se basant sur l'algorithme BF, MSA et nous avons expliqué ainsi le principe de propagation de croyance et l'algorithme de décodage SPA.

Dans le quatrième chapitre nous avons essayé de justifier pratiquement la partie théorique et d'étudier les performances des codes LDPC dans le réseau 5G, en appliquant un script qui permet d'étudier ces performances avec l'utilisation des deux algorithmes BF et MSA en faisant une comparaison entre eux. Les résultats obtenus après la simulation nous ont confirmé que le meilleur algorithme qui est plus adapté à 5G est l'algorithme Min-Sum.



# Bibliographie

- [1] DAOUD Khedidja, SIMULATION COMPARATIVE DES TECHNIQUES FBMC ET OFDM POUR LES RESEAUX 5G, Mémoire Master en Réseaux et Systèmes de Télécommunication, Université de Tlemcen Abou Bekr Belkaid, 2016.
- [2] S.Robert, *Planification des réseaux mobiles* Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud (HEIG-Vd), Institute for Information and Communication Technologies (IICT), Juin 2003.
- [3] H.MERAH, *Conception dun MODEM de la quatrième génération (4G) des réseaux de mobiles à base de la technologie MC-CDMA*, thèse Magistère en Communication, Université FERAHAT ABBAS, 2012.
- [4] <https://itsocial.fr/actualites/levolution-reseaux-mobiles-de-2g-a-5g>.
- [5] <https://www.terrafemina.com/vie-pratique/economie-a-bourse/articles/33447-telephonie-mobile-les-cinq-objectifs-de-la-5g.html>
- [6] Ahmed Sassan, *5G NR-Architecture, Technology, Implimentation of 3GPP NEW Radio standads*, Academic Press, Elsevier, Paris, France, no.50-1/2, pp.86, 2019.
- [7] Benyarou. Mourad. *Optimisation des systèmes multi-antennes appliqués aux systèmes MC-CDMA*, Université Abou Bekr Belkaid, Thèse de Doctorat En Sciences, Laboratoire de Télécommunications-Tlemcen, Année 2012- 2013.
- [8] Roger Dib, *Conception et Validation dun simulateur de system OFDM pour les réseaux intérieurs sans fil*, École De Technologie Supérieure Université de Québec, 28 Avril 2003.
- [9] RANDRIANTSOA Tantely Nomena, *Performance et Optimisation des codes LDPC Binaires dans un Système MIMO-STBC*, Mémoire de fin d'étude en vue de l'obtention du Diplôme d'ingénieur, Ecole Supérieure Polytechnique Université DANTANANARIVO, 2011/2012.
- [10] SADOU, Bachir; ALQUDAMI, Dheya Alhaq, *Etudes des techniques de modulation pour les réseaux mobiles 5G et 6G de nouvelle génération*, 2019-2020.
- [11] S.Sébastien, *Les Enjeux De LA 5G*, ARCEP, 2017.
- [12] Neal, D. Mackay and R., *Near Shannon Limit Performance of Low Density Parity Check Codes*, Electronics Letters, 2002.
- [13] 3GPP TS 38.212. Technical Specification Group Radio Access Network; NR; Multiplexing and channel coding (release 16).
- [14] J. H. Bae, A. Abotabl, H.-P. Lin, K.-B. Song, and J. Lee, *An overview of channel coding for 5G NR cellular communications*, *APSIPA Transactions on Signal and Information Processing*, vol. 8, p. e17, 2019. doi : 10.1017/ATSIP.2019.
- [15] 3GPP, Study on New Radio (NR) Access Technology Physical Layer Aspects. Technical Report TR-38.802, 2016
- [16] J-C ROLIN G Eiffel Dijon, 02/2008.

