

TABLE DES MATIERES

DEDICACES	i
REMERCIEMENTS	ii
TABLE DES MATIERES	iii
LISTE DES FIGURES	v
INTRODUCTION GENERAL	1
CHAPITRE 1 : LES CODES CORRECTEURS D'ERREURS.	2
1.1 Introduction.	2
1.2 Chaîne de communication.	2
1.3 Codage de source.	3
1.4 Codage de canal.	3
1.5 Seconde Théorème de Shannon.	4
1.6 Canal AWGN.	5
1.7 Canal BSC.	5
1.8 Théorie des codes.	6
1.8.1 Distance minimale de Hamming	7
1.9 Matrice génératrice.	7
1.10 Matrice de parité.	7
1.11 Les codes blocs et les codes convolutifs.	9
1.11.1 Les codes convolutifs.	9
1.11.2 Les codes blocs linéaires.	11
1.11.2.1 Les différentes représentations d'un code en bloc linéaire.	11
1.12 Conclusion.	13
CHAPITRE 2 : SYSTHEME MIMO.	14
2.1 Système MIMO.	14
2.1.1 Introduction.	14
2.1.3 Capacité de canal MIMO.	18
2.1.4 Comparaisons des capacités et limites des systèmes MIMO entre les systèmes SISO et SIMO.	19
2.2 Systèmes MIMO massifs.	20
2.2.1 Introduction.	20
2.2.2 Caractéristiques d'un système MIMO massif.	21
2.2.3 Techniques de transmission TDD et FDD.	22

2.2.4 Estimation du canal.	23
2.2.5 Contamination des pilotes.	24
2.2.6 Beamforming.	25
2.2.7 Conclusion.	26
CHAPITRE 3 : ETAT DE L'ART.	27
3.1 Codage et décodage polaire.	27
3.1.1 Codes polaires irréguliers.	28
FIGURE3.1:Polarisation des codes polaires irrégulier (4, 2) le WER correspondant est $P_e[53]$	29
3.1.2 Codage/décodage CA-SCL.	29
3.1.3 Comparaisons entre code polaire et LDPC applique au MIMO massif.	33
3.2 Conclusion.	33
CHAPITRE 4 : ANALYSES ET INTERPRETATIONS DES RESULTATS.	34
4.1 Introduction.	34
4.2 LES CODES POLAIRES.	34
4.2.1 Notation.	34
4.2.2 La construction de Codes Polaires	34
4.2.3 Processus de codage	35
4.2.5 Décodage SCL.	44
4.1.10 Conclusion.	48
CONCLUSION GENERAL	50
REFERENCES BIBLIOGRAPHIQUES	51

LISTE DES FIGURES

FIGURE 1.1 : Utilisation des FEC dans une transmission.....	13
FIGURE 1.2 : Paradigme de Shannon.....	14
FIGURE 1.3 : Information mutuelle	15
FIGURE 1.4 : Canal AWGN	16
FIGURE 1.5 : Canal BSC.....	16
FIGURE 1.6 : Circuit d'un codeur convolutif C (2, 1, 3).....	20
FIGURE 1.7 : Exemple de treillis.....	20
FIGURE 1.8: Codage en bloc.....	21
FIGURE 1.9 : Codage de parité par produit matriciel.....	22
FIGURE 1.10 – Matrice de contrôle de parité.....	22
FIGURE 2.1: Système MIMO avec N_t antennes en émission et N_r antennes en réception....	24
FIGURE 2.2 : Principe de la technologie MIMO.....	25
FIGURE 2.3 : Principe du multiplexage spatial.....	26
FIGURE 2.4 : Principe du codage spatio-temporel.....	27
FIGURE 2.5 : système MIMO.....	28
FIGURE 2.6 : Variation de la capacité dans les systèmes MIMO, SISO et SIMO fonction de M antennes émettrices et N antennes réceptrices.	29
FIGURE 2.7 : Types de configurations et déploiement d'une station de base pour le MIMO massif.....	30
FIGURE 2.8 : Modèle MIMO massif - M antennes et K utilisateurs.	31
FIGURE 2.9 Les régions possible (M , K) dans les systèmes TDD et FDD, pour un intervalle de cohérence $T=200$ symboles.....	32
FIGURE 2.10: contaminations des pilotes.....	34
FIGURE 2.11 : Création de faisceaux directionnels en faisant varier la phase (retard) et l'amplitude de chaque émission d'antenne.....	35

FIGURE 3.1: Polarisation des codes polaires irrégulier (4, 2) le WER correspondant est $P_e[53]$...	38
FIGURE 3.2 : L'encodeur polaire a 8bits [55]	39
FIGURE 3.3 : Le décodeur polaire SCL a 4bits [55]	39
FIGURE 3.4 : Décodeur polaire parallèle à 2 composantes.[55]	40
FIGURE 4.1 : Matrice génératrice et graphe de codeur Polaire CP(2,2)	44
FIGURE 4.2 : Matrice génératrice et graphe de codeur Polaire CP(4,4).	44
FIGURE 4.3 : Factor graph d'un codeur de Code Polaire CP (8;4) non systématique.	45
FIGURE 4.4 : Factor graph d'un Code Polaire de taille $N = 8$	46
FIGURE 4.5 : codeur deux bits.....	48
FIGURE 4.6 : Décodeur a deux bits.....	49
FIGURE 4.7 – Processus de décodage représenté sous la forme d'un arbre binaire.....	51
FIGURE 4.8 : Décodage SCL d'un Code Polaire CP (4;4) avec une liste de taille $L = 2$	53
FIGURE 4.9 : Modèle de transmission mimo massif avec codage polaire.....	54
FIGURE 4.10 : Performances des codes polaires par rapport au code LDPC appliqué à un système MIMO MASSIF avec $N = 256$, $R = 0,5$ et une modulation QPSK.....	56
FIGURE 4.11 : Performances FER pour un système MIMO massif à codage polaire et LDPC par rapport au SNR avec $N = 256$, $R = 0.5$ et modulation QPSK.....	56

ACRONYME

ARQ (Automatic Repeat Request) : Demande de répétition systématique.

AWGN Additive White Gaussian Noise) : bruit blanc gaussien additif.

BEC (Binary Erase Channel) : canal d'effacement binaire.

BLAST (Bell Labslayers Space Time) : bell labslayers espace-temps.

BLER (Block Error Rate) : taux d'erreur binaire.

BSC (symmetric binary channel) : canal binaire symétrique.

CA-SCL (List Successive Cancellation aided by CRC) : décodage LSC avec CRC.

CC (channel coding) : codage de canal.

C-RAN (Cloud-Radio Access Network) : Réseau d'accès radio.

CRC (Cyclic Redundancy Check) : contrôle de redondance cyclique.

CS (source coding) : codage de source.

DC (channel decoding) : décodage canal.

DMC (discrete, memory-less channels) : canaux discrets et sans mémoire.

DS (source decoding) : décodage source.

ECC (Error correcting code) : Code de correction d'erreur.

FDD (Frequency Division Duplex) : duplex par division de fréquence.

FEC (forword Error correction) : forword Correction d'erreur.

FER (frame error binary) : erreur de trame binaire. .

FFT (Fast Fourier Transform) : transformé de Fourier rapide.

IFFT (inverse fast Fourier transform) : transformée de Fourier rapide inverse.

LDPC (Low Density Parity Check) : Code à faible densité.

LR (Likelihood Ratio) : Rapport de vraisemblance.

MIMO (Multi Input Multi Output) : plusieurs antennes en émission et en réception.

ML (Maximum Likelihood) : décodage optimal des codes correcteurs d'erreurs.

NSC : (Non Systematic convolutional) : code convolutif non systématique.

OFDM : (Orthogonal Frequency Division Multiplexing) multiplexage par répartition de fréquence orthogonale.

PW (polarization weight) : poids de polarisation.

QPSK (Quadrature Phase Shift Keying) : modulation par déplacement de phase en quadrature.

RF (Radio Frequency) : fréquence radio.

RSC : (Recursive Systematic Convolutional codes) : Codes convolutionnels récurrents et systématiques.

SB : (Base Station) : Station de base.

SC (Successive Cancellation) : décodage d'annulation successive des codes polaires.

SCL (List Successive Cancellation) : décodage d'annulation successive utilisant une liste.

SDM : (Spatial Division Multiplexing) : multiplexage par division spatiale.

SIMO (Single Input Multi Output) : une seule antenne en émission et plusieurs en réception.

SISO (Single Input Single Output) : entrée unique sortie unique.

SNR (Signal Noise Ratio) : rapport signal à bruit.

TDD (Time-Division Duplex) : Duplex par division du temps.

TEB (Binary Error Rate) : taux d'erreur binaire.

RESUME :

L'utilisation des codes polaires à un système MIMO (Multi-Input Multiple-output) massif va participer à l'amélioration des performances des techniques de transmission des données. En utilisant des codes polaires comme code correcteur d'erreur cela augmente les performances lié au décodage selon les algorithmes de décodage utilisé, L'algorithme de décodage CA-SCL ont était proposé dans ce mémoire et les résultats on était satisfaisante. Les performances intégrées de la technologie MIMO (massive multiple-input multiple-output) associé aux codes polaires, qui sont deux technologies clés pour les systèmes de transmission mobiles. Un algorithme plus simple et plus efficace, appelé CA-SCL est utilisé dans la construction polaire. Sur la base de la réciprocité des canaux de liaison montante et descendante en mode duplex à répartition dans le temps (TDD), les performances MIMO massif avec N canaux RF indépendants et une largeur de bande choisi sont étudiées dans le cadre de différentes propositions faites dans ce sens, il est démontré dans ce mémoire que des gains de performance significatifs peuvent être obtenus dans un système MIMO massif avec codage polaire en augmentant le nombre d'antenne ou le nombre d'itération.

ABSTRAT :

The use of polar codes in a massive MIMO (Multi-Input Multiple-output) system will help improve the performance of data transmission techniques. By using polar codes as an error correcting code it increases the performance related to the decoding according to the decoding algorithms used. The CA-SCL decoding algorithm was proposed in this paper and the results were satisfactory. The integrated performance of the MIMO (massive multiple-input multiple-output) technology associated with polar codes, which are two key technologies for mobile transmission systems. A simpler and more efficient algorithm called CA-SCL is used in polar construction. Based on the reciprocity of the uplink and downlink channels in Time Division Duplex (TDD) mode, massive MIMO performance with N independent RF channels and a chosen bandwidth are studied in different proposals made in this sense, it is demonstrated in this thesis that significant performance gains can be obtained in a massive MIMO system with polar coding by increasing the number of antennas or the number of iterations.

INTRODUCTION GENERAL

Depuis quelques décennies, on observe dans le domaine des télécommunications l'apparition de nouvelles technologies permettant de communiquer de manières différentes. Du transport initial de la voix entre deux usagers, on est arrivé à un partage de données qui peut revêtir de nombreuses formes : messages, textes, photos, vidéos, etc. Tout d'abord, les premiers réseaux informatiques ont permis l'échange de données numériques par le biais de câbles et de fibres optiques. Puis, les systèmes cellulaires de première génération purement analogiques ont fait leur apparition à la fin des années 1970. En 1991, une nouvelle norme internationale nommée GSM entièrement numérique correspondant à la deuxième génération de téléphonie mobile. Cette évolution mène ainsi vers un système cellulaire de troisième génération (3G) proposant des débits théoriques maximum de 2 Mbit/s. Des débits plus importants permettant de transférer simultanément des photos et des vidéos, ou de proposer les services des réseaux locaux ou internet font toujours l'objet d'une demande croissante d'où l'introduction de la quatrième génération (4G). Ce pendant pour répondre à l'ensemble des besoins des usagers, les opérateurs et acteurs du développement des réseaux n'ont de cesse d'accroître le très haut débit fixe et le très haut débit mobile, des activités de recherche, de normalisation et de développement sont en cours sur la prochaine génération mobile (5G). Pour atteindre les performances 5G le système MIMO massif constitue un candidat sérieux et pour assurer la qualité de service et combattre les imperfections de transmission et les pertes introduites par le désaccord entre le multiplexage et le démultiplexage, l'ajout d'un codage correcteur d'erreurs FEC est considéré comme nécessaire pour assurer la performance demandée et dans ce mémoire notre choix est porté sur les codes polaires. L'objectif de ce mémoire est d'étudier les codes polaires dans un système MIMO massif.

Le premier chapitre étudie les codes correcteurs d'erreurs dans sa globalité en expliquant les techniques de détection et de correction d'erreur.

Le second chapitre étudie le système MIMO classique en énumérant les systèmes SISO et SIMO en guise de comparaison avec le MIMO.

Le troisième chapitre nous verrons les travaux précédant autour de l'utilisation des codes polaires à un système MIMO massif.

Le dernier chapitre nous présentons les résultats des stimulations en les interprétant et en dégager des perspectives.

CHAPITRE 1 : LES CODES CORRECTEURS D'ERREURS.

1.1 Introduction.

Pour assurer la qualité de service en tenant compte de l'instabilité de l'amplification optique, un code correcteur d'erreurs dit FEC (forward Error correction) est proposé. Le code correcteur d'erreur est une technique pour contrôler et corriger les erreurs de transmission des données introduites par des canaux de propagation bruités. Un mathématicien américain Richard Hamming a proposé à partir des années 1940 à 1950[1] une idée importante pour le codage FEC qui consiste à rajouter au code source un message redondant en utilisant un ECC (Error correcting code) comme illustre la figure 1.1

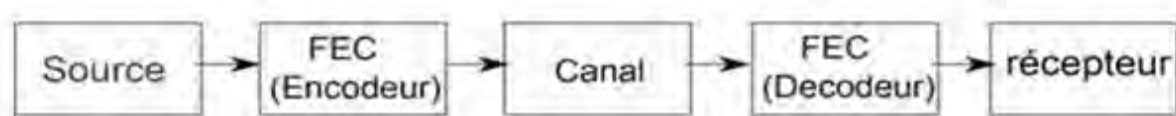


FIGURE 1.1 : Utilisation des FEC dans une transmission.

Les messages redondants ou les redondances se calculent en fonction des données (messages) ou des bits utiles, ce code permet de sécuriser les informations transmises mais également de détecter et certaines fois de corriger un nombre limité d'erreurs dans les messages reçus sans demander de retransmission de messages (mécanisme appelé ARQ : Automatic Repeat Request). Par exemple, pour un code C constitué de N bits, un message de ce code contient K bits d'informations utiles et $(N - K)$ bits de redondance. Le code est toujours noté $C(N, K)$ et sa performance dépend du nombre de bits de redondance mais aussi des règles (algorithmes) qui génèrent ces redondances. Dans le cas normal, plus il y a des bits de redondance, meilleures sont les performances que nous pouvons obtenir. En revanche, le taux de codage ou plutôt le rendement de codage qui est calculé par (K/N) , et qui correspond au rapport des bits d'information utiles sur le nombre total de bits utilisés pour coder un message, doit rester élevé dans le contexte C-RAN pour permettre d'obtenir des débits de transmission très élevés. En conséquence, nous devons faire non seulement un compromis entre le rendement de code et les performances, mais nous devons aussi améliorer l'efficacité des processus de codage et de décodage pour les différents algorithmes.

1.2 Chaîne de communication.

Il est important de connaître la capacité du canal avant la transmission. Dans l'article de Shannon paru en 1948 [1], Shannon donne le modèle simplifié d'une communication sans fil. Ce modèle est composé de 6 parties : S (source), CS (codage de source), CC (codage de

canal), DC (décodage canal), DS (décodage source), D (destination). La Figure 1.2 montre le modèle utilisé par Shannon pour la description de la communication. Evidemment, le FEC a un impact considérable sur la performance de la chaîne complète d'émission-réception.

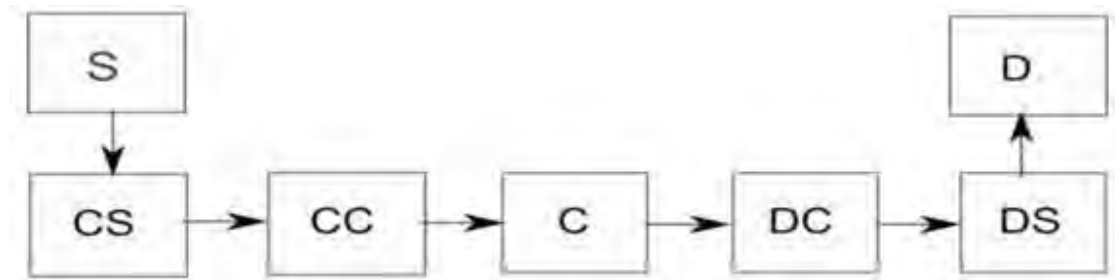


FIGURE 1.2 : Paradigme de Shannon.

1.3 Codage de source.

Des bits de redondance sont rajoutés avant que le message ne soit émis par la source S. Le codage de source a pour but d'éliminer la redondance de la source (compression). Il faut pour cela répartir équitablement l'information sur l'ensemble des symboles x_i . On définit alors l'entropie d'une source comme étant l'information moyenne émise par la source [2] :

$$H(X) = E \left[\log_2 \left(\frac{1}{p_i} \right) \right] = \sum_i p_i \log_2 \left(\frac{1}{p_i} \right) \quad (1.1)$$

p_i : probabilité d'émettre le i ème symbole. Pour une source binaire avec $p_i = x$, on peut alors déduire la fonction d'entropie binaire :

$$H_2(X) = x \log_2 \left(\frac{1}{x} \right) + (1 - x) \log_2 \left[\frac{1}{(1-x)} \right] \quad (1.2)$$

Si l'entropie de la source est inférieure à l'entropie maximale possible, alors la source est qualifiée de redondante et la taille (longueur) du message émis peut être réduite par un codeur de source. Concernant le codage Source, le premier théorème de Shannon indique qu'il existe un procédé de codage déchiffirable où la longueur moyenne des mots par symbole de la source est aussi voisine que l'on veut de sa borne inférieure $\frac{H}{\log_2(q)}$ où q est la taille de l'alphabet considéré.

1.4 Codage de canal.

Le codage de canal a pour but de protéger le message émis par la source normalisée (sans redondance) contre les bruits et les perturbations introduits par le canal de propagation. Pour assurer un transfert sur un canal bruité, il est nécessaire d'introduire de la redondance dans le message transmis. En réception, le décodeur reçoit le message émis par le codeur perturbé à

cause de bruit du canal. L'information mutuelle $I(X; Y)$ en Figure 1.3 entre l'entrée et la sortie du canal peut être exprimée sous la forme [2] :

$$I(X; Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \quad (1.3)$$

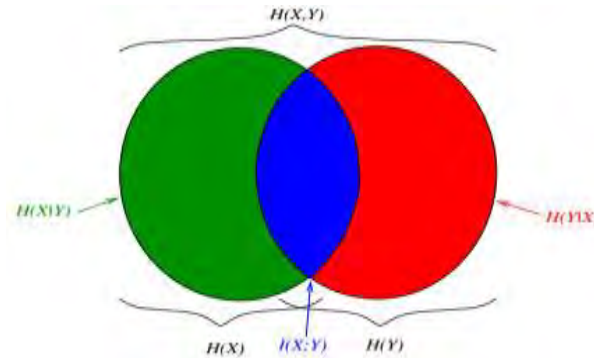


FIGURE 1.3 : Information mutuelle

Où $H(X)$ est l'entropie de la source et $H(X|Y)$ est le niveau d'incertitude sur la variable X connaissant Y .

