

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

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

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Filière Télécommunication
Spécialité Réseaux & Télécoms

Présenté par

Samadi Taieb Essedik

&

Brahmi Mohamed Naime

Deep Learning pour l'estimation et l'adaptation de la modulation dans les systèmes OFDM

Proposé par :

Dr. AIT SAADI Hocine

Année Universitaire 2020-2021

REMERCIEMENTS

Avant de vous convier à la présentation de ce travail, l'opportunité nous est donnée de témoigner notre gratitude et notre reconnaissance à toutes les personnes qui par leur aide et leurs encouragements nous ont permis de réaliser ce mémoire.

*On tient à exprimer notre immense reconnaissance à notre promoteur Monsieur **Ait SAADI Hocine** pour avoir accepté de nous encadrer et nous prodiguer ses conseils Pertinents et ses critiques constructives et nous tenons à saluer en lui sa patience Intarissable sans laquelle notre travail n'aurait jamais vu le jour.*

*On tien nous remercîment à Madame **Amirouche Nesrine**.*

Un grand merci aussi à tous les membres de ce prestigieux et distingué jury qui nous ont fait l'honneur

D'accepter d'évaluer et de juger notre travail.

Nos remerciements s'adressent à tous les enseignants qui ont contribué à notre formation durant nos années d'études à cette université.

Nous tenons à remercier tous nos amis et collègues pour leur soutien moral tout au long de la préparation de ce mémoire.

Sans oublier en fin de rendre un grand hommage à nos parents et frère(s) et Sœur(s) qui nous ont encouragé et aidé par tous leurs moyens pour que l'on puisse

Terminer notre projet.

ملخص:

في هذا العمل استخدمنا ofdm متعدد الناقل ، في وجود قناة انتقائية تردد ، وبالتالي تقدير والكشف عن رمز على أساس التعلم العميق. في أطروحتنا ، ركزنا على المقارنة بين طرق التقدير الكلاسيكية (LS) ، (MMSE) وطريقة التعلم العميق باستخدام الشبكات العصبية العميقة. (DNNs) هذا الأخير يعطي أداء أفضل. أداء التعلم العميق (DNN) في SER هو الأفضل مقارنة بالطرق التقليدية (LS)MMSE

كلمات المفاتيح: OFDM ;DNN ;LS ; MMSE ;SER

Résumé :

Dans ce travail nous avons utilisé la modulation multi-porteuse OFDM, en présence d'un canal sélectif en fréquence, d'où l'estimation et la détection de symbole basé sur l'apprentissage en profond. Dans notre mémoire, nous nous sommes intéressés à la comparaison entre les méthodes d'estimation classiques (LS ,MMSE) et la méthode de l'apprentissage profond en utilisons les réseaux de neurones profonds (DNN). Ce dernier donne de meilleures performances. Les performances d'apprentissage profond (DNN) en SER sont les meilleures par rapport au méthodes classiques (LS,MMSE).

Mots clés : OFDM ;DNN ;LS ; MMSE ;SER.

Abstract :In this work we used OFDM multi-carrier modulation, in the presence of a frequency selective channel, hence the estimation and detection of symbol based on deep learning. In our thesis, we focused on the comparison between classical estimation methods (LS, MMSE) and the deep learning method using deep neural networks (DNNs). The latter gives better performance. Deep learning performance (DNN) in SER is the best compared to conventional methods (LS,MMSE).

Keywords :OFDM;DNN ; LS ; MMSE ; SER.

Liste des abréviations

ANN	Artificial Neural Network
AWGN	Additive White Gaussian Noise.
BER	Bit Error Rate.
CAN	Convertisseur Analogique Numérique.
CP	Cyclic Prefix
CSI	Information State Channel
CNN	Convolutif Neural Network
DNN	Deep Neural Network
DFT	Discrete Fourier Transform.
FFT	Fast Fourier Transform.
IA	Intelligence Artificiel
IES	Interférences Entre Symboles.
IEP	Interférences Entre Porteuses.
IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform.
OFDM	Orthogonal Frequency Division Multiplexing.
LS	Least Square
MLP	Multi-Layer Perceptron
MMSE	Minimum Mean Square Error.
MRF	MultiplexageRéparationFrequence
MSE	Mean Square Error.
NLOS	No Line of Sight
Q-PSK	Quadrature Phase Shift Keying.
QAM	Quadrature Amplitude Modulation.
RNN	Recurrent Neuronal Network
SER	Symbol Error Rate.
SNR	Signal to Noise Ratio.

List des notations

L	Le nombre de couches.
θ	Le poids du réseau neuronal.
$\hat{X}(k)$	La valeur prédit.
$X(k)$	La valeur d'entre.
z	Cascade de transformation des données d'entrée.
x	Scalaire.
x^*	Conjugué de x .
$x_{k,m}$	Symbole Deux facteurs.
H	La réponse fréquentielle du canal.
$E\{ \cdot \}$	Esperance mathématique.
E_b/N_0	Rapport entre énergie par bit et densité spectrale du bruit.
\otimes	Produit de convolution circulaire.
T_c	Temps de cohérence.
T_s	Temps symbole.
τ_m	L'étalement temporel maximal.
σ_n^2	Variance du bruit.
(P)	Les pilotes.
R	Matrice de covariance.
T	La Période.
Δf	Largeur de bande des sous canaux.
B	Largeur de bande utilisée.
B_c	Bande de cohérence.
N	Nombre de sous-porteuses.

Table des matières

Introduction générale

Chapitre I :Généralité sur l'apprentissage profond

I.1. Introduction	5
I.2.Intelligence artificielle (IA).....	5
I.2.1.Histoire de L'IA :	5
I.2.2.Le BUT d'IA	8
I.2.3.Domaines de l'IA :	8
a. Le calcul formel :	8
b. La représentation des connaissances :	8
c. Raisonnement et résolution de problème :	8
d. Apprentissage automatique.....	9
e. Réseaux neuronaux.....	9
I.2.4.Utilisation de l'IA.....	9
a. Recherche en ligne	9
b. Achats en ligne et publicité	9
c. Assistants numériques personnels.....	9
d. Traduction automatique	10
e. Maisons et villes et infrastructures connectées	10
f. Les voitures	10
g. Cyber sécurité.....	10
h. L'IA contre la Covid-19	10
i. La lutte contre les fausses informations	10
j. Santé.....	11

Table des matières

k. Transport	11
l. Alimentation et agriculture	11
m. Administration publique et services.....	11
I.3.Machine Learning	11
I.3.1.Définition du machine Learning.....	11
I.3.2.Types d'apprentissage.....	12
a. L'apprentissage supervisé	13
b. L'apprentissage non supervisé.....	13
c. Régression et classification	14
d. L'apprentissage semi-supervisé	15
e. L'apprentissage par renforcement.....	15
I.3.3.Applications du machine Learning.....	15
I.4.Deep learning.....	16
I.4.1.Développement de deep Learning.....	16
I.4.2.Définition de Deep Learning	17
I.4.3.Domain application de Deep Learning.....	18
I.4.4.Les réseaux de neurones utilisés en deep Learning	19
a. Neurone Formel	21
b. Neurones multicouches	21
c. Réseau de neurone convolutifs.....	22
d. Réseau de neurone récurrent :	22
e. Réseau de neurone profond	22
I.5.Conclusion.....	23

Chapitre II: Système OFDM

II.1. Contexte.....	25
II.2. Canal radio-Mobile.....	26
II.2.1. Caractéristique de canal radio mobile	26
II.3. System OFDM (Orthogonal Frequency Division Multiplexing).....	28
II.3.1. Introduction.....	28
II.3.2. Historique	29
II.3.3. Définition de l'OFDM	29
II.3.4. Problème d'interférences :	30
a. Interférence entre symboles (IES).....	30
b. Interférence entre porteuse (IEP)	30
II.3.5. Notion d'orthogonalité	31
II.3.6. Préfixe cyclique et intervalle de garde.....	32
II.3.7. Modèles mathématiques de l'OFDM	34
II.3.8. Modèle discret de l'OFDM	34
II.3.9. La chaîne de transmission OFDM.....	35
II.3.10. Les importants Paramètres de l'OFDM.....	36
II.3.11. Les avantages de la technique de modulation OFDM	38
a. Immunité à la décoloration sélective de fréquence	38
b. Résilience aux interférences	38
c. Efficacité spectrale	38
d. Efficacité Contre Le Bruit Impulsif.....	39
II.3.12. Inconvénients de l'OFDM :	39
a. Rapport de Puissance de Crête à Moyenne (PAPR).....	39

b. Sensibilité aux erreurs de fréquence	39
II.4. Conclusion.....	40
Chapitre III :l'estimation et la détection	
III.1. Introduction	42
III.2. Méthode d'apprentissage.....	42
III.2.1. Fonction activation.....	43
a. Relu.....	43
b. Sigmoidé.....	44
III.3. Architecture de system.....	45
III.4. Estimation et détection basées sur l'apprentissage profond.....	46
III.4.1. Principe de fonctionnement	46
A. Architecture spécifique au réseau	47
B. Progrès de la formation	47
III.4.2. Détails d'implémentation.....	57
A. Système OFDM.....	57
B. Données pilotes.....	57
III.5. Les critères d'estimation.....	59
III.5.1. L'estimateur LS.....	59
III.5.2. L'estimateur MMSE.....	60
III.6. Conclusion.....	62
Chapitre IV : Paramètres et résultats des simulations	
IV.1. Introduction	64
IV.2. Processus global	64
IV.3. Paramètres de la simulation.....	65

Table des matières

IV.4. Le système OFDM	65
IV.5. Estimation du canal	67
IV.6. Critères d'évaluation	68
IV.7. Arrangement des pilotes	68
IV.8. Paramètre de formation	68
IV.9. Résultats des simulations	69
IV.9.1. Impact de CP	69
IV.9.2. Impact des numéros pilotes	70
IV.9.3. Comparaison entre les estimateurs a 64 et 8 pilotes	72
IV.10. Conclusion	72
Conclusion générale	73
Bibliographie	74

Liste des figures :

Figure I.1 : Schéma des différents cas d'utilisation pour un type d'entraînement donné	12
Figure I.2: Apprentissage supervise et non supervisé	14
Figure I.3 : Illustration de la différence entre classification linéaire et régression linéaire ..	14
Figure I.4 : Le scénario typique d'apprentissage par renforcement	15
Figure I.5 : Le Deep Learning est une sous-discipline du Machine Learning	18
Figure I.6 : réseaux de nuerons	19
Figure I.7 : Différence entre un réseau de neurone simple et un réseau profond.....	20
Figure I.8 : Corrélacion entre un neurone humain et un réseau neuronal artificiel	21
Figure I.9: construction d'un modèle de réseau profond (DNN)	23
Figure II.1: Propagation radio-mobile [20]	27
Figure II.2 : Spectre d'un ensemble orthogonal	32
Figure II.3 : Exemple d'intervalle de garde	33
Figure II.4 : Signaux arrivant par différents trajets.	34
Figure II.5 :Système OFDM en bande de base à temps continu	34
Figure II.6 : Système OFDM en bande de base à temps discret.....	35
Figure II.7 : Schéma d'un système OFDM en bande de base.	36
Figure II.8 : Le signal multi carrier peut être calculé comme dans l'équation 2.2 [22]	37
Figure II.9 : Schéma fonctionnel de la modulation OFDM et de la démodulation.....	38
Figure III.1 : Modèle de DNN	43
Figure III.2 : Fonction Relu	44
Figure III.3 : fonction Sigmoidé.....	44
Figure III.4 : Architecture de système	45
Figure III.5 : Réseau neurone qui se compose de deux nœuds pour l'entrée et la sortie et une couche cachée, qui a deux nœuds [30].....	49
Figure III.6 : Former le réseau neuronal à l'aide de l'algorithme de rétro propagation.	50
Figure III.7 : Continuez vers la gauche jusqu'aux nœuds cachés et calculez le delta.....	51
Figure III.8 : Dérivez l'équation pour ajuster le poids	54
Figure III.9 : Dérivez l'équation pour ajuster le poids, encore une fois.	54

Liste des figures

Figure IV.1 : Diagramme de constellation QPSK	66
Figure IV.2 : Courbes SER SANS CP	69
Figure IV.3 : Courbes SER avec 64 pilotes	70
Figure IV.4 : Courbes SER avec 8 pilotes	71
Figure IV.5 : Courbes SER basé sur l'apprentissage profond et les méthodes classiques.....	72

Liste des tableaux :

Tableau IV-1: Paramètres utilisés dans nos simulations.....	67
Tableau IV-2 : Paramètres utilisés dans la formation	68

Introduction générale

Dans les systèmes de communications sans fil, la connaissance de l'environnement de propagation ou encore le canal physique entre une antenne d'émission et une antenne de réception s'avère incontournable. En effet, dans la plupart des cas, le signal reçu est une somme de plusieurs versions retardées du signal émis. On dit alors que le signal émis se propage dans un canal multi-trajets. Ce canal peut engendrer plusieurs types d'évanouissements comme par exemple l'évanouissement sélectif ; c'est-à-dire des variations importantes dans le spectre du canal détériorant le signal. Une solution pour lutter contre ce canal est d'utiliser une modulation multi-porteuses [24].