- [17] Benyarou. Mourad. Optimisation des systèmes multi-antennes appliqués aux systems.
- [18] Ph.Piret. convolutional codes : an algebraic approche. MIT Press, Cambrij, MA,1988
- [19] C. Berrou, A. Glavieux et P. Thitimajshima, *Near Shannon limit error correcting coding and decoding : turbo-codes*, IEEE ICC'93, vol. 2/3, pp.1064-1070, Geneva, May 1993.
- [20] S. Benedetto, G. Montorsi, Iterative decoding of serially concatenated convolutionel code, Electronics Letters, vol. 32, no 13, pp. 1186-1188, Juin 1996.
- [21] A. POLI, L. HUGUET, *Codes Correcteurs : Théorie et Applications*, Masson, Paris 1989.
- [22] S. FOUGHALI, S. KHELIFA, *Concaténation des Codes Cycliques (Reed Solomon - Hamming) appliquées aux images fixes*, Institut d'Informatique, USTO 1998.
- [23] Y. Kou, S. Lin, and M. Fossorier, *Low-Density Parity-check codes based on finite geometries : a rediscovery and new results*, IEEE Transactions on Information Theory, vol. 47, November 2001.
- [24] Kenneth s.Andrews, Sam Dolima, tth developement of turbo and LDPC code for deep space application.
- [25] R.Gallager, *LOW DENSITY PARITY CHECK*, RE Trams. nformation, Theory, pp.21-88, Jan.1962.
- [26] R.M Tanner, *Arecursive approach to low complexity codes*, EEETrans. nformationTheory, pp.533-547, sept 1981.
- [27] David Hayes, *FGPA implementation of a flexible LDPC decoder*, University of Newcastle Australia, A thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering in Télécommunication Engineering, 2008.
- [28] Mathieu Cunche, *Codes AL-FEC hautes performances pour les canaux à GRENOBLE*, France, thèse de doctorat, 2010.
- [29] Ayoub Otmani, Christophe Crespelle, *Etude dun code correcteur linéaire pour*, thèse pour obtenir le titre de Docteur en Sciences mention Informatique, Antoine Roux sous la direction de Michèle Soria et Laurent Frèrebeau, 2019.
- [30] C. Poulliat, *Contribution à létude et à loptimisation de systèmes à composantes itératives*, Habilitation à diriger des recherches, Université de Cergy-Pontoise, Sep. 2010.
- [31] T. Richardson, M. Shokrollahi, R. Urbanke, *Design of Capacity-Approching Irregular Low-Density Parity-Check Codes*, IEEE Trans. Inf. Theory, vol. 47, pp. 619-637
- [32] S. M. Jamalullil, *Transmission multiple porteuse utilisant un codage détecteur/correcteur derreur de type LDPC sur canaux MIMO*, Thèse de doctorat, Ecole Doctorale Science Technologie Santé, Université de Limoges, Mar. 2005.
- [33] F. Guilloud, Architecture générique de décodeur de codes LDPC, PhD. Dissertation, Télécom ParisTech, Paris, 2004.
- [34] Tram Thi Bao Nguyen, Tuy Nguyen Tan, Hanho Lee, *Efficient QC-LDPC Encoder for 5G New Radio*, 2019.
- [35] Li, J. ; Lin, S. ; Abdel-Ghaffar, K. ; Ryan, W. ; Costello, D.J., Jr. LDPC Code Designs, Constructions, and Unification ; Cambridge University Press : Cambridge, UK, 2017.
- [36] Vincent Pignoly. Étude de codes LDPC pour applications spatiales optiques et conception des décodeurs associés. Electronique. Université de Bordeaux, 2021. Français. NNT : 2021BORD0025.
- [37] Document 3GPP ChairmanNotes 3GPP TSG RAN WG1 Meeting 89, 3GPP, May 2017.
- [38] Document 3GPP ChairmanNotes 3GPP TSG RAN WG1 Meeting AH NR2, 3GPP, june 2017.