1.5 Seconde Théorème de Shannon.

Le second théorème de Shannon apporte donc une solution au problème soulevé au niveau de la capacité du canal. Les codes qui peuvent satisfaire le théorème de Shannon avec un rendement R aussi près voulu de la capacité C du canal considéré sont appelés les codes qui atteignent la capacité. Il faut noter que la recherche d'un tel code est très difficile. En fait, à l'époque de Shannon (1948), aucun code de la sorte n'était connu. La preuve du théorème de Shannon n'est pas constructive, c'est-à-dire qu'elle ne donne pas de recette explicite pour la construction d'un code atteignant la capacité. Depuis 1948, les théoriciens des codes ont tenté sans succès de démontrer que certains codes pouvaient atteindre la capacité. Ce n'est qu'en 2009 avec l'introduction des codes polaires qu'on a pu avoir une famille de codes avec une preuve d'atteinte de la capacité et une complexité d'encodage et de décodage réaliste pour des applications pratiques [3]. Ce théorème peut être décomposé en trois parties [4] :

1. Pour tous les canaux discrets sans mémoire, la capacité du canal C a les propriétés suivantes. Pour tout $\varepsilon > 0$ et $R < C$, pour N assez grand, il existe un code de longueur N et de rendement R muni d'un algorithme de décodage tel que la probabilité d'erreur bloc maximale soit $< \varepsilon$.

$$C = \max_{p_X} I(X; Y) \quad (1.4)$$

2. Si une probabilité d'erreur binaire p_b est acceptable, le débit (p_b) peut être atteint, avec :

$$R_{p_b} = \frac{C}{1-H_2(p_b)} \quad (1.5)$$

3. Pour tout p_b , les débits plus grands que $R(p_b)$ ne peuvent pas être atteints. Selon le théorème de Shannon, on doit respecter ces critères pour atteindre le meilleur résultat pour un rendement R donné du code. Donc le canal est considéré comme un canal AWGN (Additive White Gaussian Noise). Nous avons aussi le canal BSC (canal binaire symétrique).

1.6 Canal AWGN.

Le canal AWGN possède une entrée et une sortie continue en amplitude (le canal reste discret en temps). La perturbation (bruit) introduite par le canal est l'addition d'un bruit blanc de densité spectrale de puissance constante et de distribution gaussienne en amplitude. Le signal reçu peut s'exprimer par : $Y = X + E$ où E suit une loi normale de moyenne nulle et de variance σ^2 (Figure 1.4). La perturbation ou le bruit est caractérisé par cette variance qu'on obtient à partir de la somme des variances des différents bruits rencontrés sur la transmission supposée tous gaussiens et indépendants (bruit d'émission spontanée, bruit thermique, bruit de battement entre le signal et le bruit, bruit de battement entre le bruit et lui-même...) [5].

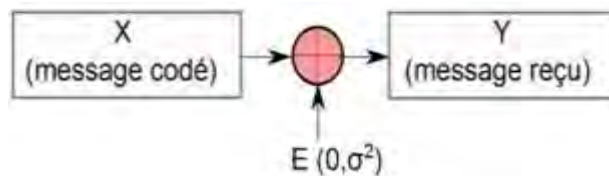


FIGURE 1.4 : Canal AWGN.

1.7 Canal BSC.

Le canal binaire symétrique (BSC : Binary Symmetric Channel) est illustré sur la Figure 1.5. Il possède 2 entrées et 2 sorties (0, 1). Il transmet chaque bit correctement avec une probabilité $1 - f$ et inverse l'entrée avec une probabilité

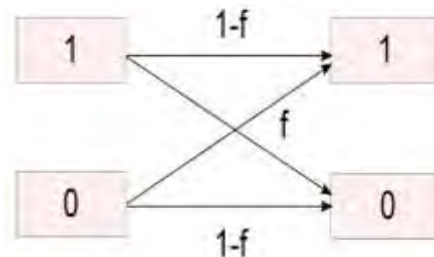


FIGURE 1.5 : Canal BSC.

1.8 Théorie des codes.

Dans l'article de Shannon 1948 [1], Shannon a prouvé qu'il était possible de transmettre un débit R de manière fiable sur un canal non-fiable de capacité C tant que $R < C$, par contre pour le cas inverse, on ne peut pas garantir la fiabilité du message reçu à travers le canal [6]. Dans ce papier, Shannon aussi introduit la notion de "codes" comme un ensemble fini de vecteurs dans l'alphabet d'entrée. Si on suppose que tous les vecteurs possèdent une longueur identique N et que N est égal à $\frac{K}{R}$ (qui peut être codé sur K bits), il faut alors N utilisations du canal pour transmettre K bits. Le débit (rendement du code) est donc égal à $\frac{K}{N}$ bits par utilisation du canal. On a tout simplement :

$$N = K + (1 - \frac{K}{N}) N = K + M \quad (1.6)$$

Avec K qui représente la taille en bits du message à coder et M qui représente la redondance apportée par le codage de canal. En réception, c'est le décodeur qui, à partir du vecteur y reçu et de l'alphabet de sortie, va en déduire le mot de code C envoyé par l'encodeur. Si le canal introduit des erreurs, alors on ne peut pas identifier le mot de code transmis de manière certaine. On peut cependant trouver le mot de code le plus probable en maximisant la probabilité $p(y|c)$. Le décodage alors effectué est dit décodage à Maximum de Vraisemblance, qui fonctionne en minimisant la distance entre le mot décodé et tous les mots de code dans l'alphabet. Shannon a prouvé qu'il existe des codes de rendement arbitrairement proche de la capacité pour lesquels la probabilité d'erreur du décodeur ML tend vers 0 quand la longueur du code tend vers l'infini. Ce théorème ne donne cependant pas la méthode permettant de construire ces codes. De plus, même si un code (aléatoire) atteignant la capacité pour un certain rendement était connu, la réalisation de l'encodage ainsi que du décodage par des algorithmes efficaces (polynomiaux) reste un problème ouvert. Depuis longtemps, les scientifiques ont cherché à trouver des codes qui atteignent ou s'approchent de la capacité de canal. Pour une efficacité de codage et de décodage, Elias et Golay introduisent le concept de code linéaire en définissant des codes comme un sous-espace vectoriel d'un espace vectoriel de dimension finie, c'est à dire, au lieu de stocker une liste de 2^K mots de code dans l'encodeur, on fait une simple multiplication matricielle [7]. Ils ont ainsi montré que la détection à Maximum de Vraisemblance (ML) pour un canal binaire symétrique (BSC) était un problème NP complet. Dans les années 1960, les codes RS [8] ont été inventés et ils peuvent être décodés par un algorithme qui est appelé algorithme de Berlekamp Massey. La complexité de ces algorithmes limite la longueur des codes (toujours fixée à $(255, K)$) à

quelques centaines de symboles entraînant un écart entre les performances des codes et la limite donnée par Shannon largement supérieur à 3 dB. En 1993, les turbo codes [9] sont présentés par C. Berrou et provoquent un émoi dans la communauté scientifique concernée puisqu'ils permettent de s'approcher très près (moins de 0.5 dB) de la borne de Shannon sur un canal AWGN tout en présentant une complexité de décodage raisonnable. Le décodage s'opère de façon itérative en échangeant à travers des entrelaceurs les informations extrinsèques issues des décodeurs de canal à sorties pondérées. Enfin en 1996, MacKay et Neal redécouvrent les codes LDPC qui avaient été inventés pour la première fois avant eux par Gallager en 1962, et généralisent l'emploi des graphes de Tanner pour modéliser les échanges d'informations extrinsèques entre nœuds de variable et nœuds de parité.

1.8.1 Distance minimale de Hamming

La distance minimale de Hamming notée d_{min} est définie comme la plus petite des distances entre deux mots de code distincts, constitue un paramètre très important dans la conception d'un code efficace. Par exemple un mot de code qui contient N bits codés à partir de K bits avec une redondance de M bits est noté (N, K, M) alors le pouvoir de détection du code, correspondant au nombre maximal d'erreurs détectables dans un mot dépend directement de la distance de Hamming minimale. Il est déterminé par $t = d_{min} - 1$ qui est capitale pour un bon code correcteur d'erreur. Le pouvoir de correction, défini comme le nombre maximal d'erreurs corrigibles dans un mot est donné par [10.69].

$$\left\lfloor \frac{t}{2} \right\rfloor = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor \quad (1.7)$$

1.9 Matrice génératrice.

Une matrice génératrice G d'un code $C(n, k)$ est une matrice de k lignes et n colonnes à éléments dans \mathbb{F}_2 (éléments binaires 0 et 1), dont les lignes constituent une base de C . Elle permet d'associer à un bloc de données $u = u_1^k$ un mot de code $c = c_1^n$ par la relation $c_1^n = u_1^k G$. Un code $C(n, k)$ admet plusieurs matrices génératrices. En effet, en permutant les lignes ou les colonnes d'une matrice génératrice ou encore en ajoutant, à une ligne, une ou plusieurs autres lignes, on obtient toujours une autre matrice génératrice du code.

1.10 Matrice de parité.

A chaque code en bloc linéaire $C(n, k)$ de matrice génératrice G , on peut associer un code en bloc linéaire dual $C(n, n - k)$, qui vérifie que tout mot du code dual est orthogonal à tout mot

du code $C(n, k)$. Le dual du code $C(n, k)$ est donc un sous-espace vectoriel de F_2^n constitué de 2^{n-k} mots de code de n symboles. Ce sous-espace vectoriel est l'orthogonal du sous-espace vectoriel constitué des 2^k mots du code $C(n, k)$. Il en résulte que tout mot c du code $C(n, k)$ est orthogonal aux lignes de la matrice génératrice H de son code dual $cH^T = 0$, pour tout mot de code c de $C(n, k)$ et $G \times H^T = 0$. T désigne la transposition. Ceci conduit à une propriété très importante :

Un mot $a = a_1^n$ donné est un mot de code si et seulement si

$$\mathbf{aH}^T = \mathbf{0} \quad (1.8)$$

Cette propriété peut être utilisée au décodage pour vérifier si le mot reçu est un mot de code et détecter ainsi la présence d'erreurs. C'est pourquoi la matrice H est appelée la matrice de contrôle de parité du code $C(n, k)$. Il se construit à partir de la matrice de parité (Parity Check matrix). La matrice de parité est définie de façon à satisfaire la relation :

$$G * H^T_{(N-K, N)} = 0 \text{ (modulo 2)} \quad (1.9)$$

Dans cette définition, $H^T_{(N-K, N)}$ correspond à la matrice transposée de la matrice de parité H de taille $(N - K, K)$. Si un mot de code s'exprime sous la forme :

$$C = \text{Message} * G \quad (1.10)$$

Alors en multipliant à droite l'égalité précédente par la transposée de H on peut obtenir :

$$C * H^T = \text{Message} * G * H^T = 0 \text{ (modulo 2)} \quad (1.11)$$

Si le produit $C * H^T \neq 0$, c'est qu'il existe des erreurs dans les bits reçus (il y a des chances qu'on ne puisse détecter aucune erreur alors qu'il y en a plusieurs erreurs dans la séquence mais que, malgré cela, le message apparaît bon dans l'équation de parité), il est important de noter que plus la distance entre deux mots de code est élevée, meilleurs la performance qu'on peut atteindre. Le résultat de $C * H^T$ s'appelle syndrome, les algorithmes de décodage des codes en bloc sont généralement basés sur le calcul du syndrome. Les décodages par syndrome les plus connus sont l'algorithme d'Euclide [11] ainsi que l'algorithme de Berlekamp Massey [12][13]. Le mot de code le plus proche est alors choisi afin d'annuler le syndrome.

1.11 Les codes blocs et les codes convolutifs.

Deux grandes familles de FEC (codes correcteurs d'erreurs) sont à distinguer : les codes convolutifs et les codes en blocs. Par définition, les codes convolutifs calculent la redondance de manière continue à mesure que le flot de données arrive alors que les codes en blocs génèrent la redondance par blocs de données. Quelle que soit la nature du code, il est dit systématique si l'on retrouve les K bits d'information utiles M dans le mot codé C . Cela se traduit dans la représentation matricielle par le fait qu'une partie de la matrice génératrice est composée de la matrice identité. Cette représentation est appelée la forme systématique de la matrice génératrice.

1.11.1 Les codes convolutifs.

Les codes convolutifs, aussi appelés codes convolutionnels ont été initialement inventés en 1954 par Elias [14]. Le principe des codes convolutifs est de considérer le message à émettre comme une séquence semi-infinie de symboles. Il consiste à passer le message utile à coder dans une succession de registre à décalage. Il y a 3 paramètres des codes convolutifs : K (bits informations utiles), B (le nombre de registres à décalage du code), N (bits codés après le codeur). On définit aussi la longueur de contrainte du code qui est égale à : $B + 1$. Pour un codeur convolutif on utilise le plus souvent un schéma de codage de rendement R . Parmi les codes convolutifs, il existe deux catégories : les codes non systématiques (NSC : Non Systematic convolutional), et les codes systématiques récurrents (ou RSC : Recursive Systematic Convolutional codes). Prenons l'exemple d'un code convolutif (2, 1,3) où 2 est le nombre de sorties, 1 nombre d'entrées et 3 nombre des registres. Le circuit d'un code C (2, 1, 3) est représenté sur la figure II.2. Ce codeur code des mots de 1 élément binaire en mots de 2 éléments binaires (il a donc un rendement $R = \frac{1}{2}$). A chaque instant t se trouvent dans le registre le dernier élément binaire $m(t)$ ainsi que les deux précédents $m(t - T_b)$ et $m(t - 2T_b)$, T_b désignant le temps bit. La sortie $c(t) = [c_1(t) \ c_2(t)]$ correspondant à l'entrée $B(t)$ vaut :

$$c_1(t) = B(t) + B(t - T_b) + B(t - 2T_b) \quad (1.12)$$

$$c_2(t) = B(t) + B(t - 2T_b) \quad (1.13)$$

Les additions étant modulo 2 [ou exclusif (1+1=0)].

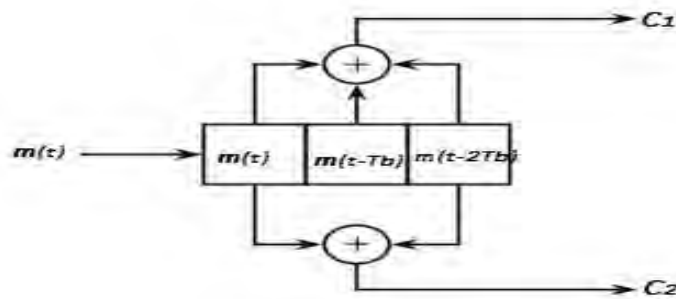


FIGURE 1.6 : Circuit d'un codeur convolutif C (2, 1, 3).

On note que Les performances de ces deux types de code sont très comparables, cependant l'architecture des codeurs RSC se prête mieux à une implantation de type turbo code. Même si les performances entre les codes NSC et les codes RSC sont très proches, il est montré dans [15] qu'un codeur récursif systématique présente de meilleures performances, cependant les différences sont minimales. Concernant le décodage des codes convolutifs (NSC ou RSC), on utilise l'algorithme de Viterbi [16] qui approxime au mieux le décodage par maximum de vraisemblance et qui utilise un diagramme en treillis comme celui illustré sur la Figure 1.9. Les états sont représentés par l'état des registres à décalage du codeur. Les états communiquent entre eux en fonction de leurs états courant et de la nouvelle entrée (0 ou 1) qui arrive. La suite des états parcourus tout au long du codage du message à transmettre associée à la suite des bits obtenus en sortie forme le treillis du code. L'algorithme de Viterbi va alors chercher parmi tous les chemins possibles à travers le treillis celui qui minimise la distance entre la séquence reçue et les différentes séquences reconstruites à travers le treillis. Le nombre d'états d'un codeur convolutif est égal à 2^B où B désigne le nombre de registres à décalage du code.

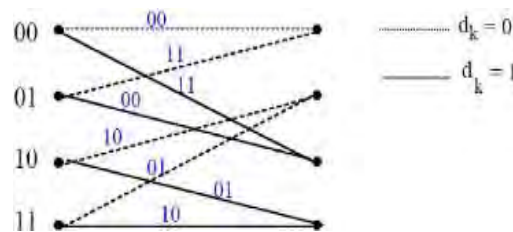


FIGURE 1.7 : Exemple de treillis.

1.11.2 Les codes blocs linéaires.

Le codage en bloc consiste à découper les informations en blocs de K bits avec un message codé en sortie qui contient N bits. Ainsi le message codé est souvent exprimé sous forme (N, K) , la redondance est donc égale à $N - K$ bits comme montre la Figure 1.10. Si on utilise la représentation matricielle, en appelant G la matrice génératrice du code, on obtient l'écriture :

$$\begin{array}{ccc}
 \text{message} & \text{génératrice de matrice : } G_{K \times N} & \text{mot de code : N bits} \\
 K \text{ bits} \begin{bmatrix} \vdots \end{bmatrix} & \times & \begin{bmatrix} g_{0,0} & \cdots & g_{0,N} \\ \vdots & \ddots & \vdots \\ g_{K,0} & \cdots & g_{K,N} \end{bmatrix} \\
 = \begin{bmatrix} \cdots \end{bmatrix} & &
 \end{array}$$

FIGURE 1.8 : Codage en bloc.

Cette écriture mathématique très simple montre bien qu'il est beaucoup plus facile d'ajuster le rendement d'un code en bloc comparé à celui d'un code convolutif. En fait, historiquement, on s'aperçoit que la plupart des codes correcteurs d'erreurs sont des codes linéaires. Ces codes linéaires admettent une représentation systématique, donc ce sont donc des codes systématiques. Un code systématique est aussi appelé un code polynomial. Parmi les codes blocs linéaire on peut citer les codes Reed Salamon, les codes LDPC et les codes polaires dans le développement de ce mémoire nous allons choisir les codes polaires comme code correcteur d'erreur.

1.11.2.1 Les différentes représentations d'un code en bloc linéaire.

Il est possible de représenter un code en bloc linéaire sous trois formes principales : forme matricielle, forme d'une matrice de parité ou bien forme d'un factor graph. L'application est linéaire car pour tout couple de messages $(u, v) \in (\mathcal{F}_2^K)^2$ et tout couple de scalaires $(\lambda, \mu) \in (\mathcal{F}_2)^2$, nous avons $\lambda u + \mu v \in \mathcal{F}_2^N$. Il est alors possible de définir la matrice correspondante à cette application linéaire, G de K lignes et N colonnes, et telle que le mot de code C du message u est calculé par :

$$C(u) = u \times G$$

EXEMPLE 1.1 Pour un code de parité de taille $N = 2$ nous avons :
 – le message de $K = 4$ bits est tel que $u = [u_0, u_1, u_2, u_3,]$

– le mot de code de $N = K + 1 = 5$ bits est tel que $C = [u_0; u_1; u_2; u_3; c_4]$,

– la matrice de l'application linéaire est $G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$

$$[u_0, u_1, u_2, u_3] \times \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} = H^T = \begin{bmatrix} u_0 & 0 & 0 & 0 \\ u_1 & 0 & 0 & 0 \\ u_2 & 0 & 0 & 0 \\ u_3 & 0 & 0 & 0 \\ u_0 & u_1 & u_2 & u_3 \end{bmatrix}$$

FIGURE 1.9 : Codage de parité par produit matriciel.

Le bit de parité est calculé tel que $c_K = \sum_{i=0}^{K-1} (u_i) \bmod 2$. En effet le produit matriciel dans la figure 1.6 permet de générer le mot de code. À partir de cette matrice génératrice, il est possible de définir une matrice de contrôle de parité, notée G , qui permet de définir les contraintes du code. Une utilisation d'une telle matrice est expliquée dans le paragraphe suivant.

Exemple 1.2 Pour un code de parité de $N = 5$ bits la matrice de contrôle de parité est définie par :

$$H = [1 \ 1 \ 1 \ 1 \ 1]$$

La figure 1.7 illustre le produit entre la matrice génératrice du code (G) et sa matrice de contrôle de parité (H).