Un canal est dit sélectif en fréquence lorsqu'il ne se comporte pas identiquement suivant la fréquence du signal. Certaines fréquences seront transmises plus rapidement que d'autres, ou encore seront atténuées plus que d'autres. Le signal sera alors déformé lors de la transmission ; les données seront dispersées dans le temps, pouvant mener à des interférences entre symboles communément appelées IES (Inter Symbole interférences). Ce phénomène de sélectivité en fréquence est aggravé par la présence de trajets multiples pour un même signal transmis. Du fait des nombreuses réflexions que le signal peut subir en environnement urbain, le récepteur recevra une série d'échos d'amplitudes et de retards variables. Cette problématique du canal à trajets multiples est critique dans le cas d'un canal radio-mobile, c'est-à-dire lorsque le récepteur et l'émetteur ne sont pas fixes l'un par rapport à l'autre. Les différents échos et amplitudes variant dans l'espace, varieront également dans le temps dans le cas d'un récepteur mobile.

La modulation multi-porteuses a été introduite à la fin des années 50, sa première utilisation était dans des systèmes de communications hautes fréquences militaires. Quelques années plus tard, elle a été améliorée avec l'apparition du concept des signaux orthogonaux à bande limitée, concept que l'on appellera par la suite "Orthogonal Frequency Division Multiplexing" (OFDM) ou multiplex à division de fréquences orthogonales.

L'estimation de canal peut être effectuée à l'aide de l'insertion de symboles pilotes sur toutes les sous-porteuses d'un symbole OFDM avec une période spécifique, connue sous le nom « estimation du canal par pilotes de type bloc ».

L'estimation du canal se fait généralement sur la réponse fréquentielle si le canal est invariant dans un symbole OFDM, ou sur la réponse impulsionnelle du canal discret équivalent si le canal varie dans un symbole OFDM.

L'estimation des canaux dans les systèmes OFDM a été soigneusement étudiée. Les méthodes d'estimation traditionnelles, moindres carré (LS) et l'erreur quadratique moyenne minimale (MMSE), ont été utilisés et optimisés dans diverses conditions. La méthode d'estimation LS ne nécessite aucune statistique de canal préalable, mais ses performances peuvent être insuffisantes. L'estimation MMSE en général conduit à de bien meilleures performances de détection par l'utilisation des statistiques de second ordre des canaux. Dans notre mémoire, nous introduisons une approche d'apprentissage profond pour estimation de canal et détection de symbole dans un système OFDM.

Notre mémoire comprend 4 différents chapitres dont le contenu est décrit ci-dessous.

Le premier chapitre est introductif, il décrit brièvement l'apprentissage profond. Nous rappelons aussi dans ce chapitre le développement de l'apprentissage profond et les types de réseaux neurones et leur utilisation.

Dans le deuxième chapitre, nous décrivons les principes généraux de la technique de multiplexage par division en fréquences orthogonales (OFDM) utilisée afin de faire face aux évanouissements par trajets multiples, on commence par discuter des principes généraux de la modulation multi-porteuses ensuite on discute de l'orthogonalité et l'utilité de l'intervalle de garde qui permettent de minimiser IES et IEP entre les sous porteuses, à la fin on cite les différents avantages et inconvénients de l'OFDM.

Le chapitre trois, décrit l'étape d'estimation et la détection du symbole dans canal au niveau du récepteur à l'aide des symboles pilotes. Ensuite, nous discutons des différents

critères d'estimation LS et MMSE et la méthode d'apprentissage profond. Ce chapitre se termine par les détails de notre simulation.

Le quatrième et dernier chapitre est consacré aux différentes étapes de la simulation, c'est-à-dire : paramètres, résultats et interprétation.

Nous terminerons ce manuscrit par une conclusion générale dans laquelle nous présentons les principaux points étudiés ainsi que les résultats obtenus à l'issue de ce travail.

Chapitre I : Généralité sur l'apprentissage profond

I.1. Introduction

Lorsqu'on parle à certaines personnes de l'apprentissage profond (Deep Learning), elles pensent à un mystère sombre et profond mais en réalité ce n'est pas vraiment un mystère, chaque fois qu'on utilise des applications en ligne et même lors d'un achat. On est entouré par le Deep Learning et on ne se rend même pas compte

En réalité, l'apprentissage profond est un sous-ensemble de l'apprentissage automatique, qui à son tour est un sous-ensemble de l'intelligence artificielle (IA).

Le premier objectif de ce chapitre est d'expliquer ce qu'est réellement l'apprentissage profond et comment il s'applique au monde d'aujourd'hui.

I.2. Intelligence artificielle (IA)

I.2.1. Histoire de L'IA :

Entre 1940 et 1955 : ont été fortement marquée par la conjonction de développements technologiques Pendant cette période furent menés les premiers travaux qui peuvent être considérés comme les débuts de l'intelligence artificielle [1].

Les travaux de McCulloch et Pitts qui ont introduit en 1943 un modèle de neurones artificiels. Quelques années plus tard, Hebb proposa une loi pour modifier des connections entre neurones, Minsky et Edmonds construisirent les premiers réseaux de neurones. Ce fut aussi pendant cette période que Turing publia son célèbre article dans lequel introduit le test de Turing [1].

La naissance de l'intelligence artificielle en 1956 c'est durant cette année qu'un groupe d'informaticiens intéressé par l'étude de l'intelligence se réunirent pour une conférence sur ce sujet. Cette conférence dura deux mois, et permit de poser les fondements de l'intelligence artificielle ce nom qui choisit à l'issue de cette assemblée [1].

Dans les années entre 1952-1969 c'est une période très active pour le domaine de l'intelligence artificielle. De grands nombre de programmes seront développés pour résoudre des problèmes d'une grande diversité. Les programmes "Logic Theorist" fait par Allen Newell

Chapitre I : Généralité sur l'apprentissage profond

et Herbert Simon et Geometry "ThéoremProver" par Herbet Gelernter furent en mesure de prouver certains théorèmes mathématiques.

Le General Problem Solver ou bien un résolveur de problèmes de Newell et Simon réussissait quant à lui à résoudre des puzzles simples avec un raisonnement ressemblant au raisonnement humain. Samuel créa un programme qui joue à un niveau moyen au jeu de dames. Des étudiants de Marvin Minsky travaillèrent sur les petits problèmes dans le cas de microworlds tels que les problèmes d'analogie ce dernier donne une naissance au programme ANALOGY, ou encore les manipulations de cubes "blocks world" avec l'idée d'augmenter la complexité, petit à petit pour développer des agents intelligents. McCarthy publia un article qui est devenu célèbre dans lequel il traite des programmes qui ont du sens commun. La recherche également poursuivie sur les réseaux de neurone. Ce fut aussi l'ère du Shakey, le premier robot à être capable de raisonner ses propres actions [1].

Les Premières déceptions ont été durant la période 1966-1973, durant ces années devient plus en plus évident que les prédictions faites par les chercheurs en intelligence artificielle avaient été beaucoup trop optimistes. Ce fut le cas de l'exemple pour la traduction automatique. Les chercheurs n'avaient compté que cinq ans pour réaliser un traducteur automatique, mais ils se sont vite rendu compte que leur approche purement syntaxique n'était pas suffisante. Cet échec a pour conséquence l'annulation de tout le financement du gouvernement américain en 1966 pour les projets de traduction automatique. De grandes déceptions se produisirent également lorsque les chercheurs en IA essayèrent d'appliquer leurs algorithmes aux problèmes de grande taille, par manque de mémoire et de puissance de calcul ils découvrirent alors qu'ils ne fonctionnaient pas, Ce fut une des critiques dédié à l'IA dans le rapport de 1973 de Lighthill , qui provoqua l'arrêt du financement de la quasi-totalité des projets en intelligence artificielle de la Grande Bretagne. Et comme si cela ne suffisait pas, Minsky et Papert prouvèrent dans leur livre "Perceptrons" 1969 que les réseaux de neurones de cette époque ne pouvaient pas calculer certaines fonctions pourtant très simples, ce qui mit en cause toute les recherches dans l'apprentissage automatique, entraînant une crise dans cette branche de l'intelligence artificielle [1].

Chapitre I : Généralité sur l'apprentissage profond

Dans les années 70 apparaissent nouveaux modèles de représentation et de traitement des connaissances : Systèmes experts. Le premier système expert a été créé en 1969, appelé DENDRAL, pour la tâche spécialisée consistant à déterminer la structure moléculaire d'une molécule étant donné sa formule et les résultats de sa spectrométrie de masse. DENDRAL, comme tous les systèmes experts, est basé sur un grand nombre de règles heuristiques (désigne une méthode de résolution d'un problème non pas basés sur une analyse détaillée mais par des solutions déjà existantes apporté à des groupes de problèmes bien connus) élaborées par des experts et des chercheurs humains. Après le succès du DENDRAL, d'autres systèmes d'experts furent créés, par exemple le système MYCIN, qui réalisait un diagnostic des infections sanguines. Avec plus de 450 règles, MYCIN réussissait à diagnostiquer à un niveau proche des experts humains et considérablement meilleur que celui des jeunes docteurs.

Les Années 80 la phase d'industrialisation d'IA : au début de ces années, l'entreprise DEC commença à utiliser un système expert d'aide à la configuration de systèmes informatiques, ce qui leur permit d'économiser des dizaines de millions de dollars chaque année. De grandes sociétés commencèrent alors à s'intéresser à l'IA et à former leurs propres équipes de recherche. La Grande Bretagne relança son programme de financement et les Etats-Unis et le Japon financèrent de gros projets en IA.

Le retour des réseaux de neurones en 1986 : la règle d'apprentissage "back-propagation" a été mise au point par quatre groupes de chercheurs indépendamment, les un des autres règles permit le développement de réseaux de neurones capables d'apprendre des fonctions très complexes (étonnamment cette règle avait déjà été proposée en 1969, mais n'avait eu aucun écho dans la communauté scientifique). Depuis, l'apprentissage automatique est devenu l'un des domaines les plus actifs de l'intelligence artificielle, et a été appliqué avec succès à de nombreux problèmes pratiques par exemple la fouille de données.