- [39] 3GPP : TS 38.212 NR ; Multiplexing and channel coding, 2018.
- [40] Document 3GPP ChairmanNotes 3GPP TSG RAN WG1 Meeting 88bis, 3GPP, Apr.2017.
- [41] Li, H. ; Bai, B. ; Mu, X. ; Zhang, J. ; Xu, H. Algebra-Assisted Construction of Quasi-Cyclic LDPC Codes for 5G New Radio. IEEE Access 2018, 6, 5022950244
- [42] Chen, T. ; Vakili, K. ; Member, S. ; Divsalar, D. ; Fellow, L. ; Wesel, R.D. ; Member, S. Protograph-Based Raptor-Like LDPC Codes. IEEE Trans. Commun. 2015, 63, 15221532.
- [43] LDPC Decoder Design Using Compensation Scheme of Group. Electronics 2021, 10, 2010.
- [44] T. J. Richardson and R. L. Urbanke, *Efficient encoding of low-density parity- check codes*, IEEE Transactions on Information Theory, Vol. 47, no. 2, pp. 638- 656, Feb. 2001
- [45] I. DIOP1, S .M FARSSI, M.CHAUMONT, O. KHOUMA, H. B DIOUF, Ecole Supérieure Polytechnique de l'Université Cheikh Anta Diop Dakar Sénégal, Université de Nîmes, F-30021 Nîmes Cedex 1, France, Université de Montpellier 2, UMR5506-LIRMM, F-34095 Montpellier Cedex 5, France, CNRS, UMR5506-LIRMM, F-34392 Montpellier Cedex 5, France
- [46] Fossorier, M.P., M. Mihaljevic, and H. Imai, Reduced complexity iterative decoding of low-density parity check codes based on belief propagation. IEEE Transactions on communications. 47(5),1999 p. 673680.
- [47] ADJUDEANU, Codes correcteurs d'erreurs LDPC structurés, QUÉBEC : FACULTE DES SCIENCES ET DE GENIE , 2010.
- [48] R. Gallager, *Low-density parity-check codes*, IRE Transactions on Information Theory, vol. 8, no. 1, pp. 2128, 1962.
- [49] Ao Li. Performances des codes correcteurs derreur LDPC appliqués au lien Fronthaul optique haut débit pour l'architecture C-RAN du réseau 5G : conception et implantation sur FPGA. Traitement du signal et de limage [eess.SP]. Université de Limoges, 2017. Français. fNNT : 2017LIMO0110ff. fftel-01744890f
- [50] W. Ryan, *Low Density Parity Check Codes Part ii - The Iterative Decoder*. [Online]. Available : <https://www.coursehero.com/le/5635883/LDPC-2>
- [51] V. Mannoni, *Optimisation des codes LDPC pour les communication multiporteuses*, Thèse de Doctorat, université de Reims Champagne-Ardenne, Juin 2004.
- [52] Hend A. Orabi , A. Zekry, G. Gomah Implementation for Two-Stage Hybrid Decoding for Low Density Parity Check (LDPC) Codes
- [53] Chung, S.-Y., G.D. Forney ; T.J. Richardson ; R. Urbanke, On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit. IEEE Communications letters. 5(2), 2001, p. 58-60.
- [54] Hamidullah, Yasir Ullah, Habib Ullah, Design and Simulation of Low Complex Minimum, July 2019

## Résumé

Au cours de ces dernières décennies, les théoriciens du codage ont cherché des codes pratiques capables de se rapprocher de la limite de Shannon. Parmi les familles de bons candidats, les codes LDPC, qui sont une classe de codes linéaires en blocs qui offrent des performances proches de la capacité sur un grand nombre de canaux de transmission tout en admettant des décodeurs implémentables. Dans ce mémoire, nous nous sommes intéressés dans un premier temps aux principaux paramètres des codes correcteurs d'erreurs. Nous avons présenté la problématique du décodage de canal et plus particulièrement le décodage optimal dans le cas des codes linéaires en blocs. Nous avons ensuite étudié les codes LDPC, notamment les principales notions liées à leurs représentations graphiques et matricielles ainsi que leurs algorithmes de décodage en exploitant leurs propriétés structurelles. Enfin, nous avons étudié, évalué et comparé les performances des algorithmes de décodages BF et MSA en termes de BER en fonction de SNR en montrant l'influence du nombre d'itérations de décodage et du rendement  $R$ .

## Abstract

Over the past few decades, coding theorists have been searching for practical codes capable of approaching the Shannon limit. Among the families of good candidates are the LDPC codes, which are a class of linear block codes that offer near-capacity performance over a large number of transmission channels while admitting implementable decoders. In this dissertation, we first focused on the main parameters of error-correcting codes. We have presented the problem of channel decoding and more particularly the optimal decoding in the case of linear block codes. We then studied the LDPC codes, including the main notions related to their graphical and matrix representations as well as their decoding algorithms by exploiting their structural properties. Finally, we studied, evaluated and compared the performances of the decoding algorithms (BF and MSA) in terms of BER versus SNR by showing the influence of the number of decoding iterations and of the efficiency  $R$ .