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}^T = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

FIGURE 1.10 : Matrice de contrôle de parité.

EXEMPLE 1.3 :

Pour un code à répétition $M = 3$, supposons que nous voulions transmettre un seul bit u alors le mot de code généré est composé 3 fois du bit u . La matrice génératrice du code est alors égal à $G = (1 \ 1 \ 1)$.

La matrice de contrôle de parité est :

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

Supposons que nous voulions transmettre 1 avec un code à répétition $M = 3$ alors $C = [111]$. Si le message reçu $r = [101]$ alors son syndrome devient :

$$r \times H^T = [101] \times \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}^T = [10] \quad (1.14)$$

Le message reçu est alors erroné. Afin de corriger cette erreur, un vote à majorité est effectué. Il permet de prendre une décision sur la valeur la plus probable émise. Comme deux bits sont à 1, nous supposons que la valeur envoyée devrait être $[111]$. Ce vecteur permet d'annuler le syndrome. Le décodage est terminé. Le bit émis originellement est donc 1.

1.12 Conclusion.

Dans ce chapitre nous avons présenté le modèle classique d'une chaîne de communications numériques. Un message passe tout d'abord par le codage source, puis le codage canal, ensuite la modulation numérique avant d'être transmis à travers le canal de transmission. Lors de la réception ce sont les fonctions inverses qui sont appliquées afin de retrouver le message original. Ce mémoire se focalise sur les codes polaires comme code correcteur d'erreurs. Avant de définir en détail les codes polaires, les deux grandes familles de codes correcteurs d'erreurs ont été définies. Tout d'abord, le principe des codes convolutifs a été brièvement expliqué. Ensuite, les codes en blocs ont été présentés plus en détails avec des exemples de codes en blocs existants. Cela nous permet d'introduire des notions qui sont utilisées ensuite pour les Codes Polaires, une des familles des codes en blocs. Dans ce mémoire on propose un système d'utilisation des codes polaire pour un système MIMO massif donc il est nécessaire d'introduire les systèmes MIMO massif dans le chapitre qui suit.

CHAPITRE 2 : SYSTHEME MIMO.

Dans ce chapitre2, nous allons développer les systèmes MIMO classique et les systèmes MIMO massif. La technologie MIMO vient pour améliorer les difficultés rencontrées au niveau de la transmission de données à haut débit, qui a fait l'objet de plusieurs recherches sur les canaux spatio-temporels et leurs systèmes.

2.1 Système MIMO.

2.1.1 Introduction.

Dans les systèmes de communications traditionnels, il n'existe qu'une antenne à l'émission et une antenne à la réception. Ces systèmes sont appelés systèmes SISO (Single Input Single Output). Les systèmes SISO présentent des limites du goulot d'étranglement en termes de capacités de canal selon les critères de Shannon et Nyquist [PROA00]. Or les futurs services de communications mobiles sans fil demandent plus de transmissions de données (augmentation de la capacité de transmission). Alors pour améliorer la capacité des systèmes SISO et satisfaire ces demandes, les bandes passantes de ces systèmes et les puissances à transmettre ont été largement augmentées. Mais les récents développements ont montré que l'utilisation de N_t antennes à l'émission et N_r antennes à la réception (Fig 2.1) permettait d'augmenter le débit de transmission des données et cela sans augmenter ni la bande passante de l'antenne réceptrice du système SISO, ni la puissance du signal à l'émission. Cette opération s'effectue en divisant le canal en plusieurs "canaux spatiaux" par lequel des flux de données indépendants peuvent être transmis. Cette technique est connue sous le nom de "spatial multiplexage". En outre, un système MIMO peut être utilisé pour augmenter le gain de diversité, où le taux d'erreur sur les bits (TEB) en fonction de la courbe du rapport signal/bruit. Le ratio (SNR) peut être considérablement augmenté, jusqu'à un maximum de $N_t \times N_r \times L$ sur un système SISO, où L est le nombre de composants multi trajets efficaces et indépendants. Cette augmentation du gain de diversité se traduit par une augmentation de la portée à un TEB donné ou augmentation des données à une fourchette donnée jusqu'au taux maximum du système SISO d'origine.

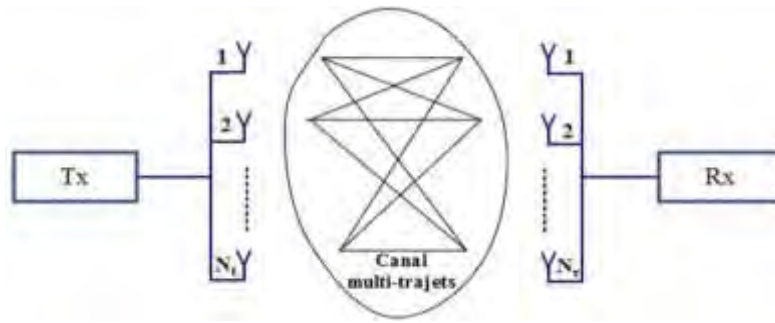


FIGURE 2.1: Système MIMO avec N_t antennes en émission et N_r antennes en réception.

Le principe de la technologie MIMO consiste à émettre dans un même canal des signaux transmis sur des antennes différentes. A la réception aussi et avec un certain nombre d'antennes et des traitements adéquats, il s'agit de simuler cette réception dans une même bande de n canaux différents [27-29].

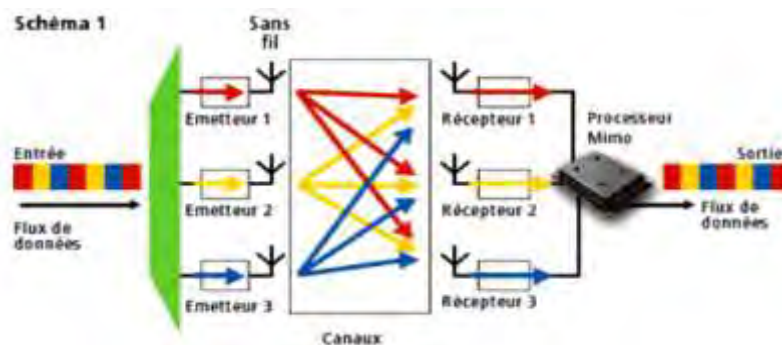


FIGURE 2.2 : Principe de la technologie MIMO.

Le flux est divisé en différents flux de même fréquence qui sont envoyés à travers 3 émetteurs à 3 récepteurs. L'algorithme permet ensuite d'identifier les différents flux en vue de les restituer en un seul flux. Cet algorithme utilise la réflexion des signaux sur les murs, le sol, etc. Alors que ces réflexions pourraient être considérées comme mauvaises, la technologie MIMO profite de ces différents canaux pour améliorer la rapidité de

transmission des données. On peut considérer trois catégories de MIMO :

- **la diversité spatiale MIMO** : on transmet simultanément un même message sur différentes antennes à l'émission. Les signaux reçus sur chacune des antennes de réception sont ensuite remis en phase et sommés de façon cohérente. Cela permet d'augmenter le ratio S/B (grâce au gain de diversité) de la transmission. Pour que cette technique soit efficace, il faut que les sous canaux MIMO soient décorrélés les uns des autres [30].
- **le multiplexage spatial MIMO** : chaque message est découpé en sous messages. On

transmet simultanément les sous-messages différents sur chacune des antennes d'émission. Les signaux reçus sur les antennes de réception sont réassemblés pour reformer le message entier d'origine. Comme pour la diversité MIMO, les sous-canaux de propagation doivent être décorrélés. Le multiplexage MIMO permet d'augmenter les débits de transmission (grâce au gain de multiplexage). Les techniques de diversité et de multiplexage MIMO peuvent être conjointement appliquées. Par exemple pour un système MIMO 5 x 5 (c'est-à-dire 5 antennes d'émission et 5 antennes de réception), on peut configurer un sous-système MIMO 2 x 3 pour faire du multiplexage et un sous-système MIMO 3 x 3 pour faire de la diversité MIMO [29.3][31.10].

• **le MIMO - Beamforming (formation de faisceau) :** le réseau d'antennes MIMO est utilisé pour orienter et contrôler le faisceau d'onde radio (amplitude et phase du faisceau). On peut ainsi créer des lobes constructifs / destructifs et optimiser une transmission entre l'émetteur et la cible. Les techniques de beamforming permettent à la fois d'étendre une couverture radio (d'une station de base ou d'un point d'accès par exemple) et de limiter les interférences entre utilisateurs et la pollution électromagnétique environnante (en ciblant le récepteur visé).

2.1.2 Les différents types de codage MIMO.

Dans un système MIMO, il existe plusieurs méthodes pour effectuer le codage des signaux :

- Le multiplexage par répartition de fréquence orthogonale (OFDM Orthogonal Frequency Division Multiplexing).

Consiste à distribuer sur un grand nombre de porteuses, le signal numérique que nous souhaitons transmettre. Comme si nous combinons le signal à transmettre sur des émetteurs indépendants et à des fréquences différentes. Pour que les fréquences des sous-porteuses soient les plus proches possibles et ainsi transmettre le maximum d'information sur une portion de fréquences donnée, l'OFDM utilise des sous-porteuses orthogonales entre elles. Les signaux des différentes sous-porteuses se chevauchent mais grâce à l'orthogonalité n'interfèrent pas entre eux. Ainsi, dans un environnement multi-trajets ou certaines fréquences sont détruites à cause des perturbations, le système sera tout de même capable de récupérer l'information perdue sur d'autres fréquences porteuses qui elles n'auront pas été détruites [WIK].

- Le multiplexage par division spatiale (SDM Spatial Division Multiplexing) ou multiplexage spatial.

Est une technique utilisée pour améliorer le débit de transmission de données. Il consiste à multiplexer simultanément plusieurs flux de données indépendants (essentiellement des canaux virtuels) dans un canal spectral. Chaque flux spatial doit disposer de sa propre paire d'antennes de transmission/réception à chaque extrémité du lien radio. Le multiplexage spatial consiste à émettre simultanément N_t symboles de modulation sur N_t antennes d'émission sans codage espace-temps de ces symboles. Ainsi, il améliore la capacité du canal avec le nombre d'antenne en émission et en réception [FOGA98]. Dans un système MIMO idéal, le maximum de canaux parallèles que l'on peut avoir est égal à $\min(N_t, N_r)$. Puisque les systèmes de communications sans fil exigent des débits de transmission très élevés et un faible taux d'erreur binaire, le BLAST (Bell Labslayers Space Time) est une technique de multiplexage utilisée dans les systèmes de communications sans fil et qui permet de transmettre plusieurs symboles en même temps avec la même fréquence. Il est important de noter qu'une chaîne de radio- fréquences RF et qu'un convertisseur analogique-numérique distincts sont nécessaires pour chaque antenne du système MIMO. Les configurations qui nécessitent plus de deux chaînes d'antennes RF doivent être conçues avec attention pour maintenir des coûts peu élevés tout en répondant aux attentes en matière de performances [INT]. Le principe de la technique de multiplexage spatial est présenté dans la figure 2.3.

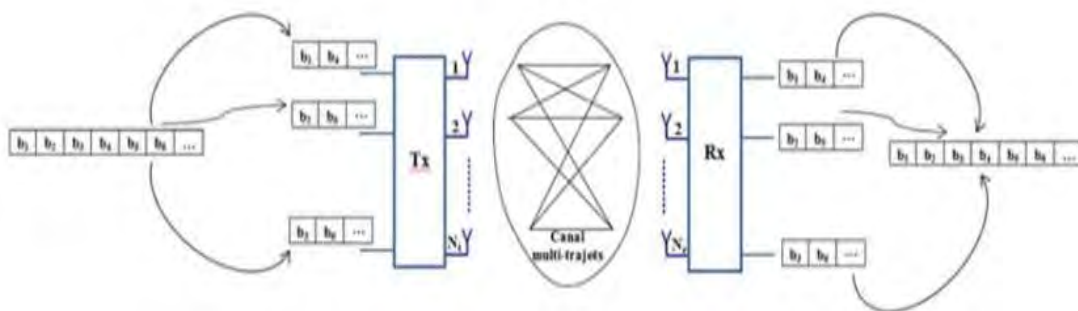


FIGURE 2.3 : Principe du multiplexage spatial.

-Le codage spatio-temporel.

Tout comme le SDM, permet d'envoyer des signaux différents sur chaque antenne. Le codage spatio-temporel consiste à introduire une corrélation spatiale et temporelle entre les signaux qui sont transmis par les différentes antennes de manière intelligente pour augmenter le gain de diversité et fournir un codage afin de minimiser les erreurs en réception [thèse I.DIOUM]. Cette technique est conçue spécialement pour l'utilisation d'antennes multiples en émission. Il permet d'améliorer la performance du système par l'exploitation des antennes multiples pour le gain de diversité plutôt que pour le gain de multiplexage spatial des flux de données

parallèles [CHIA06]. Le principe du codage spatio-temporel est d'introduire de la redondance aux données binaires émises afin d'augmenter la diversité spatiale et éviter les évanouissements propres au canal radio. La différence avec le multiplexage spatial est que le codage spatio-temporel prend un seul flux de données pour le coder dans le temps et dans l'espace pour produire des flux de symboles pour chaque antenne d'émission. Le codage spatial est utilisable dans le cas où le canal n'est pas connu à l'émission. Le principe de la technique de codage spatio-temporel est présenté dans la figure 2.4. Il est à noter que ce type de codage n'améliore pas la capacité de transmission linéairement avec le nombre d'éléments utilisés. Ainsi pour améliorer à la fois la capacité et la qualité, un système MIMO doit être implémenté avec les deux types de codages à savoir le multiplexage par division spatiale (SDM) ou multiplexage spatial et le codage spatio-temporel [PLI04 thèse Aliou diallo].

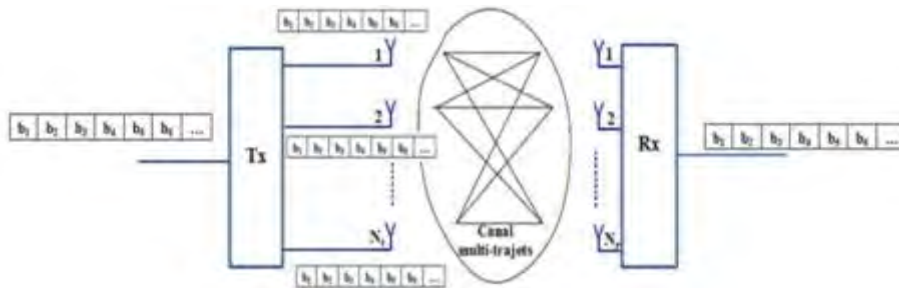


FIGURE 2.4 : Principe du codage spatio-temporel.

2.1.3 Capacité de canal MIMO.

Pour un canal MIMO (Figure 2.5), avec M antennes émettrices et N antennes réceptrices :

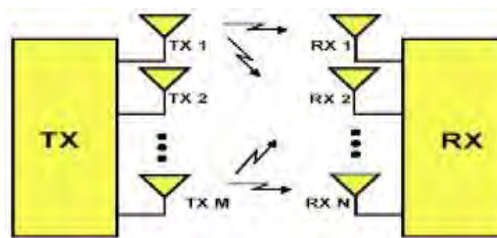


FIGURE 2.5 : système MIMO.

Dans ce cas le canal MIMO est traduit par une matrice H de taille N×M dite matrice de canal

$$H = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1M} \\ h_{21} & \cdot & \cdot & h_{2M} \\ \vdots & \cdot & \ddots & \vdots \\ h_{N1} & h_{N2} & \cdots & h_{NM} \end{bmatrix}$$

h_{ij} est le gain complexe du canal entre la j-ème antenne émettrice et la i-ème antenne réceptrice.

$$C = \log_2 \left(\det \left[I_N + \frac{\varphi}{M} H H^H \right] \right) \quad (2.1)$$

Lorsque N et M sont grands, l'espérance de la capacité pour un canal de Rayleigh croît proportionnellement à N :

$$E[C] \approx N \log_2(1 + \varphi) \quad (2.2)$$

La capacité augmente donc beaucoup plus vite que dans le cas SISO grâce au grand nombre d'antennes multiple.

2.1.4 Comparaisons des capacités et limites des systèmes MIMO entre les systèmes SISO et SIMO.

– Dans le cas SISO ($M = 1$ et $N = 1$) la capacité varie de 1 à 7,5 bps/Hz. Elle reste faible et croît lentement avec le SNR, ce qui illustre bien les limitations des transmissions SISO. Malgré les techniques qui permettent de maximiser l'utilisation d'un canal SISO, sa capacité demeure limitée et un système multi-antennes, même sous-exploité, on obtiendra de meilleures performances.

– Les deux exemples SIMO ($M = 1$, $N = 3$ et $N = 7$) montrent les limites supérieures des traitements. Le passage à trois antennes en réception permet de gagner 2 bps/Hz par rapport au SISO, en particulier à fort SNR. Avec $N = 7$ le gain est d'environ 1,5 bps/Hz, ce qui est peu pour quatre antennes supplémentaires. Comme pour les systèmes SISO, la capacité augmente lentement ce qui reste la principale limitation des systèmes SIMO.

– Les deux exemples MIMO considérés ont le même nombre total d'antennes que les systèmes SIMO, de façon à faciliter les comparaisons ($M + N = 4$ puis 8). Pour un SNR de 0 dB, le système MIMO avec $M = 2$ et $N = 2$ a une capacité équivalente à celle du système SIMO avec quatre antennes. La capacité MIMO augmente ensuite beaucoup plus rapidement avec le SNR, pour finir avec un gain de plus de 50% à 25 dB de SNR. Nous pouvons aussi vérifier que la capacité du système MIMO à huit antennes est presque le double de celle du système à quatre antennes. [Thèse aliou thiane]

La figure 2.6 représente les courbes de la capacité en fonction du SNR pour plusieurs valeurs de N et M ; on n'a obtenu ces courbes sous MATLAB.

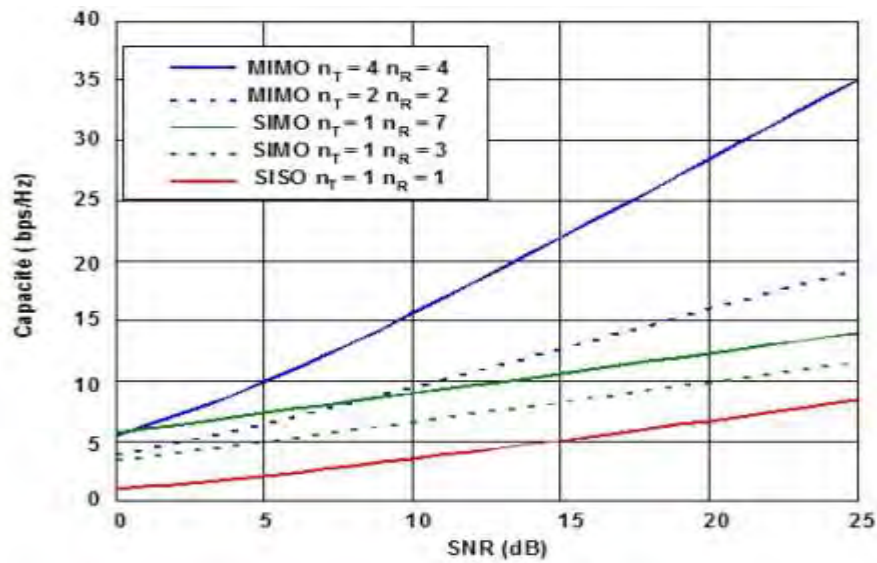


FIGURE 2.6 : Variation de la capacité dans les systèmes MIMO, SISO et SIMO fonction de M antennes émettrices et N antennes réceptrices.