L'IA Moderne en 1987 : L'intelligence artificielle est devenue au fil du temps une matière scientifique de plus en plus rigoureuse et formelle. La plupart des approches étudiées aujourd'hui sont basées sur des théories mathématiques, des études

Chapitre I : Généralité sur l'apprentissage profond

expérimentales plutôt que sur l'intuition, et sont appliquées plus souvent aux problèmes du monde réel.

En 1997, le système expert d'IBM « Deep Blue » gagne au jeu d'échec contre le champion du monde Garry Kasparov. Deep Blue a été conçu en s'appuyant sur un algorithme systématique de force brute, c'est à dire que tous les coups envisageables étaient évalués et pondérés. Ce phénomène a été marqué par la défaite de l'humain face à la machine et resta très symbolique dans l'histoire, bien que DeepBlue ne fût en réalité parvenu à ne traiter qu'un périmètre bien limité qui est celui des règles du jeu des échecs [1].

Depuis 2010 est apparu un nouvel essor dû à la collecte de données massives et aux nouvelles puissances de calcul [1].

I.2.2. Le BUT d'IA

L'IA est le domaine de l'informatique visant à concevoir des systèmes informatiques intelligents, c'est-à-dire des systèmes qui ont des caractéristiques d'intelligence, comme, dans le comportement humain, et qui ont des capacités d'apprentissage, de raisonnement et de résolution de problèmes.

I.2.3. Domaines de l'IA :

a. Le calcul formel :

Est le domaine de l'informatique et des mathématiques qui s'intéresse aux algorithmes opérationnels sur des objets de nature mathématique par le biais de représentations finies et exactes [2].

b. La représentation des connaissances :

Indique un ensemble d'outils et de procédés destinés à représenter et à organiser le savoir humain pour l'utiliser et le partager [2].

c. Raisonnement et résolution de problème :

La résolution automatique de ces problèmes consiste à essayer successivement toutes les possibilités jusqu'à trouver une solution. Cette recherche est fortement

Chapitre I : Généralité sur l'apprentissage profond

combinatoire : à partir d'un état donné, on peut généralement effectuer plusieurs actions élémentaires différentes qui débouchent sur des états différents.

d. Apprentissage automatique

En anglais "machine Learning", est un domaine d'intelligence artificielle (IA) qui permet à un système d'apprendre à partir des données et non à l'aide d'une programmation explicite.

e. Réseaux neuronaux

Un réseau de neurones formels est un modèle primitif du cerveau humain. Une cellule neuronale possède une seule sortie et des entrées reliées à d'autres neurones. Ces réseaux partagent des propriétés très importantes avec le cerveau humain. Cela exige une programmation non explicite et de la répartition d'information sur l'ensemble du réseau.

I.2.4. Utilisation de l'IA

Voici des exemples sur l'utilisation de l'IA :

a. Recherche en ligne

Les moteurs de recherche (google, quora, bing, yahoo!) évoluent qu'ils engrangent un flot massif de données fournit par des utilisateurs, afin de leur procurer des résultats plus pertinents [3].

b. Achats en ligne et publicité

L'IA est fréquemment utilisée lorsqu'on fournit des recommandations personnalisées aux utilisateurs, basé par exemple sur l'achat ou leur historique de recherche ou encore sur leur comportement en ligne [3].

c. Assistants numériques personnels

Les Smartphones emploient l'IA pour proposer un produit aussi pertinent et adapté que possible. Les assistants virtuels désormais partout présents répondent aux questions et fournissent des recommandations et aide à gérer les tâches quotidiennes.

Chapitre I : Généralité sur l'apprentissage profond

d. Traduction automatique

Les logiciels de traduction qui sont basés sur la langue écrite ou parlée [3].

e. Maisons et villes et infrastructures connectées

Certains thermostats analysent nos comportements pour mieux économiser l'énergie, alors que les urbanistes des villes connectées espèrent réduire les bouchons et mieux gérer la circulation [3].

f. Les voitures

Alors que les véhicules autonomes ne sont pas encore devenus la norme, nos voitures utilisent déjà des fonctions de sécurité basées sur l'IA. L'UE a par exemple soutenu le financement de VI-DAS, des capteurs automatiques qui détectent les situations

Potentiellement dangereuses et les accidents [3].
Le fonctionnement des systèmes de navigation repose fortement sur l'IA.

g. Cyber sécurité

Les systèmes dotés de l'IA peuvent aider à détecter et combattre les cybers attaques et autres menaces, en utilisant le flot continu de données pour déceler des figures et remonter à la source des attaques [3].

h. L'IA contre la Covid-19

L'IA permet de lutter contre la Covid-19, en l'employant dans les aéroports lors de la production d'imageries thermiques ainsi que dans d'autres cas de figure. Dans le domaine médical, l'IA permet de détecter l'infection par lorsque les poumons de patients sont soumis à une tomographie par ordinateur. Elle a aussi permis de recueillir des données afin de suivre l'évolution des infections [3].

i. La lutte contre les fausses informations

L'IA permet de détecter les fausses informations et les cas de désinformation en parcourant le contenu des réseaux sociaux pour repérer des mots clés ou des termes alarmistes, en identifiant les sources peuvent être considérées comme étant fiables [3].

j. Santé

Les chercheurs étudient comment employer l'IA pour analyser un très grand nombre de données liées à la santé pour repérer des motifs récurrents qui mèneraient à de nouvelles découvertes et des moyens d'améliorer les diagnostics individuels [3].

k. Transport

L'IA pourrait aider à améliorer la sécurité, la rapidité et l'efficacité du transport ferroviaire en réduisant le frottement des roues, augmentant la vitesse des trajets et en favorisant la conduite autonome [3].

l. Alimentation et agriculture

Par exemple de nombreuses fermes partout dans l'Union Européenne ont déjà recours à l'IA pour contrôler les mouvements, la température corporelle et la consommation alimentaires de leurs animaux.

m. Administration publique et services

En utilisant un large éventail de données et en employant la reconnaissance de formes, l'IA pourrait fournir des alertes précoces dans le cadre de catastrophes naturelles et permettre une meilleure préparation et gestion des retombées.

I.3. Machine Learning

I.3.1. Définition du machine Learning

Le machine learning ou l'apprentissage automatique est un sous ensemble de l'intelligence artificielle qui est axé sur la création de système, qui apprennent ou améliorent les performances en fonction des données qu'ils traitent. L'apprentissage automatique est l'IA sont souvent abordés ensemble, et le terme sont parfois utilisés de manière remplaçable, mais ils ne veulent pas dire la même chose. Une distinction importante est que l'intégralité du machine learning repose sur l'IA.

Si l'apprentissage automatique ne date pas d'hier, sa définition exacte demeure encore confuse pour de nombreuses personnes. Pratiquement il s'agit d'une science

Chapitre I : Généralité sur l'apprentissage profond

moderne qui permet de découvrir des modèles et de réaliser des prédictions à partir de données en se basant sur des statistiques et sur du forage de données et sur la reconnaissance de modèles, sur les analyses prédictives. Les premiers algorithmes sont créés à la fin des années 50. Le plus connu d'entre eux c'est le Perceptron.

I.3.2. Types d'apprentissage

Les algorithmes sont les principes du "machine Learning". En général, deux approches d'algorithmes de l'apprentissage automatique sont utilisées aujourd'hui : l'apprentissage supervisé et l'apprentissage non supervisé.

La différence entre les deux c'est la méthode employée pour traiter les données afin de faire des prédictions.

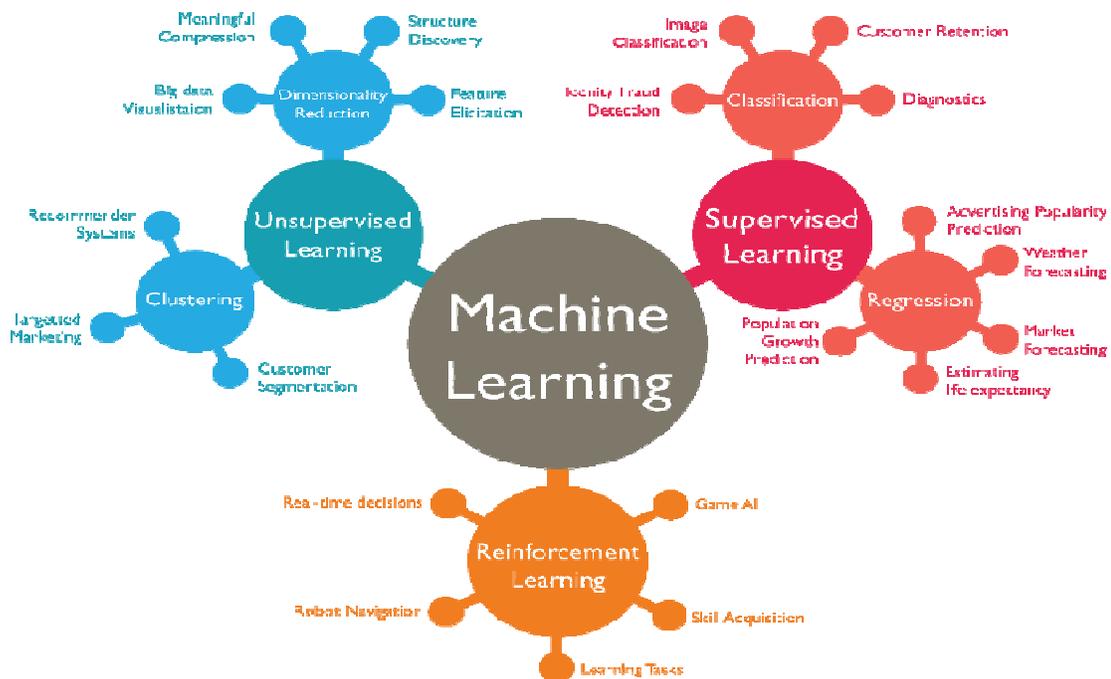


Figure I.1 : Schéma des différents cas d'utilisation pour un type d'entraînement donné

Chapitre I : Généralité sur l'apprentissage profond

a. L'apprentissage supervisé

Ces algorithmes sont les plus couramment utilisés. Avec ce modèle, un data scientist en machine learning non supervisé sert de guide et enseigne à l'algorithme les conclusions qu'il doit tirer. Tout comme un enfant apprend à identifier les fruits en les mémorisant dans un imagier, en apprentissage supervisé, l'algorithme apprend grâce à un jeu de données déjà étiqueté et dont le résultat est prédéfini. Comme exemples de machine learning supervisé, on peut citer des algorithmes tels que la régression linéaire et logistique, la classification en plusieurs catégories et les machines à vecteurs de support [4].

En apprentissage supervisé on reçoit des données d'exemple annotées : $(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_1, y_1), (x_2, y_2), (x_3, y_3), \dots$ et on espère prédire la sortie sur de nouvelles observations : $x^* \rightarrow y^*$ [5].

b. L'apprentissage non supervisé

Le machine learning non supervisé, comme classification des images, ou des méthodes de deeplearning utilise une approche plus indépendante, dans laquelle un ordinateur apprend à identifier des processus et des schémas complexes sans un quelconque guidage humain constant et rigoureux. L'apprentissage non supervisé implique une formation basée sur des données sans étiquette ni résultat spécifique défini. Pour continuer avec l'analogie de l'enseignement scolaire, l'apprentissage non supervisé s'apparente à un enfant qui apprend à identifier un fruit en observant des couleurs et des motifs, plutôt qu'en mémorisant les noms avec l'aide d'un enseignant. L'enfant cherche des similitudes entre les images et les sépare en groupes, en attribuant à chaque groupe sa propre étiquette.