Le système MIMO présente de nombreux avantages comme montré précédemment, mais il présente de nombreuses limites. Parmi ces limites, on peut citer le problème de la diversité de réception qui est faite d'envoyer le signal en parallèle sur $M \cdot N$ sous-canaux avec une superposition des ondes au récepteur créant des interférences, l'encombrement, l'espacement, la topologie des antennes. Malgré toutes ces limites, il est nécessaire d'améliorer les systèmes MIMO, d'où l'introduction des systèmes MIMO massifs.

2.2 Systèmes MIMO massifs.

2.2.1 Introduction.

MIMO massif ou Massive MIMO est un système MIMO à multi-utilisateurs où les stations de bases et le nombre d'utilisateurs sont très nombreux. Dans un système MIMO massif, un très grand nombre d'utilisateurs peuvent être connectés de manière simultanée et dans la même fréquence. Les systèmes MIMO massifs sont des catalyseurs pour le développement des réseaux à large bande (fixes et mobiles), qui sera efficace sur le plan énergétique, robuste, et utilisera le spectre de manière efficace. De nombreuses configurations et scénarios de déploiement différents peuvent être envisagés dans la figure 2.7.

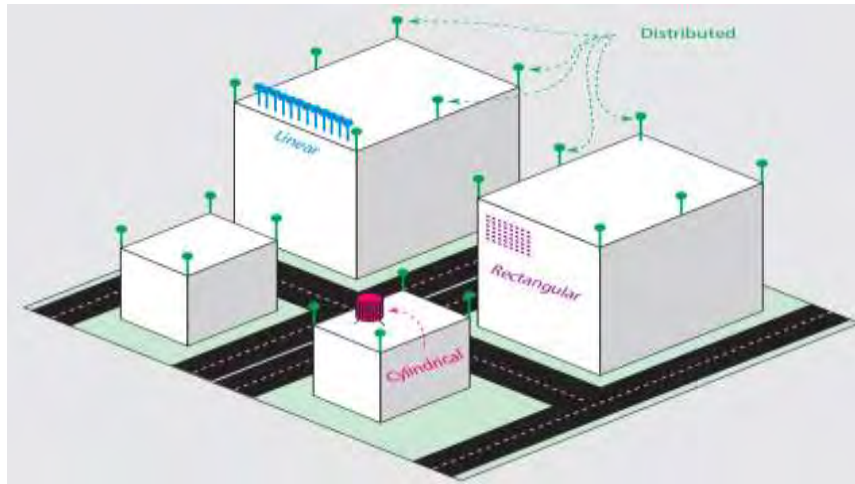


FIGURE 2.7 : Types de configurations et déploiement d'une station de base pour le MIMO massif.

Au niveau de la liaison montante, il est plus facile d'estimer le canal quand les terminaux envoient des signaux à la station de base tandis qu'il est plus complexe au niveau de la liaison descendante. Tout d'abord la station de base envoie des signaux aux terminaux, puis estime le canal pour pouvoir le quantifier. Enfin, l'estimation obtenue est renvoyée à la station de base. Ce qui est impossible pour le MIMO massif pour deux raisons. Premièrement, le signal au niveau de la liaison descendante doit être orthogonal c'est-à-dire tant que le nombre d'antennes augmente, plus de fréquence et de temps sont utilisés par rapport au MIMO classique [32.2]. Deuxièmement, vu que chaque terminal accepte plusieurs signaux, l'estimation de canal augmente alors par rapport au nombre d'antennes à l'émission. La solution à ce problème est donc d'utiliser la technique Duplex par séparation temporelle (ou *Time-Division Duplex*, TDD) et de dépendre à la réciprocité des liaisons montantes et descendantes du canal [33], donc il est nécessaire d'avoir une bonne connaissance du canal, au niveau de la liaison montante et descendante.

2.2.2 Caractéristiques d'un système MIMO massif.

Le MIMO massif est un système qui s'appuie sur l'exploitation de la dimension spatiale, créant ainsi plusieurs degrés de liberté et stimulant le principe de la transmission simultanée par multiplexage spatial. Le principe est de servir différents utilisateurs dans la même ressource temps-fréquence peut être considéré comme un élément-clé pour la 5G de réseaux mobiles en raison de l'épuisement du spectre de fréquence et pour répondre aux contraintes d'efficacité spectrale. Cette technologie consiste à équiper une station de base par un nombre

élevé M d'antennes pour servir un nombre d'utilisateurs K sous la supposition $M \gg K$. Le modèle du système MIMO massif est illustré à la figure 2.8.

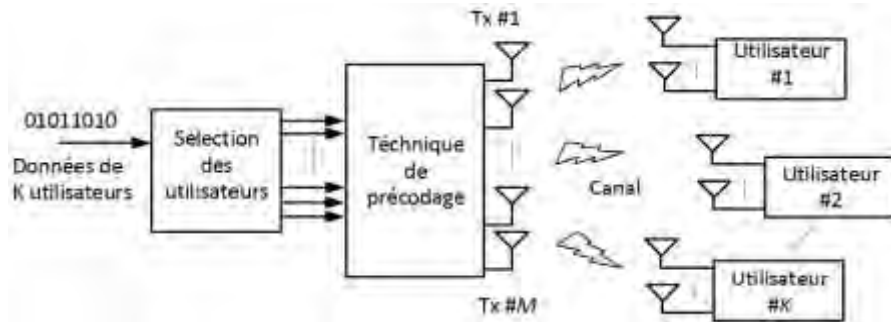


FIGURE 2.8 : Modèle MIMO massif - M antennes et K utilisateurs.

2.2.3 Techniques de transmission TDD et FDD.

Le MIMO massif en principe utilise le duplexage par répartition dans le temps TDD (Time Division Duplexing) dans lequel les canaux montants et descendants se partagent dans le temps la même ressource en fréquences. Les canaux physiques de propagation montants et descendants sont dites réciproques : il suffit d'évaluer le sens montant pour permettre à la station de base d'élaborer les paramètres à appliquer pour le précodage du signal descendant et à la combinaison des signaux reçus pour extraire le signal émis par chaque terminal. Pour ce faire, les terminaux émettent périodiquement une séquence de pilotes : ces K séquences sont en principe orthogonales. Pour l'utilisation du mode duplexage par répartition de fréquence FDD, où les sens montants et descendants utilisent des bandes de fréquences distinctes au système MIMO massif, il faut effectuer l'évaluation des canaux radio à la fois dans le sens montant et dans le sens descendant : les résultats des mesures effectuées par les terminaux doivent être retournés à la BS. On montre dans [34] que le domaine d'efficacité du MIMO massif se réduit car on note des limites qui sont du à de faible mobilité et de basse fréquence. Mais la limitation du MIMO massif au mode de duplexage TDD peut constituer une contrainte à son usage dans la mesure où beaucoup de réseaux est fondée sur le FDD. Pour le moment en réserve de certaines hypothèses la contrainte propre au FDD pourrait être levée, mais on ne sait pas si ces hypothèses sont réalistes. On en retiendra que l'utilisation de MIMO massif reste pour l'heure liée au mode TDD. Par exemple, si l'intervalle de cohérence est $T_c = 200$ symboles (correspondant au bande passante de cohérence de 200 kHz et de temps de cohérence de 1ms). Alors, en utilisant le système FDD, le nombre d'antenne à la station de base et le nombre d'utilisateur sont limité par $M+K < 200$, tandis que pour le système TDD, la limite de M et de K est $2K < 200$. La figure 2.8 montre les régions possibles (M, K) en système

FDD et TDD. On peut constater que la région FDD est plus petite que celle de la TDD. L'ajout d'antennes de plus n'affecte donc pas les ressources nécessaires pour l'estimation du canal, pour le système TDD.

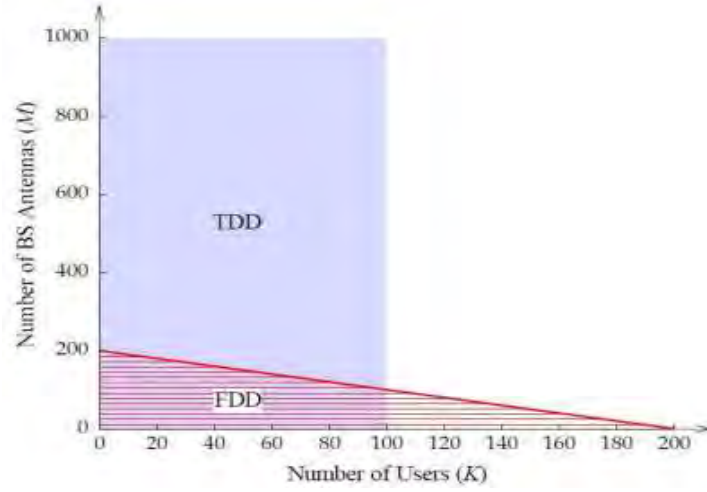


FIGURE 2.9 Les régions possible (M, K) dans les systèmes TDD et FDD, pour un intervalle de cohérence $T=200$ symboles.

2.2.4 Estimation du canal.

La connaissance du canal est capitale pour les systèmes MIMO massif, permettant d'améliorer considérablement les échanges. Cette information est très importante car elle permet d'effectuer les opérations de précodage dont le rôle est de former des faisceaux focalisant l'énergie vers, ou autour, des utilisateurs de la cellule. Une mauvaise estimation va donc provoquer une dégradation non négligeable de la capacité et de l'énergie d'un tel système. On retrouve dans [35] une étude analytique de l'impact d'une mauvaise estimation sur la capacité et l'efficacité énergétique du système.

Estimation de canal dépend aussi du temps de cohérence T_C , qui permet de connaître la durée pendant laquelle la réponse du canal ne varie pas de façon significative. Il faut donc estimer le canal avec une fréquence d'au moins $\frac{1}{T_C}$. En d'autres termes, on peut dire aussi que plus le nombre d'utilisateur augment, plus l'estimation de canal devra être réalisé souvent. On parle de *Channel Aging* pour exprimer la nature instable du canal [36] et aussi on peut remarquer que le temps alloué à cette estimation T_{est} croit proportionnellement avec le nombre K d'utilisateurs. L'estimation du canal peut être faite au niveau des terminaux utilisateurs qui remontent ensuite les informations de leurs canaux vers la BS massive ou alors directement au niveau de la BS massive, à travers de l'envoi de symboles pilotes des utilisateurs vers la BS massive.

2.2.5 Contamination des pilotes.

Dans l'idéal, chaque terminal d'un système MIMO massif se voit attribuer une séquence pilote de liaison montante orthogonale. Cependant, la gamme maximale de séquences pilotes orthogonales qui peuvent exister est limitée par la longueur de l'intervalle de cohérence divisée par l'étalement du retard du canal. Conformément à [37], pour un scénario de travail typique, la quantité maximale de séquences pilotes orthogonales dans un intervalle de cohérence d'une milliseconde est estimée à environ 200. Il est pratique d'épuiser les ressources accessibles des séquences pilotes orthogonales dans un système multicellulaire. L'impact de la réutilisation des pilotes d'une cellule à l'autre, et les conséquences négatives qui en découlent, est appelé "contamination des pilotes". Plus précisément, lorsque le réseau de services met en corrélation son signal pilote acquis avec la séquence pilote liée à un terminal particulier, il obtient en effet une estimation de canal qui est contaminée par un mélange linéaire de canaux vers les différents terminaux qui partagent la même séquence pilote. La formation de faisceau sur la liaison descendante, principalement basée sur l'estimation du canal contaminé, génère des interférences qui sont dirigées vers les terminaux qui partagent la même séquence pilote. Une interférence similaire correspond aux transmissions sur la liaison montante. Cette interférence attentive augmente avec le nombre d'antennes à un taux égal en tant que signal [37]. Même en partie, les séquences pilotes corrélées entraînent une interférence dirigée. La contamination des pilotes en tant que phénomène fondamental n'est pas particulière à la MIMO massive. Cependant, son impact sur le MIMO massif semble être beaucoup plus profond que dans le MIMO traditionnel [38], [37]. Selon [37], on a toujours soutenu que la contamination du pilote constitue une limite de performance restante, lorsque le nombre d'antennes est élevé à côté de la liaison, au moins avec les récepteurs qui dépendent de l'estimation du canal pilote. Bien que cet argument ait été contesté récemment [39], au moins sous certaines hypothèses particulières sur le contrôle de puissance utilisé, il semble possible que la contamination des pilotes doive être traitée d'une manière ou d'une autre. Certaines approches utilisées pour atténuer la contamination des pilotes sont :

-Distributions des pilotes : L'un des moyens consiste à utiliser un facteur de réutilisation des fréquences moins agressif pour les pilotes. Cette technique permet d'éloigner davantage les cellules contaminées. Il est en outre possible de coordonner l'utilisation des pilotes ou d'attribuer de manière adaptative des séquences de pilotes aux différents terminaux du système [40].

-Estimation intelligente : par exemple, les algorithmes d'estimation aveugle [39] qui évitent complètement l'utilisation de pilotes [41] peuvent également atténuer ou supprimer les conséquences de la contamination des pilotes.

-Structure du système : De nouvelles stratégies de pré codage qui tiennent compte de la structure du système, comme le pré codage de la contamination du pilote [42], peuvent exploiter la transmission coopérative sur une multiplicité de cellules en dehors de l'opération de formation de faisceau pour annuler, au moins partiellement, l'interférence dirigée qui résulte de la contamination du pilote. Contrairement à la formation de faisceau coordonnée sur quelques cellules qui nécessite des estimations des canaux appropriés entre les terminaux de l'utilisateur et la station de base des cellules contaminées, le pré codage de la contamination pilote repose sur les coefficients d'évanouissement.

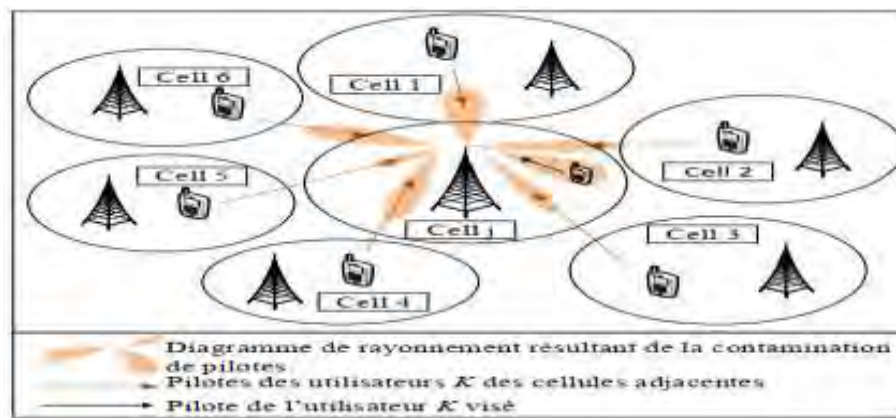


FIGURE 2.10: contaminations des pilotes.

2.2.6 Beamforming.

Le "beamforming" est utilisé en système MIMO pour une transmission guidée. On peut dire en général que la formation de faisceau utilise plusieurs antennes pour contrôler la direction d'un front d'onde en pondérant de manière appropriée l'amplitude et la phase des signaux d'antenne individuels dans un réseau d'antennes multiples. Autrement dit, le même signal est envoyé par plusieurs antennes qui sont suffisamment espacées les unes des autres. Dans un endroit donné, le récepteur recevra donc plusieurs copies du même signal. Selon l'emplacement du récepteur, les signaux peuvent être en opposition de phase, en faisant la moyenne destructive des uns et des autres, ou se résumer de manière constructive si les différentes copies sont dans la même phase, ou tout ce qui se trouve entre les deux.

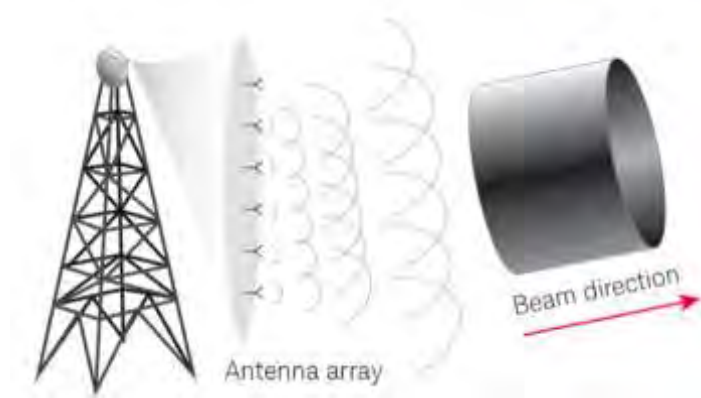


FIGURE 2.11 : Création de faisceaux directionnels en faisant varier la phase (retard) et l'amplitude de chaque émission d'antenne.

Le signal est pré-codé (modifications d'amplitude et de phase) dans le traitement en bande de base avant la transmission RF. Plusieurs faisceaux (un par utilisateur) peuvent être formés simultanément à partir du même ensemble d'éléments d'antenne. La formation numérique de faisceaux améliore la capacité de la cellule car les mêmes peuvent être utilisées pour transmettre des données simultanément pour plusieurs utilisateurs.

2.2.7 Conclusion.

Dans ce chapitre, nous avons étudié les caractéristiques fondamentales des systèmes MIMO en faisant l'historique des systèmes en partant des systèmes SISO et SIMO. La comparaison entre les systèmes SISO, SIMO et MIMO ont montré que les systèmes MIMO sont plus efficaces, mais le système MIMO présente des limites. On n'a montré que le système MIMO massif complète les limites avec la diversité spatiale inhérente et c'est très efficace pour réduire l'impact des différents problèmes tels que la limitation de la bande fréquentielle du canal, les évanouissements et les trajets multiples, sur les performances des systèmes. L'utilisation du TDD pour le MIMO massif a le potentiel de devenir une technologie qui change la donne dans l'espace des communications cellulaires, en permettant d'augmenter la capacité et l'efficacité des cellules. Grâce à la réciprocité du canal entre les antennes de la station de base et les utilisateurs, toute la complexité du traitement du signal peut être conservée à la station de base, et la caractérisation du canal peut être effectuée sur la liaison montante.

L'importance des systèmes MIMO massif a suscité plusieurs recherches concernant l'association des codes polaires avec les MIMO massif, par conséquent il est nécessaire de faire l'état de l'Art.

CHAPITRE 3 : ETAT DE L'ART.

Dans ce chapitre nous allons faire l'analyse et la synthèse des différentes propositions récentes par rapport à notre sujet de mémoire. Premièrement nous allons comparer les performances des types de codage et de décodage polaire proposé dans ce sens, deuxièmement analysé les performances des MIMO massif avec codage/décodage polaire et en fin nous allons dégager les limites des propositions

3.1 Codage et décodage polaire.

Les codes polaires [1] ont été considérés comme forts candidats pour les systèmes de communication sans fil de cinquième génération (5G) [43]. Ils sont capables de fournir un faible taux d'erreur qui est très compétitif par rapport aux autres codes comme les codes LDPC et RS. Il est donc intéressant d'étudier les codes polaires pour les communications sans fil, notamment dans le contexte des systèmes MIMO massifs à entrées multiples et sorties multiples. Pour les réseaux 5G, l'Internet des objets [44] ils ont reçu une grande attention en tant qu'application importante pour réaliser les interconnexions entre un grand nombre de systèmes hétérogènes, tels que les appareils portables et les téléphones intelligents. Ces systèmes sont souvent des systèmes d'énergie et de latence et nécessitent donc des codes de longueur relativement courte avec de faible complexité de codage/décodage. Arıkan a montré que l'annulation successive (SC) à de faible complexité pour le décodage polaire, il permettrait d'atteindre la capacité des canaux discrets et sans mémoire (DMC) [1]. Plus récemment, il a été signalé que le décodage SC liste avec redondance cyclique (CRC) surpasse les codes LDPC [45-47]. Pour les applications à haut rendement énergétique, un certain nombre de variables moins complexes comme l'algorithme de décodage de listes polaires ont également été proposées [48-52].

Dans [53] une nouvelle famille de codes polaires a été proposée, dont les unités de polarisation sont irrégulières pour réduire la complexité des calculs de codage et de décodage. Pour améliorer les performances on rend inactif des codes polaire lorsque les canaux binaires sont suffisamment bons ou mauvais. Cependant, le code polaire assoupli [54] ne considère pas toutes les combinaisons possibles d'inactivation en particulier, une fois qu'une unité de polarisation est désactivée, les autres unités de polarisations restent actives. Cela limite la flexibilité dans la sélection d'unités de polarisation qui une fois désactivées il ne sera pas exploiter donc la complexité est réduite davantage. Dans [53], des codes polaires totalement irréguliers ont été proposés en prenant la fiabilité non uniforme des canaux de bits en

effectuant l'inactivation des unités de polarisation. L'approche traditionnelle des codes polaires réguliers peuvent être améliorées considérablement en réduisant la complexité de l'encodage et le décodage jusqu'à 50%.

3.1.1 Codes polaires irréguliers.

La polarisation irrégulière consiste à inactiver certaines unités de polarisation pour réduire la complexité, améliorer les performances et réduire le taux d'erreur.

Le taux d'erreur (WER) des codes polaires irréguliers pour le canal d'effacement binaire (BEC) peut être exprimé par l'évolution du paramètre de Bhattacharyya [1]. En partant du canal, le paramètre de Bhattacharyya est donné par :

$$Z_i^n \triangleq \sum_{y_i \in \mathcal{Y}} \sqrt{W_i(y_i|0)W_i(y_i|1)} = \epsilon. \quad (3.1)$$

où ϵ est la probabilité d'effacement du BEC

Soit $Z_{r_U}^l$ et $Z_{r_L}^l$ les paramètres entrants de Bhattacharyya, respectivement, pour les branches supérieures et inférieures à l'unité de polarisation de $U(r, l)$. Si une unité de polarisation $U(r, l)$ est désactivée, la fiabilité des bits ne changent pas on a :

$$Z_{r_L}^{l-1} = Z_{r_L}^l, \quad Z_{r_U}^{l-1} = Z_{r_U}^l. \quad (3.2)$$

Considérant que l'évolution des paramètres de Bhattacharyya est donnée pour les unités de polarisation activée comme suit :

$$Z_{r_U}^l = Z_{r_U}^l + Z_{r_L}^l - Z_{r_L}^l Z_{r_U}^l, \quad Z_{r_U}^{l-1} = Z_{r_L}^l Z_{r_U}^l. \quad (3.3)$$

Cette évolution ci-dessus sera identique à celle des codes polaires [1] uniquement si les paramètres Bhattacharyya entrants aux branches supérieures et inférieures des unités de polarisation sont égaux, soit $Z_{r_L}^l = Z_{r_U}^l$. Afin de tenir compte d'une éventuelle non-uniformité de la fiabilité des bits, nous utiliserons l'évolution modifiée en (5). Les codes dont l'inactivation est irrégulière peuvent, par nature, produire des fiabilités non uniformes même lorsque les canaux sont identiques. Cela est illustré à la figure 3(d), où une polarisation inactivée de l'unité $U(1,2)$ implique des paramètres de Bhattacharyya non égaux pour le premier stade de polarisation $U(1,1)$ et $U(2,1)$. Une fois que nous avons obtenu les paramètres de Bhattacharyya Z_i^0 , le WER P_ϵ du décodage SC est exprimé comme suit [1] :

$$P_\epsilon = 1 - \prod_{i \in K} (1 - \frac{Z_i^0}{2}) < \sum_{i \in K} \frac{Z_i^0}{2} \quad (3.4)$$

La figure 3.1, présente les paramètres WER et Bhattacharyya à chaque unité de polarisation pour un taux d'effacement du canal de $\epsilon = 0,5$. Le WER du code polaire régulier conventionnel est représenté à la figure 3(a) et le $P_e = 0,2365$, alors que les codes polaires irréguliers sont représentés dans les figures 3.1 (b), (c) et (d) et ils donnent respectivement les valeurs des P_e suivantes $P_e = 0,2365$; $0,21875$ et $0,226560$. Cela indique qu'avec une inactivation appropriée de la polarisation les codes polaires irréguliers peuvent être plus performants que les codes polaires réguliers. Plus important encore, étant donné aucune calcul n'est nécessaire pour l'unité de polarisation inactivées pendant l'encodage et le décodage, la complexité de calcul des codes polaires irréguliers peut être réduit de manière significative sans aucune pénalité.

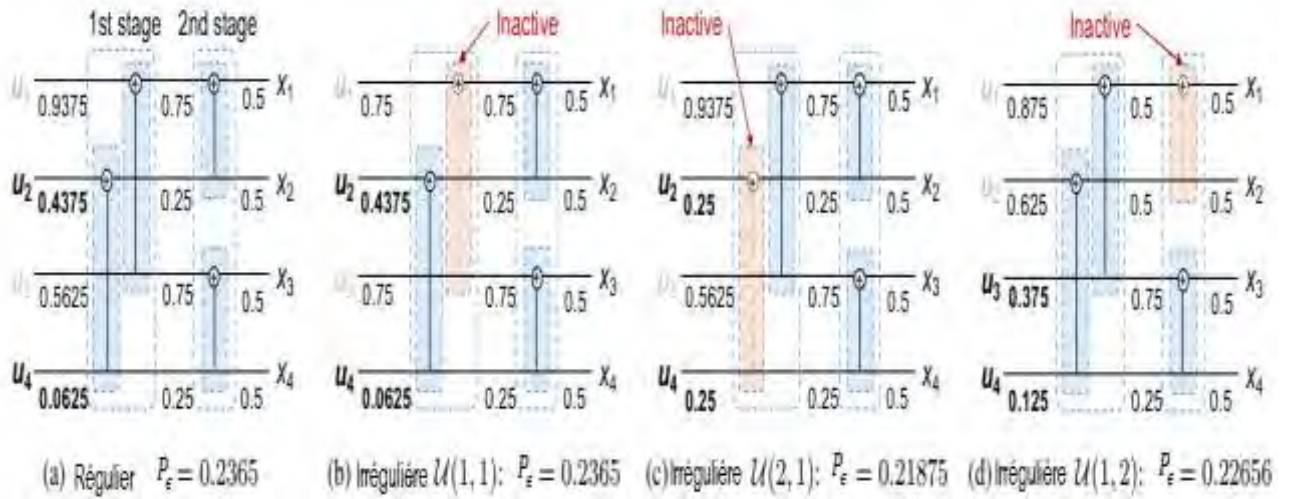


FIGURE 3.1: Polarisation des codes polaires irrégulier (4, 2) le WER correspondant est P_e [53].

3.1.2 Codage/décodage CA-SCL.

Comme le décodeur à annulation successive (SC) lorsque la taille du bloc de code N est suffisamment longue [58]. La complexité pour le codage et le décodage sont maintenus à l'ordre de $O(N \log N)$. Alors que pour les blocs de taille courte et modérée, la performance des codes polaires avec le décodage SC n'est pas aussi bonne que les codes LDPC ou Turbo code. Le décodage SC-List (SCL) fonctionne presque de la même manière que l'algorithme optimal de décodage à maximum de vraisemblance (ML) lorsque la taille du bloc de code est modérée. [55][53] ont proposé d'ajouter le CRC aux bits d'information, le SCL aidé par le CRC (CA-SCL) surpasse le Turbo et le LDPC pour des tailles de blocs de code courtes et modérées. Les principes de codage/décodage polaires sont présentés comme suit :

-Notation :

Soit une matrice $F(2, 2) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, $F^{\otimes n}$ est la matrice $N \times N$, où $N = 2^n$, $\otimes n$ désigne le nième puissance de Kronecker, et $F^{\otimes n} = F \otimes F^{\otimes(n-1)}$. La matrice génératrice est défini comme suit : $G_N = B_N F^{\otimes n}$, où B_N est une matrice de permutation. Le code polaire est généré par : $x_1^N = u_1^N G_N = u_1^N B_N F^{\otimes n}$ où $u_1^N = [u_1, u_2, \dots, u_N]$ et $x_1^N = [x_1, x_2, \dots, x_N]$ sont respectivement les séquences de bits sources et de bits encodés. La séquence de bits source u_1^N contient les bits d'information et les bits gelés, la Figure 3.2 est une représentation d'un encodeur ou la séquence des bits sources d'information son $u_1^8 = \{u_1, u_2, \dots, u_8\}$ il est composé des bits $= \{i_1, i_2, i_3, i_4\}$ et les bits gelés $\{0, 0, 0, 0\}$. La matrice génératrice $F^{\otimes n}$ est obtenu par l'addition modulo \oplus et on obtient des bits codés $\{x_1, x_2, u_3, u_4, u_5, u_7, u_7, u_8\}$.

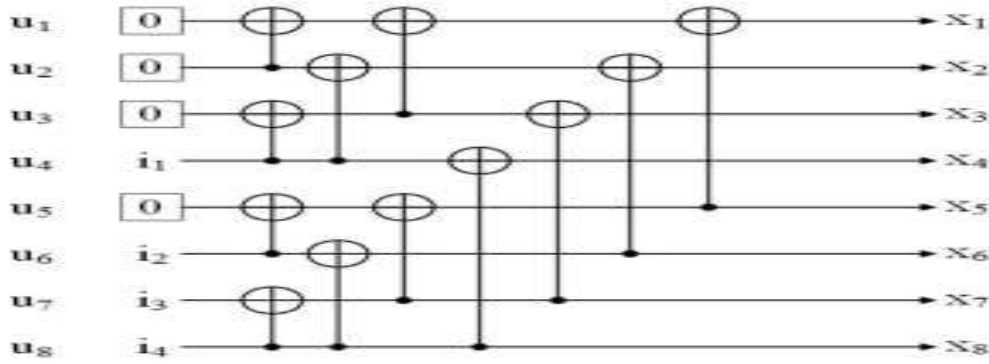


FIGURE 3.2 : L'encodeur polaire a 8bits [55]

Le décodeur SCL est proposé dans [55] ou on utilise un arbre binaire complet avec la profondeur de N , chaque arbre binaire est associé à un bit codé x_n , $1 \leq n \leq N$. Pour chaque nœud de la feuille, le côté gauche et le côté droit de chaque nœud prend respectivement les bits 0 et 1 et le chemin du nœud racine à n'importe quel nœud feuille peut être considéré comme une séquence de décodage. À n'importe quel nœud de l'arbre binaire une valeur est mesure et calculée pour le chemin du nœud racine [56]. Sur chaque couche, les L chemins ayant les plus grandes mesures sont sélectionné, L est également appelé largeur de recherche. Par conséquent, les séquences L qui représentent les bits de décodage du candidat peuvent être obtenues à la dernière couche. Les derniers bits de décodage sont ceux qui comportent la plus grande mesure. La figure 3.4 montre une illustration du décodeur Polaire a 4 bits et qui a une largeur de recherche $L = 2$, les valeurs sont étiquetés à côté de chaque nœud.

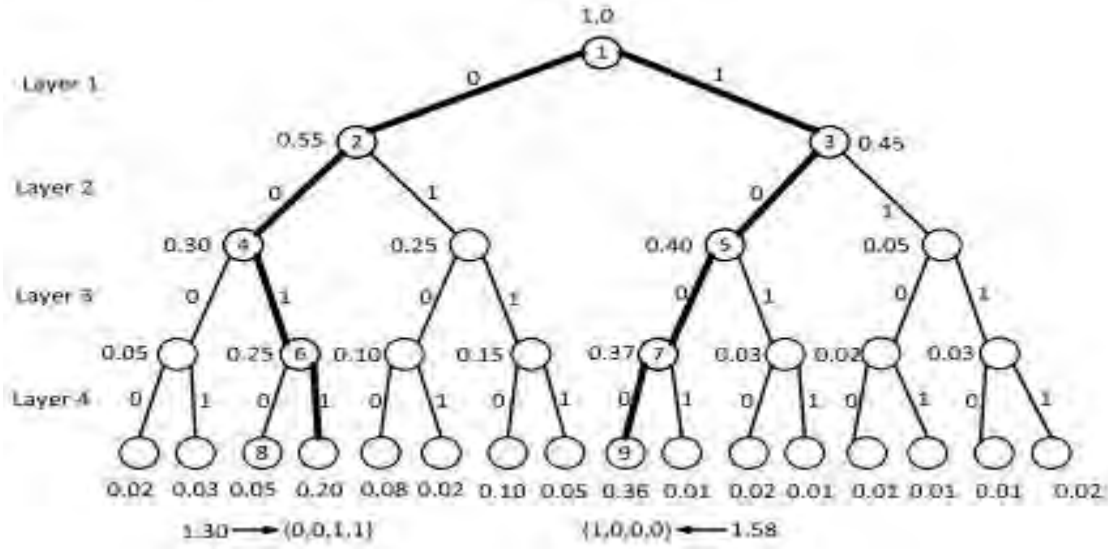


FIGURE 3.3 : Le décodeur polaire SCL a 4bits [55]

Deux séquences de décodage $\{0, 0, 1, 1\}$ et $\{1, 0, 0, 0\}$ sont trouvées après la recherche par couches, puis séquentielle. Les bits $\{1, 0, 0, 0\}$ présente la plus grande mesures avec 1.58 donc ils sont sélectionnées comme les bits décodés. [55] propose une amélioration du décodeur SCL car malgré les performances sur la capacité, elle présente également des inconvénients sur la faible vitesse et une latence élevée en raison du schéma de décodage un par un. Un décodeur SC-List parallèle est proposé pour améliorer les performances du décodeur SCL conventionnel [57]. Le décodeur parallèle est constitué de $M = 2^m$ décodeurs de composants travaillant en parallèle, chaque décodeur de composants décode un signal polaire avec la taille de bloc de $1/M$ du code polaire original. Ainsi, l'efficacité du décodeur SCL peut être encore améliorée. Deux décodeurs SCL parallèles A et B sont utilisés pour décoder $a_1^{\frac{N}{2}}$ et $b_1^{\frac{N}{2}}$ respectivement de $y_1^{\frac{N}{2}}$ et $y_{\frac{N}{2}+1}^{\frac{N}{2}}$. Selon le principe de l'analyse de recherche

dans l'arbre binaire du décodeur SCL, les chemins L sélectionnés à la couche $k - 1$ pour les décodeurs A et B peut être formulée comme suite : $P_{k-1} = \{P_{k-1}^A, P_{k-1}^B\}$, où $1 \leq k \leq N/2$ avec $P_{k-1}^A = \{A_1^{k-1}, A_2^{k-1}, \dots, A_{L_A}^{k-1}\}$ et $P_{k-1}^B = \{B_1^{k-1}, B_2^{k-1}, \dots, B_{L_B}^{k-1}\}$, nous avons $L_A + L_B = L$. Chaque chemin du décodeur A et B peut être formulé par une séquence. Lorsque le décodage passe à la couche k , chaque chemin pour le décodeur A ou B est complété par 2 nouvelles voies. Par conséquent, le nombre total de chemins pour les deux décodeurs A et B obtiennent $2L$. Ensuite, nous combinons tous les chemins du décodeur A et B pour sélectionner les chemins L qui on les plus grandes mesures. Le nombre de chemins combinés peut être classés par bits de source $\{v_k, v_{k-N/2}\}$ comme suite : v_k et $v_{k-N/2}$ sont des bits gelés, nous avons

$a_{l,k} = b_{l,k} = 0$, donc le bord droit de chaque nœud étiqueté avec 1 est exclue, on réalise la recherche du chemin A_l^{k-1} seulement en 0 pour le décodeur. La même observation s'applique également pour le décodeur B. Comme la recherche se poursuit avec le chemin fixe, le nombre total de chemins reste à L.

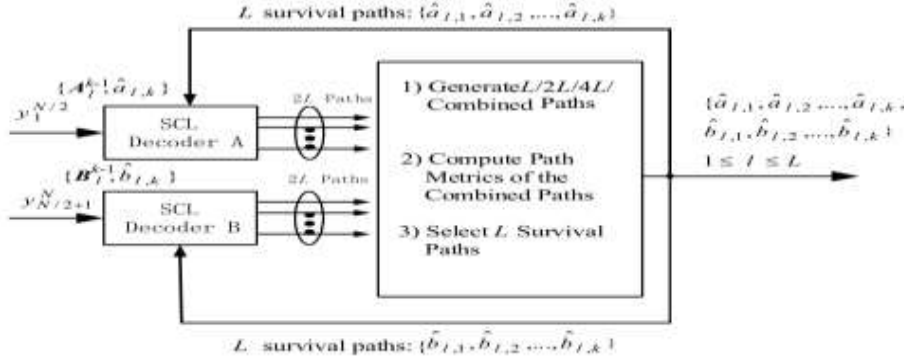


FIGURE 3.4 : Décodeur polaire parallèle à 2 composantes.[55]

Le décodeur SCL malgré ces performances il présente des limites par rapport à sa vitesse et le temps de latence, pour améliorer ces limites plusieurs propositions ont été faite, parmi les propositions :

[53] propose le décodage CA-SCL pour les codes polaires irréguliers, même avec l'inactivation des unités de polarisation dans les régions les codes, la modification pour l'encodage et le décodage n'est pas lourde car elle ne nécessite aucun calcul pour l'unité de polarisation inactivée. Par conséquent, la complexité du codage et du décodage peut être réduit de manière significative par rapport au nombre total d'unités de polarisations actives. [53] utilise des codes polaires avec les unités de polarisation élaguées sont en partie liées à une liste arborescente de décodage [59.16], dans lequel la complexité du décodage est réduite en élaguant les branches d'arbres comportant des morceaux gelés pour éviter les calculs inutiles. Toutefois, on applique une modifie directe de la distribution des degrés de la matrice générateur du code polaire pour réduire considérablement la complexité de l'encodage et du décodage sans limitation de la structure arborescente, qui est indépendant des bits, et cela donne la possibilité d'intégrer toute autre méthode codage/décodage y compris [59.16] pour réduire encore la complexité du décodage. Nous constatons que le concept de codage polaire irrégulier est applicable aux noyaux non binaires car en élaguant le noyau ou en le remplaçant par d'autres grains de plein Nrang de manière irrégulière, cela confirme que la méthode de codage polaire systématique dans [60.29] reste applicable à notre codage polaire irrégulier.

[61] propose aussi le décodeur CA-SCL en ajoutant le CRC (Cyclic Redundancy Check). Le décodeur CA-SCL augmente de manière considérable les performances du décodeur SCL en combinant le décodage des listes avec le CRC et chaque bit d'information \hat{u}_i dont les deux estimations possibles $\hat{u}_i = 0$ et $\hat{u}_i = 1$ sont réservés dans les chemins de décodage sous les contraintes de la taille du liste L. Ensuite, parmi les candidats ce lui qui passe à la sortie avec succès les tests du CRC est sélectionné comme le bon code décodé (ou celle dont le chemin est le plus petit), si aucun candidats ne passe avec succès le test CRC. Cependant, le décodage CA-SCL présente un problème car il y a aucun moyen efficace de reconstituer les résultats, bien que les performances du décodage CA-SCL est bien meilleure que celle des algorithmes BP ou SCAN. Pour résoudre ce problème [2019] utilise l'appliquer l'algorithme de Chase, proposé par Pyndiah dans [62.15], pour générer les informations avec les chemins L existants fournies par le décodeur CA-SCL.