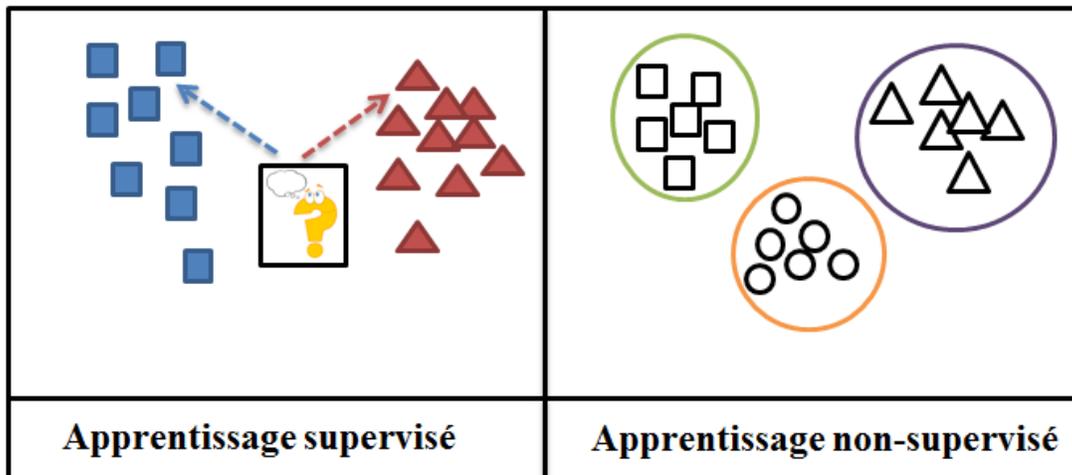


Figure I.2: Apprentissage supervise et non supervise

En apprentissage non supervisé, on reçoit uniquement des observations brutes de variables aléatoires : $x_1, x_2, x_3, x_4, \dots$ et on espère découvrir la relation avec des variables latentes structurelles : $x_i \rightarrow y_i$

c. Régression et classification

Une autre distinction qui aide dans le choix d'un algorithme de machine learning est le type de sortie que l'on attend de notre programme : est-ce une valeur continue (un nombre) ou bien une valeur discrète (une catégorie) ? Le premier cas est appelé une régression, le second une classification.

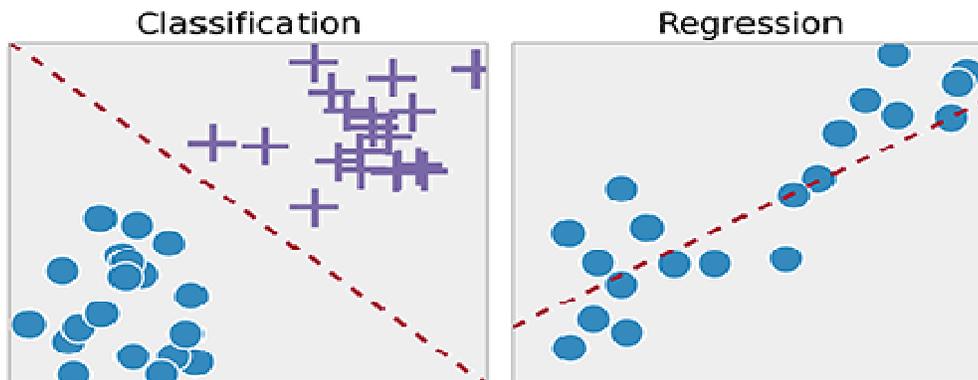


Figure I.3 : Illustration de la différence entre classification linéaire et régression linéaire

Il existe en fait deux autres familles d'algorithmes

Chapitre I : Généralité sur l'apprentissage profond

d. L'apprentissage semi-supervisé

Qui prend en entrée certaines données annotées. Ce sont des méthodes intéressantes qui tirent parti des deux mondes supervisé et non supervisé, mais bien sûr apportent leur part de difficultés.

e. L'apprentissage par renforcement

Qui se base sur un cycle d'expérience récompense et améliore les performances de chaque itération. Une analogie souvent citée est celle du cycle de dopamine : une "bonne" expérience augmente la dopamine et donc augmente la probabilité que l'agent répète l'expérience.

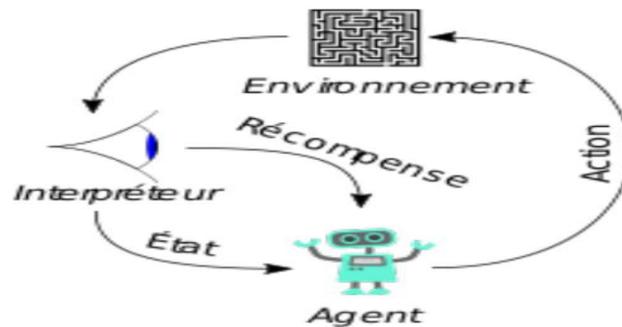


Figure I.4 : Le scénario typique d'apprentissage par renforcement

I.3.3. Applications du machine Learning

Voici quelque application de ML

- * Interfaces cerveau-machine
- * Publicité informatique
- * Financement informatique
- * Vision par ordinateur, y compris la reconnaissance d'objets
- * Détection de fraude sur Internet
- * Perception de la machine
- * Traitement du langage naturel
- * Optimisation et méta heuristique

I.4. Deep learning

I.4.1. Développement de deep Learning

L'intelligence artificielle n'est pas une découverte récente, elle date des années 1950, mais malgré l'évolution de sa structure il reste certain problème tel que le volume de données et la vitesse de calcul. Toutefois, un point pertinent portait sur la façon dont les grands modèles machine Learning ont été traités avec de grandes quantités d'informations, car les ordinateurs n'ont pas pu effectuer une telle action.

Le deuxième problème est la programmation parallèle dans les GPU. Les unités de traitement graphique, qui permettent la réalisation d'opérations mathématiques en parallèle, en particulier celles avec des matrices, qui sont présentes dans les modèles de réseaux artificiels, ont permis l'évolution actuelle « Big Data »

La base d'un réseau neuronal artificiel est un neurone mathématique, également appelé un nœud, basé sur le neurone biologique. Les liens entre ces neurones mathématiques sont liés à ceux des cerveaux biologiques.

A la deuxième moitié des années 80 et le début de la décennie de 90, le développement de la structure des réseaux artificiels se sont produits. Cependant, la quantité de temps et d'informations nécessaires pour obtenir de bons résultats.

Dans les années qui suivirent, John von Neumann imitait les fonctions simples des neurones avec des tubes à vide ou des télégraphes, tandis que Frank Rosenblatt initia le projet Perceptron, analysant le fonctionnement de l'œil d'une mouche. Le résultat de cette recherche était un matériel, qui est le plus ancien réseau neuronal utilisé jusqu'à nos jours. Cependant, le Perceptron est très limité, ce qui a été prouvé par Marvin et Papert [6].

En 1959, Bernard Widrow et Marcian Hoff développent deux modèles appelés «Adaline» et «Madaline». La nomenclature dérive de l'utilisation de plusieurs éléments : ADAPtiveLINear. Adaline a été créé pour identifier les modèles binaires afin de faire des prédictions sur le bit suivant, tandis que "Madaline" a été le premier réseau neuronal

Chapitre I : Généralité sur l'apprentissage profond

appliqué à un problème réel, à l'aide d'un filtre adaptatif. Le système est toujours en cours d'utilisation, mais seulement commercial [7].

En 1985, l'Institut américain de physique a lancé une réunion annuelle appelée "Neural Networks For Computation. En 1986, les médias ont commencé à rapporter les réseaux neuronaux de plusieurs couches, et trois chercheurs ont présenté des idées similaires, appelées réseaux de rétro propagation, car ils distribuent des échecs d'identification de modèles à travers le réseau [8].

Les réseaux hybrides n'avaient que deux couches, tandis que les réseaux de rétro propagation en présentent beaucoup, de sorte que ce réseau conserve les informations plus lentement, parce qu'ils ont besoin de milliers d'itérations pour apprendre [9].

En 1987, la première Conférence internationale sur les réseaux neuronaux de l'Institut de génie électrique et électronique (IEEE).

En 1989, les scientifiques ont créé des algorithmes qui utilisaient des réseaux neuronaux profonds. En 1992, JuyangWengDiulga la méthode Cresceptron pour réaliser la reconnaissance d'objets 3D [6].

En 2000, le terme Deep Learning commence à être diffusé après un article de Geoffrey Hinton et RuslanSalakhutdin[11].

En 2009, l'atelier de traitement des systèmes de réseau neuronal sur l'apprentissage profond pour la reconnaissance vocale a lieu il est vérifié que avec un groupe de donnée étendu, les reseaux de neurones n'ont pas de formation préalable et les taux d'échec chutent de manière significative [12].

En 2015, Facebook utilise le deep Learning pour marquer et reconnaître automatiquement les utilisateurs sur les photos [13].

I.4.2. Définition de DeepLearning

Le Deep Learning, ou apprentissage profond, est l'une des principales technologies de l'intelligence artificielle, leDeep Learning est basé sur des algorithmes, mais aussi des technologies comme la reconnaissance d'images ou la vision robotique.

Chapitre I : Généralité sur l'apprentissage profond

Comme est représenté sur la figure 1.5, pour bien comprendre l'apprentissage profond, on doit commencer par l'extérieur - c'est-à-dire qu'on commence par l'IA, puis on progresse dans l'apprentissage automatique, puis l'apprentissage profond.

L'apprentissage profond est basé sur ce qui a été appelé, par analogie « Les réseaux de neurones artificiels » inspirés par les neurones du cerveau humain. Ils sont constitués de plusieurs neurones artificiels connectés entre eux. C'est le nombre de couches de neurones est élevé, plus le réseau est profond.

Les progrès de l'apprentissage profond ont été possibles notamment grâce à l'augmentation de la puissance des ordinateurs et au développement de grandes bases de données « big data ». Et la flexibilité dans la modélisation des problèmes.

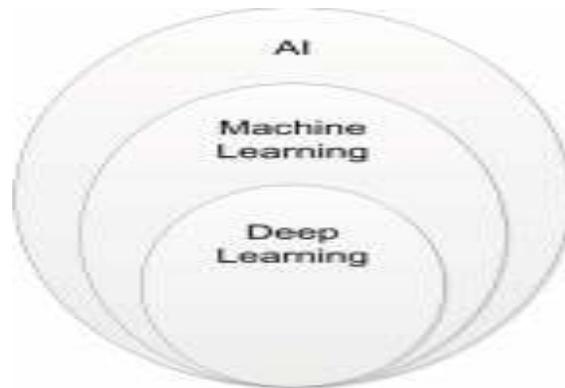


Figure 1.5 : Le Deep Learning est une sous-discipline du Machine Learning

I.4.3. Domain application de Deep Learning

Le Deep Learning travaille sur des systèmes informatiques pour effectuer des tâches telles que la reconnaissance d'image, la voiture autonome, la localisation des points caractéristiques de visage, la bio-informatique, la sécurité, la santé etc....

Dans le domaine des télécommunications on peut travailler avec le deep Learning. Parmi ces applications :

- Optimisation du routage et de la qualité de service en analysant le trafic réseau en temps réel.
- Estimation de canal et la détection de signal dans les System OFDM

Chapitre I : Généralité sur l'apprentissage profond

- Analyser les enregistrements de données d'appels en temps réel pour identifier immédiatement les comportements frauduleux.
- Permettre aux représentants des centres d'appels de modifier immédiatement les plans d'appel des abonnés de manière flexible et rentable.
- Adaptation des campagnes marketing aux clients individuels en utilisant des technologies basées sur la localisation et les réseaux sociaux.
- Utilisation d'informations sur le comportement et l'utilisation des clients pour développer de nouveaux produits et services.
- Utiliser le Deep Learning pour prédire le taux de désabonnement des clients dans un Réseau de télécommunications mobiles
- Utilisation du Deep Learning pour la gestion des réseaux de télécommunications mobiles
- Deep Learning et ses applications au traitement du signal et de l'information (reconnaissance vocale et traitement de la parole et du langage)

I.4.4. Les réseaux de neurones utilisés en deepLearning

Dans le cerveau humain, chaque neurone reçoit environ 100 000 signaux électriques des autres neurones. Chaque neurone en activité peut produire un effet excitant ou inhibiteur sur ceux auxquels il est connecté. Au sein d'un réseau artificiel, le principe est similaire. Les signaux voyagent entre les neurones. Toutefois, au lieu d'un signal électrique, le réseau de neurones assigne un certain poids à différents neurones. Un neurone qui reçoit plus de charge exercera plus d'effet sur les neurones adjacents. La couche finale de neurones émet une réponse à ces signaux.



Figure I.6 : réseaux de neurones

Chapitre I : Généralité sur l'apprentissage profond

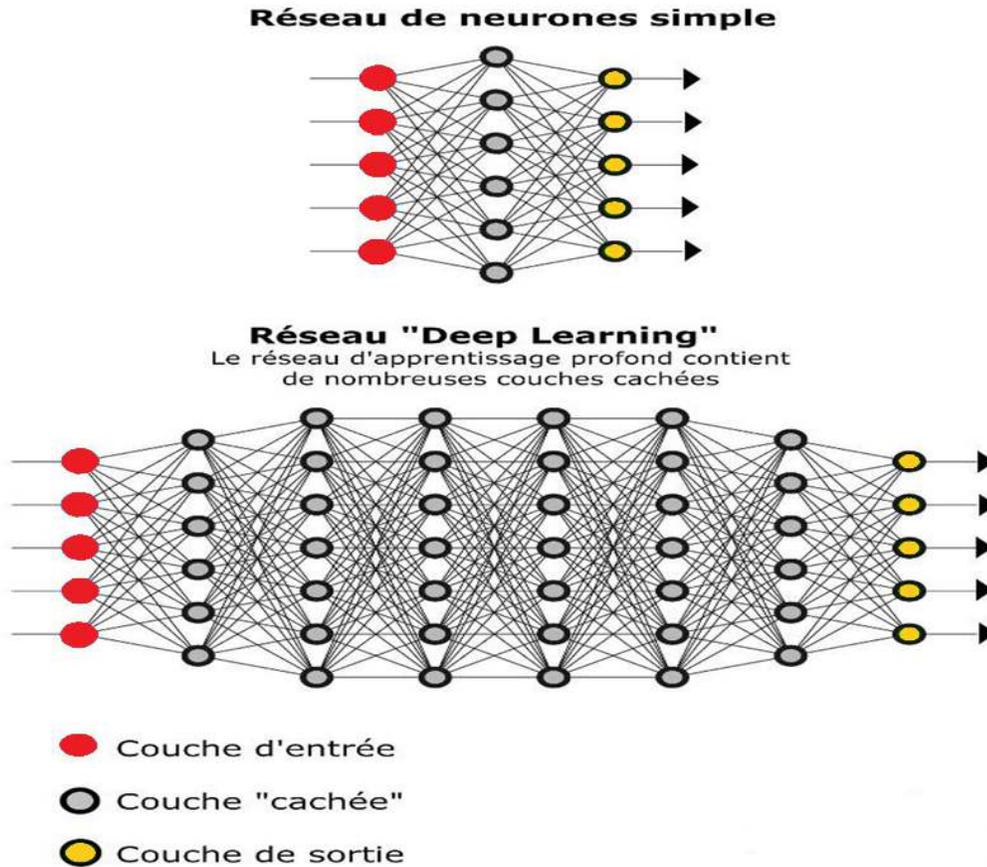


Figure I.7 : Différence entre un réseau de neurone simple et un réseau profond

Exemple : Reconnaissance d'un chat sur une photo

L'algorithme doit être en mesure de distinguer les différents types de chats, et de reconnaître un chat de manière précise quel que soit l'angle sous lequel il est photographié.

Le réseau de neurones doit être entraîné. Pour ce faire, il est nécessaire de compiler un ensemble d'images d'entraînement pour pratiquer le Deep Learning. Cet ensemble va regrouper des milliers de photos de chats différents, mélangés avec des images d'objets qui ne sont pas des chats. Ces images sont ensuite converties en données et transférées sur le réseau. Les neurones artificiels assignent ensuite un poids aux différents éléments. La couche finale de neurones va alors rassembler les différentes informations pour déduire s'il s'agit ou non d'un chat.

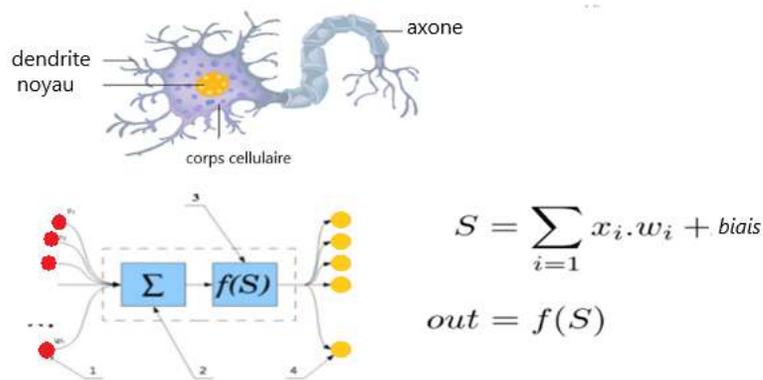


Figure 1.8 : Corrélation entre un neurone humain et un réseau neuronal artificiel

Il est noté que le neurone est formé par les dendrites qui sont les points d'entrée, un noyau qui représente dans les réseaux neuronaux artificiels le noyau de traitement et le point de sortie qui est représenté par l'AXON. Dans les deux systèmes, les informations entrent, sont traitées et changent.

Considérant qu'il s'agit d'une équation mathématique, le neurone reflète la somme des entrées multipliées par les pondérations, et cette valeur passe par une fonction d'activation. Cette somme a été réalisée par McCulloch et Pitts en 1943 [14].

a. Neurone Formel

Le neurone formel est une représentation mathématique et informatiqued'un neurone biologique. Le neurone formel possède plusieurs entrées et unesortie quicorrespondent respectivement aux dendrites et au cône d'émergence du neuronebiologique (l'axone).

b. Neurones multicouches

Le perceptron multicouche (multi layer perceptron MLP) est un classifieur linéaire detype réseau neuronal formel organisé en plusieurs couches au sein desquellesune information circule de la couche d'entrée vers la couche de sortie uniquement : il s'agitdonc d'un réseau de type feedforward (en). Chaque couche est constituée d'un nombrevariable de

Chapitre I : Généralité sur l'apprentissage profond

neurones, les neurones de la couche de sortie correspondant toujours aux sorties du système [15].

c. Réseau de neurone convolutifs

Un réseau de neurone convolutifs (ou réseau de neurones à convolution, ou CNN ou ConvNet) est un type de réseau de neurones artificiels acycliques dans lequel le motif de connexion entre les neurones est inspiré par le cortex visuel des animaux. Les neurones de cette région du cerveau sont arrangés de sorte à ce qu'ils correspondent à des régions qui se chevauchent lors du pavage du champ visuel. Leur fonctionnement est inspiré par les processus biologiques, ils consistent en un empilement multicouche de perceptrons, Les réseaux neuronaux convolutifs ont de larges applications dans la reconnaissance d'image et vidéo [15].

d. Réseau de neurone récurrent :

Les réseaux récurrents (ou RNN pour *Recurrent Neural Networks*) sont des réseaux de neurones dans lesquels l'information peut se propager dans les deux sens, les RNNs sont particulièrement adaptés aux applications faisant intervenir le contexte, et plus particulièrement au traitement des séquences temporelles comme l'apprentissage et la génération de signaux.

Les RNNs (réseaux de neurones récurrents simples ou Vanilla RNNs) ne sont capables de mémoriser que le passé dit proche, et commencent à « oublier » au bout d'une cinquantaine d'itérations environ. Ce transfert d'information à double sens rend leur entraînement beaucoup plus compliqué, et ce n'est que récemment que des méthodes efficaces ont été mises au point comme les LSTM (Long Short Term Memory).

e. Réseau de neurone profond

DNN est une version plus approfondie d'un réseau de neurones qui comprend généralement trois types de couches: entrée, caché et sortie. Les couches d'entrée et de sortie sont des couches uniques, tandis que les couches cachées peuvent être plusieurs couches en fonction de la complexité du traitement du signal.

Chapitre I : Généralité sur l'apprentissage profond

Chaque couche contient plusieurs nœuds et n'affecte que les couches adjacentes. La structure du modèle DNN est montrée dans la figure 1.9

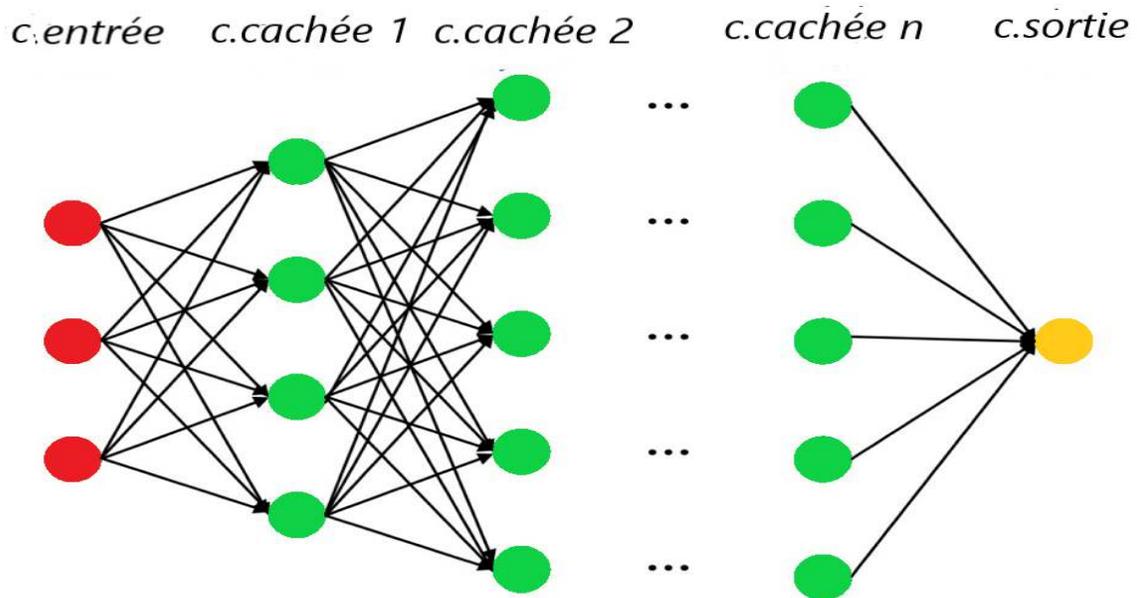


Figure 1.9: construction d'un modèle de réseau profond (DNN)

1.5. Conclusion

Nous avons présenté dans ce chapitre quelques généralités sur l'apprentissage profond ((définition, Architectures...etc.). Nous avons détaillé les réseaux neurones et leur utilisation. Le prochain chapitre, traite les détails de la conception, ainsi quela méthode et les outils utilisés pour la réalisation de notre application.