3.1.3 Comparaisons entre code polaire et LDPC applique au MIMO massif.

Pour atteindre un compromis entre la complexité et les frais généraux [61] propose de comparer le codage polaire et LDPC entre les systèmes MIMO massif avec récepteurs itératifs, Il convient de noter que la performance du récepteur avec les codes polaires est toujours plus avantageuse par rapport aux codes LDPC sous différents conditions. [61] montre également aussi qu'une liste plus longue peut apporter une meilleure performance des systèmes à codage polaire. Une augmentation de la longueur du code de 256 à 512 peut permettre au récepteur d'atteindre 0,5 dB supplémentaire après 4 itérations, tandis que la modulation d'ordre supérieur conduit plutôt à un gain moindre. Ces deux résultats résultent du changement du nombre de symboles souples. Plus précisément, plus le nombre de symboles disponibles pour aider à l'estimation du canal augmente, plus le gain d'itération augmente.

3.2 Conclusion.

Nous avons proposé dans ce chapitre le bilan sur les différentes propositions récent en codage/décodage polaire en MIMO massif. Dans ce document, nous proposons un récepteur itératif pour les systèmes MIMO massifs avec codage polaire. Avec suffisamment de symboles souples pour faciliter l'estimation du canal et on n'a montré aussi que les résultats peuvent être considérablement amélioré par des itérations et cela surpasse les codes LDPC en matière de performance. Cela nous permet de proposer dans la chapitre qui suit une amélioration des performances des codes polaires comparait au codes LDPC avec différente itération

CHAPITRE 4 : ANALYSES ET INTERPRETATIONS DES RESULTATS.

4.1 Introduction.

Dans ce chapitre nous allons d'abord présenter en première lieu les codes polaires avec ces différentes décodeurs SC, SCL et CA-SCL en suite décrire le model proposés pour une transmission MIMO massif avec codage polaire et décodage CA-SCL et en fin l'analyses et l'interprétassions des résultats sont présenter .Une simulation d'antenne a antenne est proposer avec une analyse et des perspectives pour poursuivre les recherches dans ce domaine.

4.2 LES CODES POLAIRES.

Les codes polaires, comme premiers codes à capacité prouvée sur des entrées binaires sans mémoire discrète ce sont les seuls codes pour lesquels on peut démontrer mathématiquement qu'ils atteignent la limite de Shannon pour un code de taille infinie. [1], ont fait l'objet d'une attention particulière parmi divers codes de correction d'erreur directe (FEC). En raison de leur structure explicite et leur faible complexité $O(N \log 2N)$ de codage/décodage, les codes polaires sont devenus un des codes les plus importants de la théorie du codage. A ce jour, de nombreux efforts ont porté sur plusieurs aspects théoriques des codes polaires [1]. Dans [11], [12], [19], Les décodeurs polaires à annulation successive (SC) ont été présentés. Ces architectures de faible complexité conviennent à des applications à contraintes de surface ; toutefois, en raison de la complexité inhérente des natures de l'algorithme SC, ces décodeurs SC ne sont pas à une longue latence et un faible débit.

4.2.1 Notation.

on notera a_i^N pour désigner un sous-vecteur (a_i, \dots, a_N) . \oplus Pour addition modulo 2 et \otimes pour désigner le produit de Kronecker.

4.2.2 La construction de Codes Polaires

Un Code Polaire, CP $(N; K)$, est un code en bloc linéaire [1] de taille $N = 2^n$, n étant un entier naturel, contenant K bits d'informations. La matrice génératrice du code est une sous-matrice de la n ème puissance de Kronecker de $K = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, notée $\mathcal{G} = K^{\otimes n} = \begin{bmatrix} K^{\otimes n-1} & 0_{n-1} \\ K^{\otimes n-1} & K^{\otimes n-1} \end{bmatrix}$, composée de K lignes. Ces K lignes sont choisies en supposant un décodage à annulation Successive (SC) qui permet de polariser la probabilité d'erreur des bits du message. Un Code Polaire peut être représenté sous forme matricielle à

partir de la matrice F et sous forme de factor graph. Ce dernier pouvant être utilisé pour le codage et pour le décodage.

4.2.3 Processus de codage

Comme tout code en bloc, les mots de codes associés à des Codes Polaires sont définis par une matrice génératrice, G, de dimension $(K \times N)$. Cette matrice génératrice, G, est obtenue en supprimant $N - K$ lignes de la matrice F. Le processus de codage équivalent consiste ensuite à multiplier un vecteur de taille K par cette matrice G. Un processus de codage alternatif consiste à construire un vecteur, noté U, contenant les K bits d'information et $N - K$ bits gelés (Ces bits gelés, forcés à 0, sont placés au niveau des indices les moins fiables) fixés à 0. Ce vecteur est construit de telle manière que les bits d'information soient localisés sur les indices les plus fiables correspondant aux K lignes de $K^{\otimes n}$ sélectionnées précédemment. Le mot de code correspondant, X, peut ensuite être calculé simplement tel que $X = U \times K^{\otimes n}$.

Exemple 1 Pour un code CP (8;4) et un message $U = [0, 0, 0, u_3, 0, u_5, u_6, u_7]$, avec la position des bits gelés arbitraire, le mot de code est obtenu par la multiplication matricielle suivante :

$$[0, 0, 0, u_3, 0, u_5, u_6, u_7] \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} u_3 + u_5 + u_6 + u_7 \\ u_3 + u_5 + u_7 \\ u_3 + u_6 + u_7 \\ u_3 + u_7 \\ u_5 + u_6 + u_7 \\ u_5 + u_7 \\ u_6 + u_7 \\ u_7 \end{bmatrix}^T$$

Un code en bloc peut-être représenté sous la forme d'un factor graph. Dans le cas des Codes Polaires, nous avons vu que la construction de la matrice génératrice est récursive. Il est alors possible de montrer que la construction du graphe est également récursive.

Exemple 2. Dans cet exemple, deux matrices génératrices ainsi que leur représentation graphique sont présentées. Dans la figure 4.1 la matrice génératrice et le graphe de codage d'un Code Polaire de taille $N = 2$ et de message $U = [u_0, u_1]$ sont présentés.

$$U \times \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \Rightarrow \begin{array}{c} \begin{array}{cc} u_0 & u_0 + u_1 \end{array} \\ \hline \begin{array}{cc} u_1 & u_1 \end{array} \end{array}$$

FIGURE 4.1 : Matrice génératrice et graphe de codeur Polaire CP(2,2).

De même dans la figure 4.2, la matrice génératrice et le graphe de codage d'un Code Polaire de taille $N = 4$ et de message $U = [u_0, u_1, u_3, u_4]$, sont donnés.

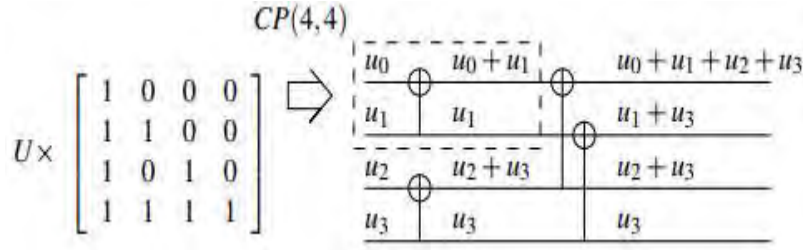


FIGURE 4.2 : Matrice génératrice et graphe de codeur Polaire CP(4,4).

Les nœuds de parités (\oplus) se comportent comme des XOR et les nœuds de variables (?) font simplement passer la valeur binaire à l'étage suivant. On peut remarquer alors qu'un code de taille $2N$ utilise deux fois la structure d'un code de taille N puis fait la somme des N premières lignes et transmet directement les N dernières lignes comme démontré ci-dessous :

$$U_{n+1} = [U_n, U'_n]$$

Avec :

U_n Les N premiers bits de U_{n+1}

U'_n Les N derniers bits de U_{n+1}

$$\begin{aligned} U_{n+1} \times \mathcal{F}_{n+1} &= [U_n, U'_n] \times \mathcal{F}_{n+1} \\ &= [U_n, U'_n] \times \begin{bmatrix} \mathcal{F}_n & 0_n \\ \mathcal{F}_n & \mathcal{F}_n \end{bmatrix} \\ &= [U_n \times \mathcal{F}_n + U'_n \times \mathcal{F}_n, U'_n \times \mathcal{F}_n] \\ &= [(U_n, U'_n) \times \begin{bmatrix} \mathcal{F}_n \\ \mathcal{F}_n \end{bmatrix}, U'_n \times \mathcal{F}_n] \end{aligned}$$

Plus généralement, le factor graph d'un Code Polaire, présenté dans [1], de taille $N = 2^n$ est composé de n étages de $\frac{N}{2}$ nœuds de parité de degré 3 et $\frac{N}{2}$ nœuds de variables de degré 3. Le degré d'un nœud représente son nombre de connections avec d'autres nœuds. Le factor graph peut être utilisé pour le codage et le décodage. Pour le codage, le vecteur d'entrée U , sur le côté gauche, est propagé dans le graphe dans le but de générer le mot de code X , sur la droite. Un Code Polaire de rendement $R = 0.5$ est donné en exemple dans la figure 1.13 ; cela signifie que la moitié des bits du vecteur U sont des bits gelés et l'autre moitié est composée de bits d'information d_i .

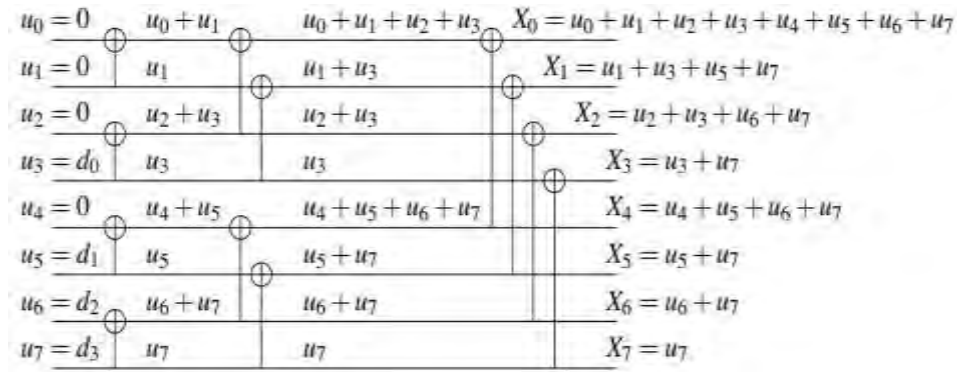


FIGURE 4.3 : Factor graph d'un codeur de Code Polaire CP (8;4) non systématique.

4.2.4 Le décodage par annulation successive (SC)

L'algorithme de décodage SC permet de décoder les Codes Polaires. Un **algorithme** est un ensemble de règles opératoires dont l'application permet de résoudre un problème énoncé au moyen d'un nombre fini d'opérations. (Dictionnaire Larousse) Un algorithme se caractérise en particulier par sa complexité. Cette dernière correspond au nombre d'opérations nécessaires pour compléter l'algorithme.

Un algorithme de décodage est une manière de traiter l'information reçue afin de récupérer, avec le moins d'erreurs possible, ce qui a été envoyé. L'information reçue correspond au mot de code qui a traversé le canal. Le décodage va alors permettre de récupérer le message transmis, avec le moins d'erreur possible, à partir de ces informations du canal. Les algorithmes sont plus ou moins complexes et performants. Les performances de décodage de l'algorithme représentent sa capacité à corriger les erreurs du mot reçu. En général, nous verrons que ces deux notions sont opposées. Pour les Codes Polaires, un premier algorithme a été proposé dans [1] et est détaillé dans le paragraphe qui suit.

-Décodage SC.

Une fois le message transmis au travers du canal de communication, la version bruitée,

$Y = Y_0^{N-1} = [Y_0, Y_1, \dots, Y_N]$, du mot de code $X = X_0^{N-1} = [X_0, X_1, \dots, X_N]$ est reçue. Le but du décodage est d'estimer le vecteur U à partir de la version bruitée du mot de code (Y). Arıkan a montré, que les Codes Polaires atteignaient la capacité du canal sous l'hypothèse d'un décodage par annulation successive. Ce décodage consiste à estimer un bit u_i à partir de l'observation du canal et de la connaissance des bits précédemment estimés. La valeur du bit estimé est notée \hat{u}_i .

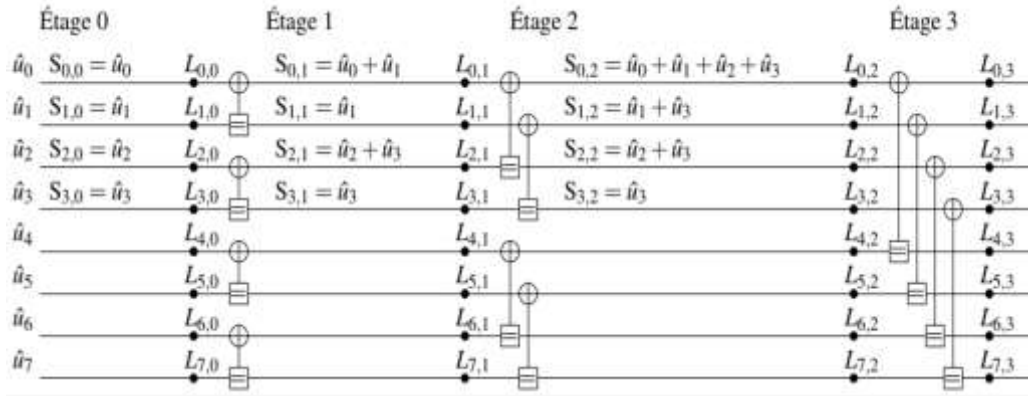


FIGURE 4.4 : Factor graph d'un Code Polaire de taille $N = 8$.

Chaque échantillon Y_i est converti dans un format appelé Likelihood Ratio - Rapport de vraisemblance (LR). Ces LR, notés $L_{i,n}$, sont positionnés sur le graphe comme dans la figure 1.14 pour $N = 8$, avec $0 \leq i \leq N - 1$. Les nœuds de parités (\oplus) et les nœuds de variables (\ominus), représentés différemment, ne symbolisent plus un simple XOR ni une simple copie. Au cours du décodage les différentes valeurs $L_{i,j}$ ($0 \leq j \leq n$) sont mise à jour ainsi que les valeurs $S_{i,j}$. Ces dernières, appelées sommes partielles, représentent le recodage des bits \hat{u}_i , au fur et à mesure de leur estimation. Le séquençement particulier des opérations est expliqué ci-après.

Tout d'abord, il est à noter que la mise à jour des LR, $L_{i,j}$, et des sommes partielles, $S_{i,j}$, peut être calculée efficacement en utilisant la représentation graphique des Codes Polaires. En effet, la matrice génératrice est inversible dans l'ensemble F_2 et son inverse est-elle même. On peut donc aisément montrer par récurrence que $\mathcal{F} \times \mathcal{F} = I_N$.

$$\begin{bmatrix} K^{\otimes n-1} & 0_{N/2} \\ K^{\otimes n-1} & K^{\otimes n-1} \end{bmatrix} \times \begin{bmatrix} K^{\otimes n-1} & 0_{N/2} \\ K^{\otimes n-1} & K^{\otimes n-1} \end{bmatrix} = \begin{bmatrix} I_{N/2} & 0_{N/2} \\ 0_{N/2} & I_{N/2} \end{bmatrix} \quad (4.1)$$

Nous pouvons alors exprimer l'égalité suivante :

$$[X] \times \mathcal{F} = [U \times \mathcal{F}] \times \mathcal{F} = U : \quad (4.2)$$

Pour estimer successivement chaque bit u_i , le décodeur se base sur l'observation du vecteur issu du canal $L_{(0,n)}^{(N-1)} = [L_{(0,n)} \dots L_{(N-1,n)}]$ et des bits précédemment estimé $\hat{u}_0^{i-1} = [\hat{u}_0 \dots \hat{u}_{i-1}]$. Dans ce but, le décodeur doit calculer les valeurs des LR suivants :

$$L_{i,0} = \frac{\Pr(Y_0^{N-1}, \hat{u}_0^{i-1} | u_i = 0)}{\Pr(Y_0^{N-1}, \hat{u}_0^{i-1} | u_i = 1)} \quad (4.3)$$

Au cours du décodage, lors de la mise à jour d'un étage $j > 0$, ce ne sont pas les bits \hat{u}_i qui sont utilisés directement, mais les sommes partielles, $S_{i,j}$, qui sont une combinaison de ces bits estimés. Lorsque le décodage met à jour un LR de l'étage 0, $L_{i,0}$, alors le décodeur prend une décision quant la valeur du bit \hat{u}_i est tel que :

$$\hat{u}_i = \begin{cases} 0 & \text{si } L_{i,0} > 1 \\ 1 & \text{sinon.} \end{cases} \quad (4.4)$$

Le décodeur a connaissance des bits gelés. Par conséquent, si u_i est un bit gelé, alors $\hat{u}_i = 0$ peu importe la valeur de $L_{i,0}$. L'algorithme de décodage est appelé Annulation Successive (SC), se déroulement est expliqué ci-après. Le décodeur estime successivement les bits \hat{u}_i à partir des LR, $L_{i,j}$, et des sommes partielles, $S_{i,j}$, qui sont calculés tels que :

$$L_{i,j} = \begin{cases} F(L_{i,j+1}, L_{i+2^j,j+1}) & \text{si } B_{i,j} = 0 \\ G(L_{i-2^j,j+1}, L_{i,j+1}, S_{i-2^j,j}) & \text{si } B_{i,j} = 1 \end{cases}, \quad (4.5)$$

$$S_{i,j} = \begin{cases} H_1(S_{i,j-1}, S_{i+2^{j-1},j-1}) & \text{si } B_{i,j-1} = 0 \\ G(L_{i-2^j,j+1}, L_{i,j+1}, S_{i-2^j,j}) & \text{si } B_{i,j-1} = 1 \end{cases}, \quad (4.6)$$

Avec :

$$\begin{cases} F(a, b) = \frac{1+ab}{a+b} \\ G(a, b, S) = b \times a^{1-2S} \\ H_u(S) = S \\ H_1(S, S') = S \oplus S' \\ B_{i,j} = \frac{i}{2^j} \bmod 2, 0 \leq i < N \text{ and } 0 \leq j < n \end{cases} \quad (4.7)$$

Preuve :

Notations

Nous travaillons dans l'ensemble fini binaire dont les valeurs sont contenues dans l'ensemble $\{0,1\}$, d'où l'opération de somme et de ou exclusif sont équivalentes. Dans la suite les notations suivantes sont utilisées :

- La probabilité de transition du canal W est notée $W(y_j|x_i)$. Cela représente la probabilité d'obtenir y_j sachant que x_i a été envoyé à travers le canal.
- Les vecteurs sont notés $u_a^b = [u_a, u_{a+1}, \dots, u_b]$, $a < b$
- Le rapport de vraisemblance (Likelihood Ratio - LR) : sachant que N est la taille du mot de code et que \hat{u}_i est le i^{me} bit à être estimé au cours du décodage, on a :

$$L_N^i(y_1^N, \hat{u}_1^{i-1}) = \frac{\Pr(y_1^N, \hat{u}_1^{i-1} | u_i = 0)}{\Pr(y_1^N, \hat{u}_1^{i-1} | u_i = 1)} \quad (4.8)$$

- La combinaison de canaux :
 \mathcal{X} est l'ensemble des mots de codes possibles en entrée du canal.
 \mathcal{Y} est l'ensemble des mots de codes possibles en sortie du canal.

K est le noyau de la matrice génératrice utilisée $K = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. Donc, pour coder un mot de code plus grand, la matrice est élevée à une puissance de Kronecker qui dépend de la taille du code.