Chapitre II: Système OFDM

II.1. Contexte

Un signal radiofréquence est émis sur un canal de transmission qui constitue son support physique, Ainsi, toute transmission numérique est limitée par les contraintes physiques de son support. Un canal est dit sélectif en fréquence lorsqu'il ne se comporte pas identiquement suivant la fréquence du signal

Certaines fréquences seront transmises plus rapidement que d'autres, ou encore seront atténuées plus que d'autres.

Le signal sera alors déformé lors de la transmission les données seront dispersées dans le temps, pouvant mener à des interférences entre symboles communément appelées ISI (Inter Symbole interférences), la sélectivité de fréquence et causée par la présence de multi trajets de même signal transmis. Subir plusieurs réflexions de signal.

Le récepteur recevra une série d'échos d'amplitudes et de retards variables.

Cette problématique du canal à trajets multiples est critique dans le cas d'un canal radio mobile, c'est-à-dire lorsque le récepteur et l'émetteur ne sont pas fixes l'un par rapport à l'autre. Les problèmes d'interférence entre symboles (ISI) dans les systèmes de communication à porteuse unique sont considérablement réduits lorsque la période de symbole est supérieure au temps.

De communication à porteuse unique est considérablement réduit lorsque la période de symbole est plus grande que le retard temps [16]. Cependant, une longue période de symbole entraîne un débit de données très faible, ce qui rend le système de communication inefficace. Ne suffit pas pour transférer des données à un débit élevé. La demande de communications à large bande augmente chaque jour. La communication multi porteuse est utilisée pour répondre à ce besoin croissant [17].

Le multiplexage par répartition en fréquence (MRF) est une technique multi porteuse qui subdivise le spectre du canal de communication afin de transmettre des données en parallèle sur plusieurs porteuses.

L'interférence entre porteuses (ICI) est un autre problème possible, les porteuses sont très proches les unes des autres pour atteindre un débit de données élevé. Ce problème est résolu en plaçant des bandes de garde entre les porteuses, ce qui réduit le débit de données en contrepartie.

Actuellement, l'OFDM est la technique la plus courante pour de nombreux systèmes de communication, en raison de sa capacité à fournir un débit élevé sans être confronté au problème de l'ICI et de l'ISI [18].

II.2. Canal radio-Mobile

II.2.1. Caractéristique de canal radio mobile

Dans un environnement sans fil tel que celui de la figure 2.1, les signaux radio peuvent arriver au récepteur par deux voies ou plus, dans un phénomène appelé propagation par trajets multiples. Le récepteur par deux chemins ou plus dans un phénomène connu sous le nom de propagation par trajets multiples.

La propagation par trajets multiples est causée par de nombreux facteurs tels que la canalisation atmosphérique, la réflexion de l'ionosphère et la réfraction. Des déphasages, des interférences destructives et constructives sont possibles dans un environnement à trajets multiples [20].

Il en résulte un évanouissement de Rayleigh qui affecte la qualité du signal reçu après avoir causés des erreurs.

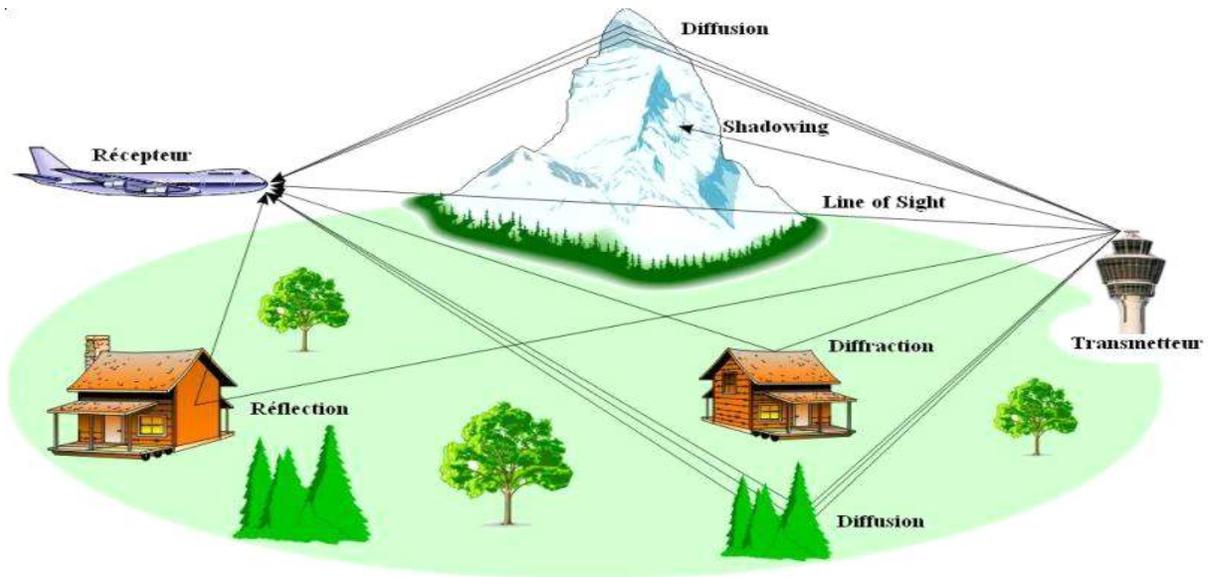


Figure II.1: Propagation radio-mobile [20]

L'évanouissement est simplement le résultat du chevauchement du signal reçu qui se produit après la réception de plusieurs copies de ce signal à des moments différents. Dans ce cas, le récepteur signaux qui peut avoir des amplitudes et des phases différentes en raison du temps de propagation, de la distribution de l'intensité du signal et de la durée de l'onde. Temps de propagation, de la distribution de l'intensité des ondes et de la largeur de bande du signal émis [20].

L'évanouissement peut être divisé en deux catégories principales : l'évanouissement à grande échelle et l'évanouissement à petite échelle.

La première catégorie est généralement due à la présence d'objets autour de l'environnement de propagation, tels que des collines, des montagnes etc....

L'évanouissement à petite échelle se produit lorsqu'il n'existe pas de ligne de vue unique (SOF) entre les deux côtés.

Cela se produit généralement lorsque les antennes sont plus courtes que certains des objets qui l'entourent. L'évanouissement à petite échelle se produit parfois même s'il y a une

ligne de vue parce que la réflexion d'autres objets peut conduire à la réception de signaux avec des différentes ondes dont les amplitudes et les phases sont distribuées de manière aléatoire.

La façon pratique de lutter contre l'évanouissement est d'envoyer plus d'une version du signal et de les combiner de façon cohérente au niveau du récepteur. Cela peut être réalisé par l'utilisation de plusieurs antennes. La façon la plus simple de modéliser l'évanouissement est de multiplier le signal par une variable aléatoire représentant un coefficient dépendant du temps. On obtient ainsi un rapport signal/bruit SNRs.

D'un point de vue statistique, les fluctuations de l'enveloppe du signal à court terme sont généralement caractérisées par une loi de Rayleigh ou de Rice. La première correspond généralement au milieu urbain, quand il n'y a pas une ligne visuelle directe (NLOS : No Line-Of-Sight), tandis que la seconde correspond au milieu rural, quand il y a une ligne visuelle directe (LOS).

II.3. System OFDM (Orthogonal Frequency Division Multiplexing)

II.3.1. Introduction

L'OFDM est basé sur l'idée de la modulation multi porteuse qui a été introduite au milieu des années 60.

Les données sont représentées sous forme de bits dans les communications numériques et le terme symbole indique un groupe de bits de différentes tailles.

Les symboles M-PSK ou QAM sont utilisés pour générer des données OFDM. Les symboles sont ensuite convertis en N flux parallèles au moyen de convertisseurs série/parallèle. La tâche des différentes sous-porteuses est d'exécuter chacun des N symboles. La transformation de Fourier discrète inverse (IDFT) convertit dans le domaine temporel.

Elle est appliquée dans ce cas car elle est plus efficace, [20]. Toutes les porteuses OFDM transmettent par le canal de communication simultanément en parallèle en utilisant

la bande passante maximale disponible. Au niveau du récepteur, l'application d'une transformée de Fourier rapide (FFT).

En résumé, l'idée principale de l'OFDM consiste à diviser la bande spectrale disponible en sous-canaux (sous-porteuses). Cette subdivision spectrale conduit à des sous canaux dont la bande de fréquence reste très faible par rapport à la bande de cohérence du canal. Ceci garantit la propriété de non sélectivité en fréquence du canal [20]. Pour obtenir une efficacité spectrale élevée, les réponses fréquentielles des sous-canaux sont en partie non disjointes mais orthogonales, d'où l'appellation OFDM.

II.3.2. Historique

La modulation multi-porteuse a été introduite à la fin des années 50, sa première utilisation était dans des systèmes de communications hautes fréquences militaires. Quelques années plus tard elle a été améliorée avec l'apparition du concept des signaux orthogonaux à bande limitée, concept que l'on appellera par la suite "Orthogonal Frequency Division Multiplexing" (OFDM) ou multiplex à division de fréquences orthogonales. La mise en œuvre de l'OFDM à l'époque consistait à utiliser des filtres de Naquis [16]. Du fait de la complexité à générer des bancs de filtres de sinusoides, l'OFDM n'a pas tout de suite intéressé les industriels civils. 20 ans plus tard le schéma de modulation-démodulation a été simplifié avec l'utilisation de la transformée de Fourier discrète inverse (TDFI) à l'émission et de la TFD au niveau du récepteur, ce qui rend facile son implémentation numérique. Dans le milieu des années 1980, la technique OFDM a été développée dans les domaines industriels civils tels que le projet de radiodiffusion numérique DAB (Digital Audio Broadcasting).

II.3.3. Définition de l'OFDM

OFDM est un schéma de modulation multi carrier dans lequel la bande passante totale est divisée en sous-porteuses orthogonales plus petites. Chaque sous-porteuse est ensuite modulée avec un flux de données numérique différent. Lorsque les symboles d'information sont modulés avec de nombreuses porteuses, les bandes latérales s'étalent normalement à l'intérieur des sous-porteuses voisines.

Le récepteur dans ce cas ne pourra démoduler que lorsqu'il y a une séparation considérable entre les porteurs. Il doit y avoir une séparation entre les différentes fréquences porteuses sinon des interférences inter carrier (ICI) se produiront. OFDM est différent de la modulation traditionnelle dans ce sens, même si toutes les bandes latérales se chevauchent, encore, l'information peut être reçue. La raison en est le fait que dans OFDM les sous-porteuses sont orthogonales les unes aux autres. Ceci est réalisé en divisant la bande passante disponible en plusieurs porteuses, avec un espacement égal, où l'espacement est égal à l'inverse de la période de symbole.