Par exemple : $K^{\otimes 2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ $W_2 : x^2 \rightarrow y^2$

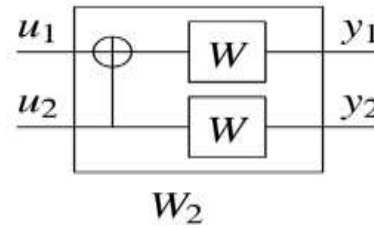


FIGURE 4.5 : codeur deux bits

$$w_2(y_1^2, u_1^2) = W(y_1|u_1 \oplus u_2) W(y_2|u_2) \quad w_2(y_1^2, u_1^2) = w_2(y_1^2, u_1^2) * G$$

– Probabilités associées à la règle de somme :

$$\Pr(A) = \sum_B \Pr(A, B) = \sum_B [\Pr(A|B)\Pr(B)]$$

$$\text{Si } A = (y_1^2|u_1 = 0)$$

$$w_2(y_1^2|u_1=0) = \sum_{u_2} [w_2(y_1^2|u_1 = 0, u_2) \times \Pr(u_2)]$$

La plupart du temps $\Pr(u_2) = \frac{1}{2}$, donc

$$w_2(y_1^2|u_1=0) = \frac{1}{2} \sum_{u_2} [w_2(y_1^2|u_1 = 0, u_2)]$$

Décodage

Les seules données disponibles sont celles provenant du canal. Cette fois, le traitement ne correspond pas seulement à des additions car nous travaillons avec des Likelihood Ratio - Rapport de vraisemblance (LR). L'exemple suivant décrit le principe de décodage sur un code de taille 2.

- décodage \hat{u}_1 de fonction F

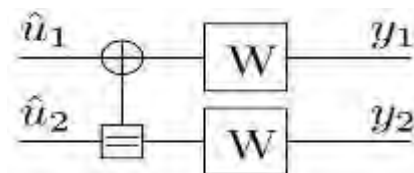


FIGURE 4.6 : Décodeur a deux bits

Nous cherchons à calculer le Likelihood Ratio - Rapport de vraisemblance (LR) de u_i .

1 :

$$\Pr(y_1^2|u_1=0) = \frac{1}{2} [w_2(y_1^2|u_1=0, u_2 = 0) + w_2(y_1^2|u_1=0, u_2=1)]$$

$$\begin{aligned}
&= \frac{1}{2} [W^2(y_1^2|u \cdot G) + W^2(y_1^2|u \cdot \bar{G})] \\
&= \frac{1}{2} [W(y_1|x_1=0) W(y_2|x_2=0) + W(y_1|x_1=1) W(y_2|x_2=1)]
\end{aligned}$$

2 : X

$$\begin{aligned}
\Pr(y_1^2|u_1=1) &= \frac{1}{2} [w_2(y_1^2|u_1=1, u_2=0) + w_2(y_1^2|u_1=1, u_2=1)] \\
&= \frac{1}{2} [W^2(y_1^2|u \cdot G) + W^2(y_1^2|u \cdot \bar{G})] \\
&= \frac{1}{2} [W(y_1|x_1=1) W(y_2|x_2=0) + W(y_1|x_1=0) W(y_2|x_2=1)]
\end{aligned}$$

3 : on applique le rapport entre 1 et 2

$$\begin{aligned}
L_2^{(1)}(y_1^2) &= \frac{\Pr(y_1^2|u_1=0)}{\Pr(y_1^2|u_1=1)} \\
&= \frac{\frac{1}{2} [W(y_1|x_1=0) W(y_2|x_2=0) + W(y_1|x_1=1) W(y_2|x_2=1)]}{\frac{1}{2} [W(y_1|x_1=1) W(y_2|x_2=0) + W(y_1|x_1=0) W(y_2|x_2=1)]} \\
&= \frac{\frac{W(y_1|x_1=0) W(y_2|x_2=0)}{W(y_1|x_1=1) W(y_2|x_2=1)} + 1}{\frac{W(y_2|x_2=0) + W(y_1|x_1=0)}{W(y_2|x_2=1) + W(y_1|x_1=1)}} \\
F(L_1^{(1)}(y_1), L_1^{(1)}(y_2)) &= \frac{1 + L_1^{(1)}(y_1) L_1^{(1)}(y_2)}{L_1^{(1)}(y_1) L_1^{(1)}(y_2)}
\end{aligned}$$

La fonction de décodage F est maintenant définie : $F(a, b) = \frac{1+a \times b}{a+b}$. Il reste encore l'autre LR à calculer.

- Décodage \hat{u}_2 de fonction F

Nous supposons que les bits précédents sont connus. Dans notre exemple \hat{u}_0 est connu et nous pouvons l'utiliser. Nous cherchons à calculer le Likelihood Ratio - Rapport de vraisemblance (LR) de u_2 .

$$L_2^{(2)}(y_1^2, \hat{u}_1) = \frac{\Pr(y_1^2, \hat{u}_1=0|u_2=0)}{\Pr(y_1^2, \hat{u}_1=0|u_2=1)}$$

1:

$$\Pr(y_1^2, \hat{u}_1=0|u_2=0) = \frac{1}{2} [w_2(y_1^2|\hat{u}_1, u_2=0)]$$

$$= \frac{1}{2} [W(y_1|x_1 = \hat{u}_1) W(y_2|x_2 = 0)]$$

2:

$$\begin{aligned} \Pr(y_1^2, \hat{u}_1 = 0 | u_2 = 1) &= \frac{1}{2} [w_2(y_1^2 | \hat{u}_1, u_2 = 1)] \\ &= \frac{1}{2} [W(y_1|x_1 = \hat{u}_1 \oplus 1) W(y_2|x_2 = 1)] \end{aligned}$$

3 : on applique le rapport entre 1 et 2

$$L_2^{(2)}(y_1^2, \hat{u}_1) = \frac{\frac{1}{2} [W(y_1|x_1 = \hat{u}_1) W(y_2|x_2 = 0)]}{\frac{1}{2} [W(y_1|x_1 = \hat{u}_1 \oplus 1) W(y_2|x_2 = 1)]}$$

Si $\hat{u}_1 = 0$:

$$\begin{aligned} L_2^{(2)}(y_1^2, \hat{u}_1) &= \frac{\frac{1}{2} [W(y_1|x_1 = 0) W(y_2|x_2 = 0)]}{\frac{1}{2} [W(y_1|x_1 = 1) W(y_2|x_2 = 1)]} \\ &= L_1^{(2)}(y_1, \hat{u}_1 = 0), L_1^{(2)}(y_2, \hat{u}_1 = 0) \end{aligned}$$

Si $\hat{u}_1 = 1$:

$$\begin{aligned} L_2^{(2)}(y_1^2, \hat{u}_1) &= \frac{\frac{1}{2} [W(y_1|x_1 = 1) W(y_2|x_2 = 0)]}{\frac{1}{2} [W(y_1|x_1 = 0) W(y_2|x_2 = 1)]} \\ &= \frac{L_1^{(2)}(y_2, \hat{u}_1 = 0)}{L_1^{(2)}(y_1, \hat{u}_1 = 0)} \end{aligned}$$

$G(L_1^{(2)}(y_1, \hat{u}_1 = 0), L_1^{(2)}(y_2, \hat{u}_1 = 0)) = L_1^{(2)}(y_2, \hat{u}_1 = 0) L_1^{(2)}(y_1, \hat{u}_1 = 0)^{1-2\hat{u}_1}$ La fonction G est maintenant définie : $G(a, b, SP) = b \times a^{SP_{i,j}}$. On remarque que pour calculer cette fonction, nous avons besoin d'un bit particulier noté $SP_{i,j}$. Ceci génère donc une dépendance, d'où un ordonnancement particulier pour le décodage.

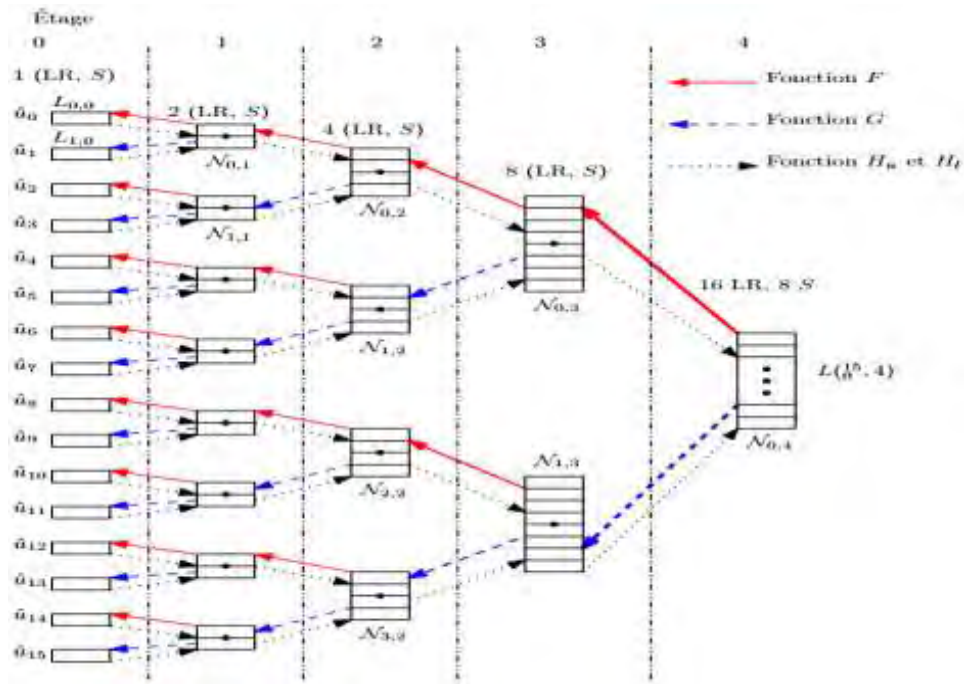


FIGURE 4.7 – Processus de décodage représenté sous la forme d'un arbre binaire.

La somme partielle $S_{i,j}$ correspond à la propagation des décisions dures dans le factor graph.

Par exemple, dans la figure 1.14, $S_{1,2} = \hat{u}_1 + \hat{u}_2$ (somme modulo 2).

L'algorithme de décodage peut être représenté par un arbre binaire complet, comme illustré dans la figure 1.17. L'arbre de profondeur 4 représente le décodage d'un mot de code de taille $N = 16$. Les branches symbolisent les fonctions F , G , H_u et H_l . Les nœuds représentent les LR et les sommes partielles intermédiaires, calculés durant le processus de décodage. Les fonctions F et G traitent les LR de la droite vers la gauche et stockent les résultats dans les nœuds de gauche. Les fonctions H récupèrent les sommes partielles du nœud de gauche, les traitent puis les stockent dans le nœud de droite. Dans la description actuelle de l'algorithme de décodage, un nœud N est mis à jour lorsque tous les nœuds qui sont à sa gauche sont mis à jour et que toutes ses sommes partielles sont mises à jour. Cet algorithme récursif s'exécute à partir de la racine de l'arbre. Le nœud en cours de mise à jour est appelé N . Si N a des fils, alors la branche supérieure est calculée. Les fonctions F traitent les LR de N et stockent les LR résultant dans le fils supérieur de N noté N_u . Quand N_u est mis à jour, la fonction H_u récupère les sommes partielles de N_u et les stocke dans N . Ensuite, la branche inférieure est calculée : les fonctions G traitent les LR de N , en utilisant les sommes partielles précédemment récupérées. Les LR résultant sont ensuite stockés dans le fils inférieur à N noté N_l . Quand N_l est mis à jour, la fonction H_l combine les sommes partielles de N_l et N et stocke le résultat dans N . Le nœud N est maintenant mis à jour. Si le nœud N n'a pas de fils

(niveau des feuilles) alors la somme partielle est obtenue en fonction de la valeur du LR de N . Le processus de décodage est récapitulé dans l'algorithme.

Algorithme 1 : Algorithme de décodage par annulation successive

Résultat	:	Sommes	partielles	de	N	mises	à	jour.
Entrées	:				Nœud			N
MAJ		(Node	N)	-	Mise	À		Jour
si		N	a	des	fil			alors
		1. Calculer F à partir des LR de N et stocker les LR résultant dans N_u						
		2. MAJ (N_u)						
		3. Calculer la fonction H_u : elle récupère les sommes partielles de et N_u les stocke dans N						
		4. Calculer G à partir des LR et sommes partielles de N et stocker les LR résultants dans N_1						
		5. MAJ (N_1)						
		6. Calculer la fonction H_1 : elle combine les sommes partielles de N_1 et N , les stocke ensuite dans N						
sinon								
		Prend une décision pour la somme partielle de N suivant la valeur de son LR						

Le décodeur d'annulation successive (SC) est simple et de faible complexité mais présente des limites. Par conséquent, des progrès importants ont été réalisés en ce qui concerne l'amélioration des performances avec une longueur de bloc courte ou modérée [17][18][19] d'où la proposition du décodage SCL.

4.2.5 Décodage SCL

On n'a montré au paragraphe précédent que le décodeur SC a des limites. On peut améliorer les performances de décodage médiocres SC pour des faibles tailles de codes en utilisant le décodeur SCL. Le décodeur SCL présenté dans cette section permet d'améliorer les performances pour les Codes Polaires de petite taille. L'algorithme de décodage SC-LIST a été introduit dans Vardy [23]. C'est une généralisation du décodage SC. L'algorithme de décodage peut considérer jusqu'à L chemins différents en parallèle au cours du processus de décodage. Par exemple, lorsqu'un bit u d'information est estimé, alors le processus de décodage est dupliqué. Un des processus considère que la valeur estimée est un 0, l'autre processus considère que la valeur estimée est un 1, comme dans l'exemple de décodage avec une liste de taille $L = 2$ d'un Code Polaire $CP(4;4)$ dans la figure 1.13. Il est à noter que lorsqu'un bit gelé est rencontré, le processus n'est pas dupliqué, car la valeur du bit gelé n'est pas modifiable. Quand plus de L processus sont considérés en parallèle, alors seulement les L

processus de décodage les plus probables, qui ont la plus grande métrique dans la liste, sont conservés, comme illustré dans la figure 1.13.

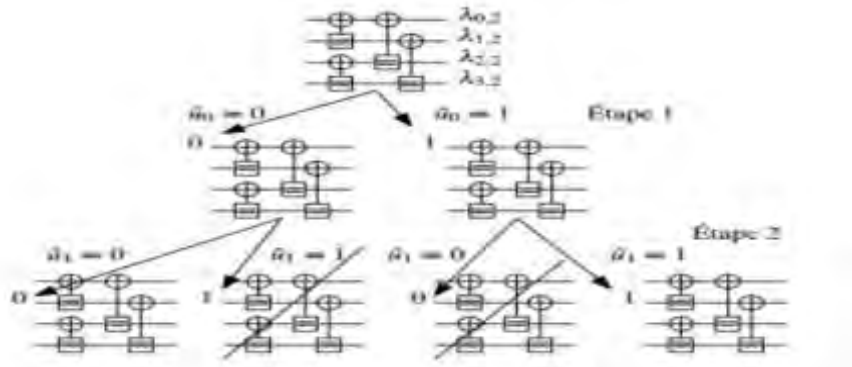


FIGURE 4.8 : Décodage SCL d'un Code Polaire $CP(4;4)$ avec une liste de taille $L = 2$.

4.2.6 Décodage CA-SCL.

Les codes polaires avec décodage SCL assisté par contrôle de redondance cyclique (CA-SCL) peuvent être supérieurs aux codes de contrôle de parité à faible densité (LDPC) en termes de performance et de complexité sous certaines conditions [20][21]. Ainsi, le décodage CA-SCL est le plus pratiqué. Malgré cela, ses performances et la complexité en longueur finie sont encore limitées dans les communications. Actuellement, beaucoup de recherches ont été faites sur la base de Codes polaires CA-SCL incluant l'amélioration de la performance et la réduction de la complexité.

Les codes polaires associés CRC (Cyclic Redundancy Check) sont proposés dans [22] en mettant l'accent sur la réduction de la complexité de décodage et de diminuer le temps de décision par rapport à une méthode de décodage SCL assisté par un seul CRC. Les résultats montrent que l'espace mémoire de décodage et le délai peuvent être réduits considérablement grâce à une conception raisonnable de multi-CRC. Cependant, la déficience des multi-CRC polaires est que la perte de performance ne peut être ignorée à des niveaux de rapport signal/bruit (SNR) importants, il s'agit donc d'un algorithme sous-optimal qui ne peut pas atteindre le décodage du maximum de vraisemblance (ML) le niveau le plus performant. Dans [23], le décodage polaire segmenté CA-SCL est proposé, qui fonctionne de telle sorte que de nombreuses CRC devraient être nécessaires. Le délai de décodage est réduit en mettant fin au calcul préalable des trajectoires inutiles et ce schéma peut atteindre les mêmes performances que le décodage CA-SCL. Mais, le décodage CA-SCL segmenté augmente la complexité du codage par rapport à la polarité d'origine de l'encodage avec un seul CRC.

Compte tenu de ce point, le code polaire CA-SCL est proposé [24.9][25.10]. Cette méthode est également utile pour améliorer les performances du BLER (**B**lock **E**rror **R**ate). Cependant, il n'y a pas de règle claire pour placer les bits de contrôle de parité mais on peut les placer en introduisant en fonction du poids de polarisation (PW) des canaux [26.11].

4.2.7 Performance des codes polaires.

Le code polaire [1], proposé en premier lieu par Arikan en 2009, est l'un des plus prometteurs en matière de correction d'erreurs ces dernières années. C'est le seul code correcteur d'erreur qui atteint mathématiquement la capacité des canaux symétriques discrets sans mémoire (BI-DMS) à entrer binaire. Par conséquent, il bénéficie d'une attention importante du à l'appui théorique de beaucoup de recherche.

4.2.8 Modèle de transmission mimo massif avec codage/décodage polaire.

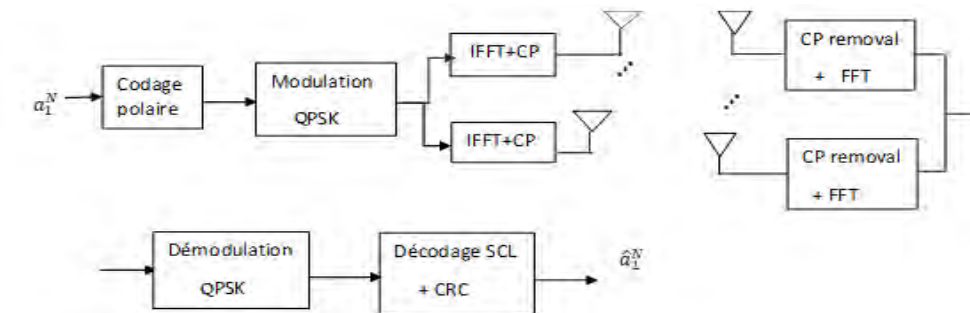


FIGURE 4.9 : Modèle de transmission mimo massif avec codage/décodage polaire.

Dans ce modèle de transmission on émet un message a_1^N qui est codé avec les codes polaires et moduler, pour chaque antenne d'émission on ajoute la transformée de Fourier rapide inverse IFFT et le CP est ajouté aussi mais au niveau de la réception on applique une transformée de Fourier rapide FFT, un removal CP, une démodulation et en fin le décodage CA-SCL est appliqué et le message reçu est sous la forme \hat{a}_1^N .

4.2.9 Analyses et interprétations des résultats.

La figure 4.2 étudie l'évolution des codes LDPC par rapport aux codes polaires en augmentant le nombre d'antenne progressivement pour les codes polaires, la figure 4.2 représente le FEC (10^0 à 10^{-3}) en fonction du SNR (-10 à 30dB).