II.3.4. Problème d'interférences :

a. Interférence entre symboles (IES)

Dans un environnement à trajets multiples, un symbole transmis prend différents retards pour arriver au récepteur par différents chemins de propagation. Du point de vue du récepteur, le canal présente une dispersion temporelle dans laquelle la durée du symbole reçu est étalée. Prolonger la durée de symbole fait chevaucher le symbole reçu à l'instant avec les symboles reçus aux autres instants. C'est le phénomène des interférences entre symboles (IES). Dans le but de surpasser ce problème l'OFDM utilise un intervalle garde.

b. Interférence entre porteuse (IEP)

Ce genre d'interférence (IEP) peut apparaître dans les systèmes de transmissions multi-porteuses lors des chevauchements des spectres des différentes sous-porteuses. Ces interférences sont causées principalement par une perte de l'orthogonalité entre les différentes sous-porteuses adjacentes.

Afin d'éliminer ces interférences Dans les systèmes OFDM, les spectres des sous-porteuses se recouvrent mais demeurent orthogonaux entre eux, ceci signifie qu'au maximum de chaque spectre de sous-porteuse, tous les spectres des autres sous-porteuses s'annulent.

II.3.5. Notion d'orthogonalité

La différence fondamentale entre les techniques classiques de modulation multi-porteuses et l'OFDM réside dans le fait que cette dernière autorise un fort recouvrement spectral entre les sous-porteuses, ce qui permet d'augmenter sensiblement leur nombre ou d'amoinrir l'encombrement spectral. Cependant, pour que ce recouvrement n'ait pas d'effet néfaste, les sous porteuses doivent respecter une contrainte d'orthogonalité, à la fois dans les domaines temporel et fréquentiel.

L'utilisation d'un très grand nombre de sous porteuses est une alternative contraignante car nécessitant beaucoup de modulateurs, de démodulateurs et de filtres. Il faut aussi davantage de largeur de bande. Il est heureusement simple de résoudre ces deux problèmes en spécifiant un espacement rigoureusement régulier de $f = 1/T$ entre les sous-porteuses [21], où T est la période (utile ou active) du symbole pendant laquelle le récepteur intègre le signal démodulé. Les sous porteuses forment alors un ensemble orthogonal.

En considérant d'abord le signal OFDM comme un simple multiplexage en fréquence, la k ème sous-porteuse (en bande de base) peut s'écrire sous la forme :

$$\Psi_k(t) = e^{j2\pi f_k t}$$

Avec $f_k = f_0 + k \frac{1}{T_s}$

Les porteuses doivent satisfaire la condition d'orthogonalité suivante :

$$\int_t^{t+T_s} \Psi_k(t) \Psi_m^*(t) dt = \int_t^{t+T_s} e^{j2\pi f_k t} e^{-j2\pi f_m t} dt = \begin{cases} 0, & k \neq m \\ 1, & k = m \end{cases} \quad (2.1)$$

Les fonctions $\Psi_k(t)$ forment une base orthogonale de l'espace-temps-fréquence, ce qui permet de récupérer facilement les symboles.

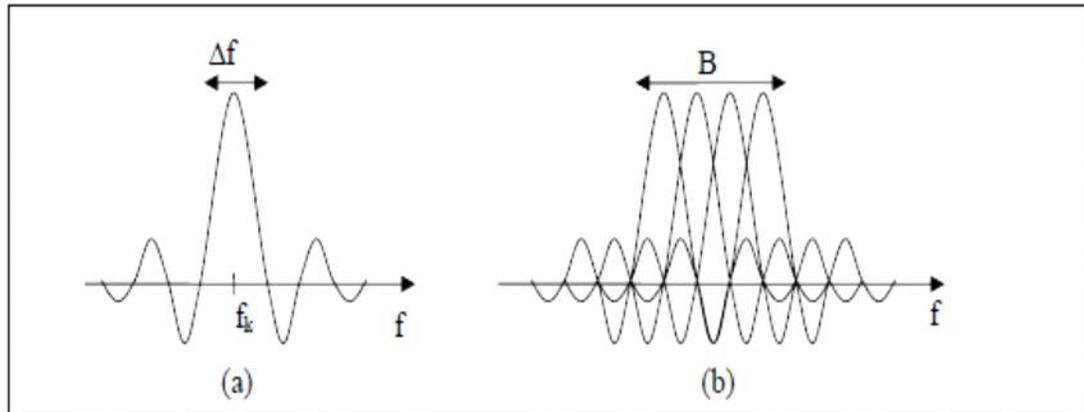


Figure II.2 : Spectre d'un ensemble orthogonal

II.3.6. Préfixe cyclique et intervalle de garde

Prenons une même suite de symboles arrivant à un récepteur par deux chemins différents qui se présente comme une même information arrivant à deux instants différents. Elles vont donc s'additionner provoquant ainsi une interférence entre symboles (IES). Entre chaque symbole transmis, on insère une zone "morte" appelée intervalle de garde. De plus, la durée utile d'un symbole sera choisie suffisamment grande par rapport à l'étalement des échos. Cette précaution va limiter l'interférence entre symbole. La durée pendant laquelle est émise l'information diffère de la période symbole car il faut prendre en compte, entre deux périodes utiles, un "temps de garde" qui a pour but d'éliminer l'IES qui subsiste malgré l'orthogonalité des porteuses.

Pour que cet intervalle de garde soit efficace, sa durée doit être au moins égale à l'écho non négligeable le plus long (c'est-à-dire, le retard maximal).

L'intervalle de garde peut être un temps de garde "blanc" [22], pendant lequel on n'émet rien, mais aussi il peut être rempli d'une copie de la fin de la trame OFDM. Si cette méthode est efficace pour lutter contre l'IES, elle pénalise cependant sensiblement le débit de transmission, et certains systèmes OFDM profitent des progrès dans le domaine de l'égalisation pour s'en affranchir.

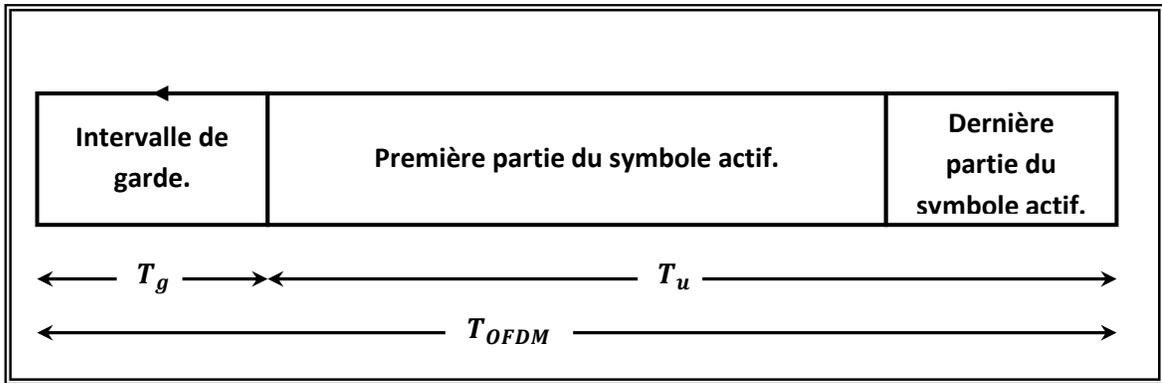


Figure II.3 : Exemple d'intervalle de garde

La figure illustre l'ajout d'un intervalle de garde. La période du symbole est prolongée de manière à être supérieure à la période utile. Le segment ajouté au début du symbole pour former l'intervalle de garde est donc identique au segment de même longueur à la fin du symbole. Tant que le retard d'un trajet par rapport au trajet principal (le plus court trajet) est inférieur à l'intervalle de garde, les composantes du signal à l'intérieur de la période d'intégration viennent toutes du même symbole : Le critère d'orthogonalité est satisfait. Les brouillages IEP et IES ne se produisent que lorsque le retard relatif est plus long que l'intervalle de garde. La longueur de l'intervalle est choisie de manière à correspondre au niveau de trajets multiples prévu. Elle ne devrait pas représenter une trop grande partie de T_u , pour ne pas sacrifier trop de capacité en données (et de rendement spectral).

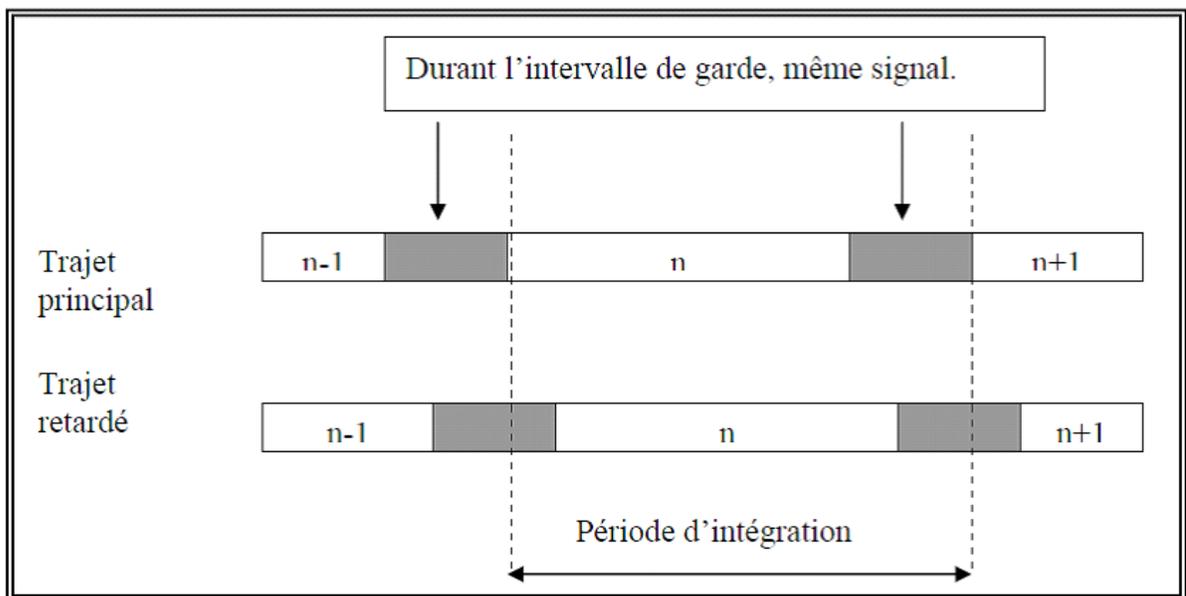


Figure II.4 : Signaux arrivant par différents trajets.

Les signaux, arrivant des différents trajets comme montre la Figure 2.5 peuvent s'ajouter de manière constructive ou destructive.

Grace à l'OFDM, l'étape d'égalisation du canal se simplifie, en fait, il est possible de montrer que le signal démodulé à partir d'une porteuse donnée est très similaire au signal émis multiplié par la réponse fréquentielle équivalente du canal sur la même fréquence porteuse.

II.3.7. Modèles mathématiques de l'OFDM

La transmission avec modulation OFDM (Orthogonal Frequency Division Multiplexing) peut être modélisée de plusieurs manières. La figure illustre une représentation en modèle continu du système OFDM.