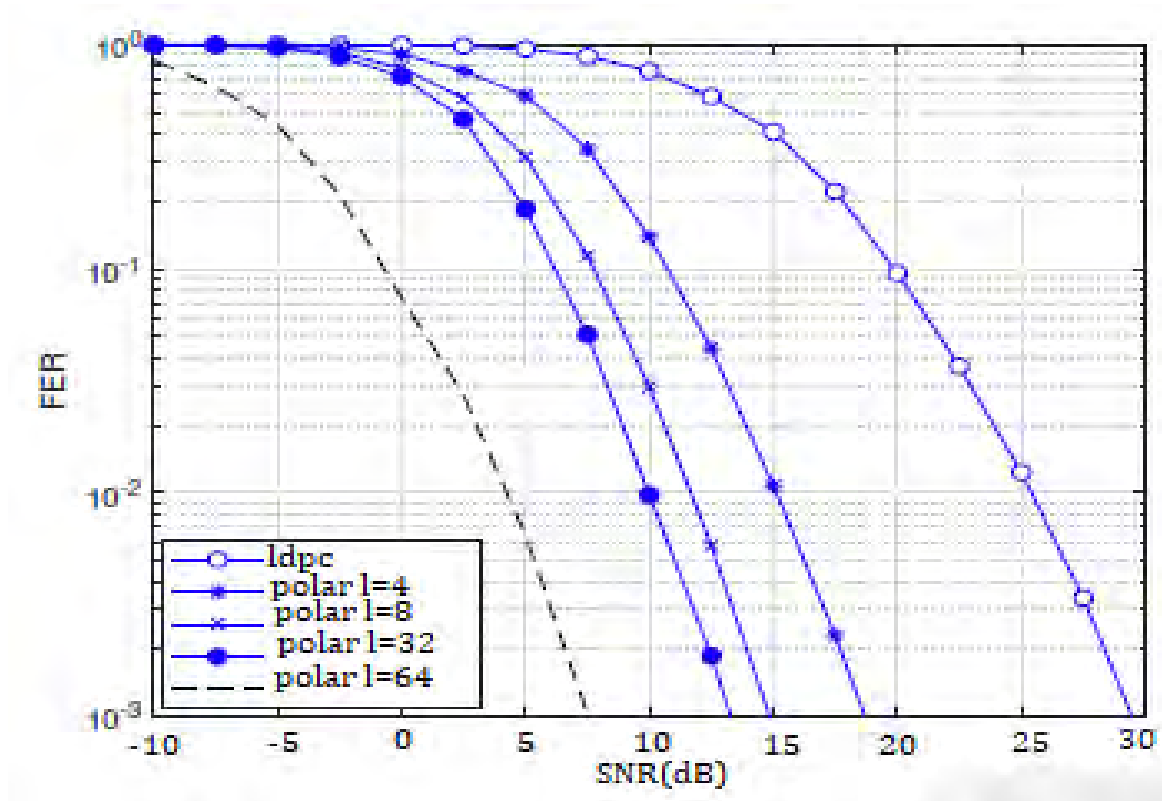


FIGURE 4.10 : Performances des codes polaires par rapport au code LDPC appliqué à un système MIMO MASSIF avec $N = 256$, $R = 0,5$ et une modulation QPSK.

Les résultats on montrer une diminution du SNR en fonction du FER des codes polaire par rapport au code LDPC cela confirme que les codes polaires sont plus performants que les codes LDPC. On note une nette diminution du SNR pour les codes polaire en augmentant le nombre d'antenne de 4 à 64 et le gain passe de 13dB et sa confirme les résultats obtenus par [Aliou Thiame] plus le nombre d'antennes est élevé plus les performances des systèmes mimo massif augmente cela est due à une augmentation du nombre de degré de liberté, qui donne plus de possibilité aux antennes de la station de basse BS.

La figure 4.11 représente la variation du FER (10^0 à 10^{-3}) en fonction du SNR (-5 à 30dB) par rapport au code polaire et LDPC en variant le nombre d'antenne et le nombre d'itération

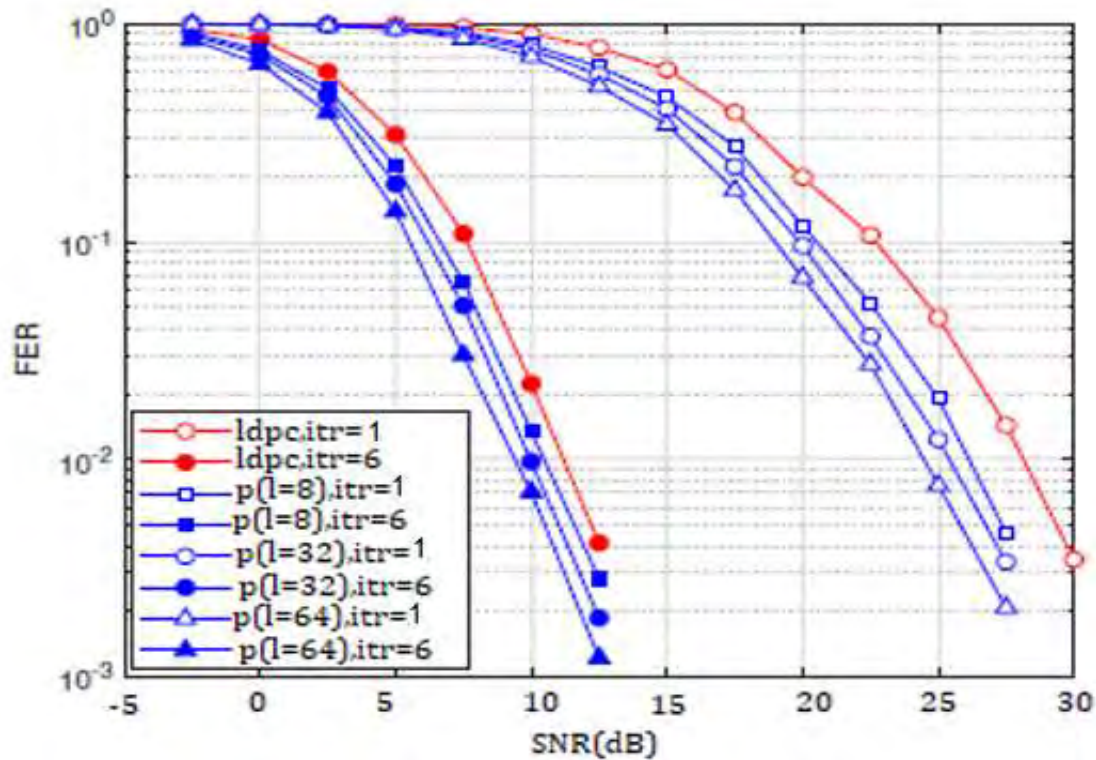


FIGURE 4.11 : Performances FER pour un système MIMO massif à codage polaire et LDPC par rapport au SNR avec $N = 256$, $R = 0.5$ et modulation QPSK.

Les résultats montrent une nette différence si on augmente le nombre d'itération, par exemple pour le code LDPC en augmentant le nombre d'itération on obtient un gain de 17.5dB même chose pour les codes polaires si on prend par exemple le code polaires à $L=8$ et $L=64$ on obtient respectivement un gain de 15 et 15.3dB, ces résultats montrent qu'une augmentation du nombre d'itération démontre le SNR ce qui rend plus performant le code cela confirme les résultats obtenus par [61] qui a montré que plus le nombre d'itération augmente plus le système est performant.

4.1.10 Conclusion.

Dans ce chapitre, nous avons montré en premier lieu la construction des Codes Polaires, l'algorithme de décodage SC, a été détaillé car beaucoup de travaux en découlent, suite nous avons montré, la performance des codes polaires par rapport au code LDPC et la dernière partie expliquée l'évolution des décodeurs SC vers le SCL et le CA-SCL en montrant les limites en terme de performance et de complexité. Les codes polaires ont

montrait des avantages par rapport à leur simple décodeur et de faible complexité. Nous avons montré que les performances des systèmes MIMO massifs son proportionnel au nombre d'antenne de la station de base SB et qu'on peut améliorer les performances des codes polaires et LDPC en augmentant le nombre d'itération. D'après tous ces résultats on peut dire que le système MIMO massif associé au codes polaires son performant et il constitue un candidat potentiel pour la cinquième génération 5G.

CONCLUSION GENERAL

Dans ce mémoire, nous avons étudié la performance du système MIMO massif avec codage/décodage polaire. Nous avons utilisé les codes polaires récemment introduit par **Erdal Arikan**, ces codes sont considérés comme étant la première famille de codes qui atteignent la capacité du canal (limite de Shannon). Leur construction se base sur un phénomène particulier appelé *polarisation de canal*. L'algorithme de décodage utilisé, est appelé SC (*Successive Cancellation*) qui est peu performant pour les codes de tailles faibles. C'est pourquoi tous les travaux qui ont suivi leur découverte se sont penchés sur l'amélioration des performances de décodage. Actuellement le meilleur algorithme de décodage des codes polaires est le SCL (*Successive Cancellation List*) associé par un CRC (Cyclic Redundancy Check) connu sous le nom de CA-SCL. Pour notre modèle de transmission MIMO massif modulé QPSK, le décodeur CA-SCL est utilisé pour le codage/décodage du système. Nous avons montré dans les stimulations que les systèmes MIMO massif son beaucoup plus performant avec le codage/décodage polaire par rapport au code LDPC.

D'après les résultats obtenus, nous pouvons dire que le système MIMO massif avec codage polaire améliore les performances et constitue un candidat incontournable pour la technologie 5G(Cinquième Génération).

En guise de perspective pour nos travail de recherche dans le futur, nous voulons nous intéresser a amélioré les contaminations des pilote et centraliser notre système par rapport au Beamforming c'est-à-dire faire des transmissions guidé par des pilotes.

REFERENCES BIBLIOGRAPHIQUES

- [1] SHANNON, Claude Elwood. A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review, 2001, vol. 5, no 1, p. 3-55.
- [2] COVER, Thomas M. et THOMAS, Joy A. Elements of information theory. John Wiley & Sons, 2012.
- [3] Erdal Arıkan. Channel polarization : A method for constructing capacityachieving codes for symmetric binary-input memoryless channels. IEEE Transactions on Information Theory, 55(7) :3051–3073, July 2009. arXiv : 0807.3917.
- [4] MACKAY, David JC. Information theory, inference and learning algorithms. Cambridge university press, 2003.
- [6] CALDERBANK, A. Robert. The art of signaling : Fifty years of coding theory. IEEE Transactions on Information Theory, 1998, vol. 44, no 6, p. 2561-2595.
- [8] REED, Irving S. et SOLOMON, Gustave. Polynomial codes over certain finite fields. Journal of the society for industrial and applied mathematics, 1960, vol. 8, no 2, p. 300-304.
- [7] BERLEKAMP, Elwyn R., MCELIECE, Robert J., et VAN TILBORG, Henk CA. On the inherent intractability of certain coding problems. IEEE Transactions on Information Theory, 1978, vol. 24, no 3, p. 384-386.
- [9] JOHNSON, Sarah J. Iterative error correction : turbo, low-density parity-check and repeat accumulate codes. Cambridge University Press, 2009.
- [10] HAYKIN, Simon. Communication systems. John Wiley & Sons, 2008.
- [11] SUGIYAMA, Yasuo, KASAHARA, Masao, HIRASAWA, Shigeichi, et al. A method for solving key equation for decoding Goppa codes. Information and Control, 1975, vol. 27, no 1, p. 87-99.
- [12] MASSEY, James. Shift-register synthesis and BCH decoding. IEEE transactions on Information Theory, 1969, vol. 15, no 1, p. 122-127.
- [13] BERLEKAMP, Elwyn R. Coding theory and the Mathieu groups. Information and Control, 1971, vol. 18, no 1, p. 40-64.

- [14] REED, Irving S. et SOLOMON, Gustave. Polynomial codes over certain finite fields. *Journal of the society for industrial and applied mathematics*, 1960, vol. 8, no 2, p. 300-304.
- [16] HOSSAIN, Ekram et HASAN, Monowar. 5G cellular : key enabling technologies and research challenges. *IEEE Instrumentation & Measurement Magazine*, 2015, vol. 18, no 3, p. 11-21.
- [15.58] WANG, Yuanye et PEDERSEN, Klaus I. Performance analysis of enhanced inter-cell interference coordination in LTE-Advanced heterogeneous networks. In : *Vehicular Technology Conference (VTC Spring)*, 2012 IEEE 75th. IEEE, 2012. p. 1-5.
- [17] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE International Symposium on Information Theory Proceedings*, pp. 1–5, 2011.
- [18] I. Tal and A. Vardy, "How to Construct Polar Codes," *IEEE Transactions on Information Theory*, vol. 59, no.10, pp. 6562-6582, October 2013.
- [19] K. Niu and K. Chen, "CRC-aided decoding of polar codes," *IEEE Communications Letters*, vol. 16, no. 10, pp. 1668–1671, 2012.
- [20] O. Iscan, D. Lentner, and W.Xu, "A Comparison of Channel Coding Schemes for 5G Short Message Transmission," *GLOBECOM Workshops*, IEEE, pp. 1–6, 2017.
- [21] D. Millar, K. Kojima, K. Sugihara, K. Parsons, S. Draper, and W. Matsumoto, "Bit Interleaved Polar-Coded Modulation for Low-Latency Short-Block Transmission," *Optical Fiber Communications Conference and Exhibition*, IEEE, 2017.
- [22] J. Guo, Z. Shi, Z. Liu, Z. Liu, Z. Zhang, and Q. Liu, "Multi-CRC Polar Codes and Their Applications," *IEEE Communications Letters*, vol. 10, no. 2, pp. 212–215, February 2016.
- [27] H. Zhou, C. Zhang, W. Song, S. Xu, and X. You, "Segmented CRCAided SC List Polar Decoding," *IEEE Vehicular Technology Conference*, pp. 1–5, 2016.
- [23] I. Tal and A. Vardy, "List decoding of polar codes," in *Proceedings IEEE Transactions on Inform*
- [24] A. Bouacha, F. Debbat, F.T. Bendimerad, —Modified blind beamforming algorithm for smart antennal, *Journal of Radio Electronics*, n° 1, 2008.
- [25] S.A. Djennas, F.T. Bendimerad, S.M. Djennas, «A new technique for planar array antennas synthesis», *AEUE-International Journal of Electronic and Communications*, ISSN 1434-8411, vol. 63, n° 6, pp. 859-870, 2009.
- [26] B. Benadda, F.T. Bendimerad, —Quadratic error optimization algorithm applied to 3D

space distributed array sensors], *Journal of Applied Sciences Research*, 5(10), pp. 1320-1324, 2009. *ation Theory*, vol. 61, pp. 2213-2226, May, 2015.

[32] T. Marzetta, Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no.11, pp. 3590-3600, Sep. 2010.

[33] M. Jain, J. Choi, T. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, and P. Sinha, "Practical, real-time, full duplex wireless," in *Proc.Int. Conf. On Mobile Computing and Networking (ACM)*, New York, NY, USA, 2011, pp. 301-312.

[34] L. Gille, J.-F. Marchandise, "La dynamique d'Internet. Prospective 2030", (Etude réalisée pour le Commissariat général à la stratégie et à la prospective), Télécom ParisTech, Fondation Internet Nouvelle Génération, 2013.

[35] Hien Quoc Ngo, E. Larsson et T. Marzetta, "Energy and Spectral Efficiency of Very Large Multiuser MIMO Systems", *IEEE Transactions on Communications*, vol. 61, no. 4, pp. 1436-1449, 2013.

[36] K. Truong et R. Heath, "Effects of channel aging in massive MIMO systems", *Journal of Communications and Networks*, vol. 15, no. 4, pp. 338-351, 2013.

[37.20] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wirel. Commun.*, vol. 9, no. 11, pp. 3590–3600, 2010.

[38.5] J. Hoydis, S. Ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 160–171, 2013.

[39.21] R. R. Müller, L. Cottatellucci, and M. Vehkaperä, "Blind pilot decontamination," *IEEE J. Sel. Top. Signal Process.*, vol. 8, no. 5, pp. 773–786, 2014.

[40] and Y. L. Haifan Yin, David Gesbert Fellow, IEEE, Miltiades Filippou, "A Coordinated Approach to Channel Estimation in Large-scale Multiple-antenna Systems," *IEEE J. Sel. Top. Signal Process.*, vol. 4553, no. c, pp. 1–1, 2012.

[41] H. Q. Ngo and E. G. Larsson, "EVD-based channel estimation in multicell multiuser MIMO systems with very large antenna arrays," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, no. 1, pp. 3249–3252

[42.6] J. Jose, A. Ashikhmin, T. L. Marzetta, and S. Vishwanath, "Pilot Contamination and Precoding in Multi-Cell TDD Systems," pp. 1–23, 2009.

- [43] B. Zhang et al., “A 5G trial of polar code,” GLOBECOM, 2016.
- [44] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of things for smart cities,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
- [45] T. Koike-Akino et al., “Bit-interleaved polar-coded OFDM for lowlatency M2M wireless communications,” *ICC*, 2017.
- [46] I. Tal and A. Vardy, “List decoding of polar codes,” *IEEE Trans. IT*, vol. 61, no. 5, pp. 2213–2226, Mar. 2015.
- [47] B. Li, H. Shen, and D. Tse, “An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check,” *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044–2047, Dec. 2012.
- [48] C. Cao, Z. Fei, J. Yuan, and J. Kuang, “Low complexity list successive cancellation decoding of polar codes,” *IET Commun.*, vol. 8, Nov. 2014.
- [49] Z. Zhang, L. Zhang, X. Wang, C. Zhong, and H. V. Poor, “A split reduced successive cancellation list decoder for polar codes,” *IEEE JSAC*, vol. 34, no. 2, pp. 292–302, Feb. 2015.
- [50] X. Liang, J. Yang, C. Zhang, W. Song, and X. You, “Hardware efficient and low-latency CA-SCL decoder based on distributed sorting,” GLOBECOM, 2016.
- [51] W. Song, H. Zhou, Y. Zhao, S. Zhang, X. You, and C. Zhang, “Lowcomplexity segmented CRC-aided SC stack decoder for polar codes,” *Asilomar Conf. Signals, Systems and Computers*, pp. 1189–1193, 2016.
- [52] P. Giard, G. Sarkis, C. Leroux, C. Thibeault, and W. J. Gross, “Lowlatency software polar decoders,” *J. Sig. Proc. Syst.*, pp. 1–5, July 2016.
- [53] <https://ieeexplore.ieee.org/document/8292320>
- [54] M. El-Khamy, H. Mahadavifar, G. Feygin, J. Lee, and I. Kang, “Relaxed polar codes,” *IEEE Trans. IT*, vol. 63, no. 4, pp. 1986–2000, Apr. 2017.
- [55] <https://ieeexplore.ieee.org/document/7937785>
- [56] I. Tal and A. Vardy, “List decoding of polar codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.
- [57] B. Li, H. Shen, and D. Tse. (Sep. 2013). “Parallel decoders of polar codes.” [Online]. Available: <https://arxiv.org/abs/1309.1026>

- [58.11] E. Arıkan, "Channel polarization: A method for constructing capacityachieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.
- [59] P. Giard, G. Sarkis, C. Leroux, C. Thibeault, and W. J. Gross, "Lowlatency software polar decoders," *J. Sig. Proc. Syst.*, pp. 1–5, July 2016.
- [60] G. Sarkis, I. Tal, P. Giard, A. Vardy, C. Thibeault, and W. J. Gross, "Flexible and low complexity encoding and decoding of systematic polar codes," *IEEE Trans. Commun.*, vol. 64, no. 7, pp. 2732–2745, 2016.
- [61] <https://ieeexplore.ieee.org/document/8891345>
- [62] R. M. Pyndiah, "Near-optimum decoding of product codes: Block turbo codes," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1003–1010, 2002.