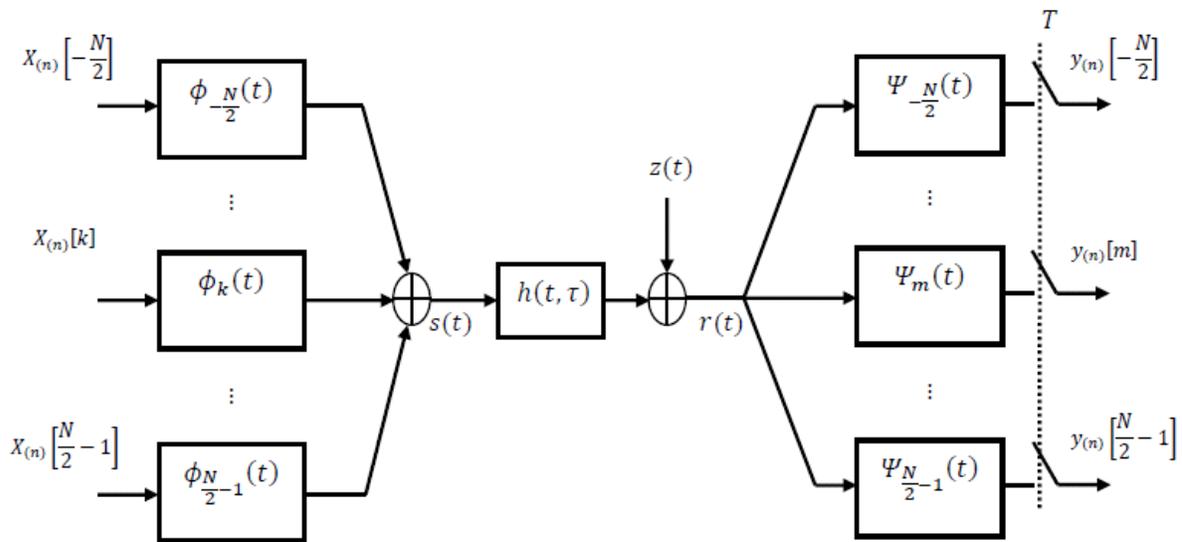


Figure II.5 :Système OFDM en bande de base à temps continu

II.3.8. Modèle discret de l'OFDM

Mettre en œuvre un système OFDM continu tel que nous l'avons décrit précédemment nécessite l'utilisation de N filtres analogiques en parallèle parfaitement orthogonaux. Ces filtres sont pratiquement très difficiles à réaliser et leur implantation est

très coûteuse. C'est pourquoi cette opération est aujourd'hui réalisée numériquement. Le modèle du système OFDM à temps discret est illustré dans la figure.

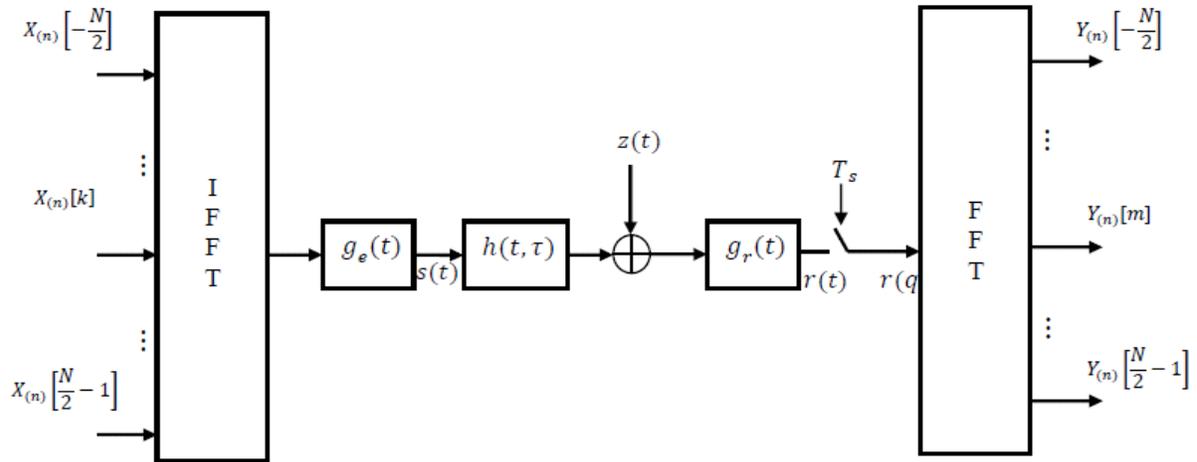


Figure II.6 : Système OFDM en bande de base à temps discret

II.3.9. La chaîne de transmission OFDM

Le synoptique de la Figure 2.7, illustre les différents blocs qui composent la chaîne de transmission OFDM. Le modulateur numérique (Q-PSK) transforme les données binaires de durée en symboles complexes de durée, où est la taille de la constellation de la modulation numérique (Q-PSK) utilisée. Le convertisseur série-parallèle dispose les symboles en groupes (trames) de symboles, la durée d'une trame est fois plus grande que la durée d'un symbole en série. Par conséquent, l'effet de canal devient moins nuisible. En appliquant ensuite une transformée de Fourier discrète inverse, on obtient la trame (symbole) OFDM. La TFDI est utilisée afin de transformer le spectre du signal OFDM au domaine temporel pour la transmission à travers le canal. Un préfixe cyclique de durée copie les derniers symboles de la trame OFDM et les ajoute ensuite au début de la trame. Après conversion parallèle-série, on obtient enfin le symbole OFDM qui contient symboles de durée totale que l'on transmet à travers le canal.

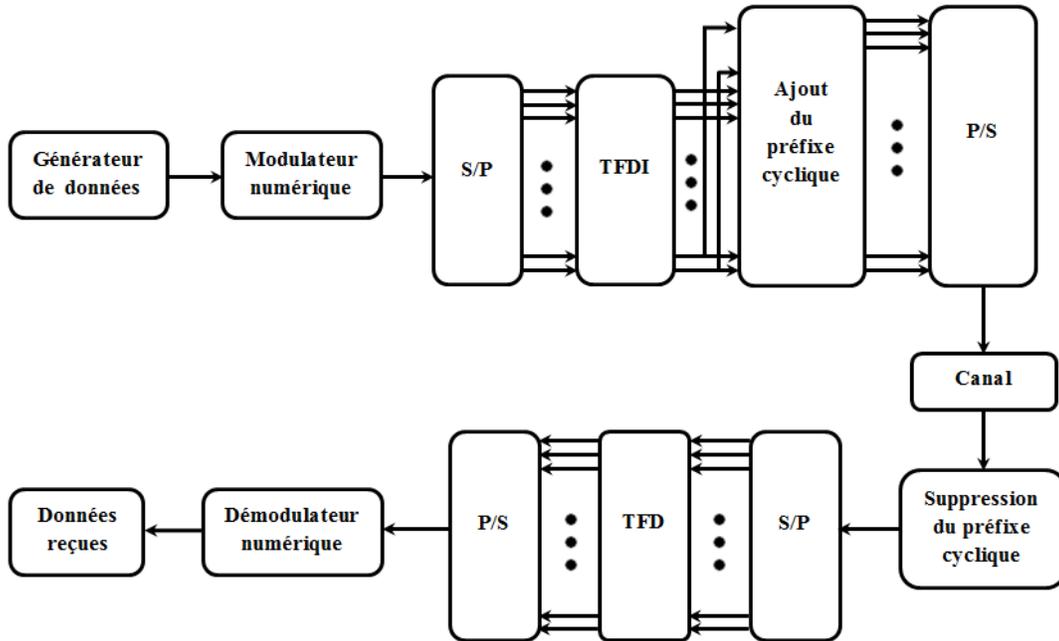


Figure II.7 : Schéma d'un système OFDM en bande de base.

II.3.10. Les importants Paramètres de l'OFDM

Les largeurs de sous porteuse sont le résultat de la division de la bande passante w de la communication canal, Elle est donnée par $\Delta f = w/N$ où N indique le nombre de sous-porteuses. La largeur de bande du canal est divisée orthogonalement comme sur la figure (2.9). L'efficacité spectrale est un avantage de ce processus car le pic de chaque signal chevauche les autres creux de signaux sans causer d'interférence. Par conséquent, le système qui répond à cette exigence est connu sous le nom de système OFDM. En fait, on peut dire que les porteuses orthogonales sont l'assentiment principal de l'OFDM.

Deux signaux sont orthogonaux l'un à l'autre s'ils satisfont à une condition. Cette condition exige que l'intégrale de deux signaux multipliés soit nulle sur une période de temps. En fait, deux sinusoides quelconques sont considérées comme orthogonales à condition que leurs fréquences soient des multiples entiers de la fréquence. Ceci est exprimé dans l'équation 2.2

$$\int_0^T \cos(2\pi nft) \cos(2\pi mft) dt = 0 \quad (n \neq m) \quad 2.2$$

Où indiquer n et m deux entiers, T est la période de l'intégration, qui est le période d'un symbole. Il est pratique de régler f à afin d'obtenir le plus d'efficacité du système OFDM.

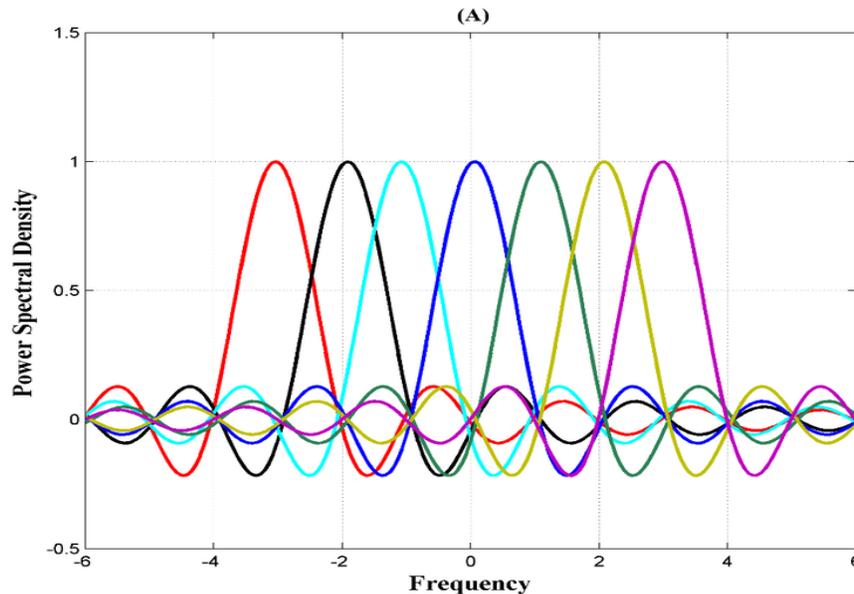


Figure 11.8 : Le signal multi carrier peut être calculé comme dans l'équation 2.2 [22]

$$S(t) = \sum_{m=-\infty}^{\infty} \left(\sum_{K=0}^{N-1} x_{k,m} \psi_k(t - mT_c) \right) \quad 2.3$$

Dans l'équation 2.3 le sous-porteuse est modulée par $x_{k,m}$ symbole

Deux facteurs contrôlent le nombre de sous-porteuses dans n'importe quel système OFDM. Ces deux les facteurs sont la taille d'IFFT et la bande passante du canal [23].

La taille de l'IFFT est déterminée par la relation dans l'équation 2.3. ISI est réduit lorsque le nombre de porteuses augmente.

$$\left(\text{nombre de porteuse} \leq \frac{\text{taille IFFT}}{2} - 2 \right) \quad 2.4$$

La modulation et la démodulation OFDM sont illustrées à la figure 2.9. Nous pouvons clairement voir sur la figure que les sous porteuses d'une fréquence sont modulés par le symbole du domaine fréquentiel X [K] pour N=6 i.e k=0,1,2,..5.

Les sous-porteuses orthogonales sont utilisées pour les démoduler dans le récepteur. Chaque symbole OFDM est composé de N symboles sous une forme parallèle. En outre, il est

important de noter que la longueur originale de l'symbole X [K] est l'extension $T_s aT_s = NT_s$ à la suite de la transmission simultanée de N symboles. Chaque symbole module l'une des porteuses orthogonales, ce qui signifie qu'il y'a N sous-porteuses.

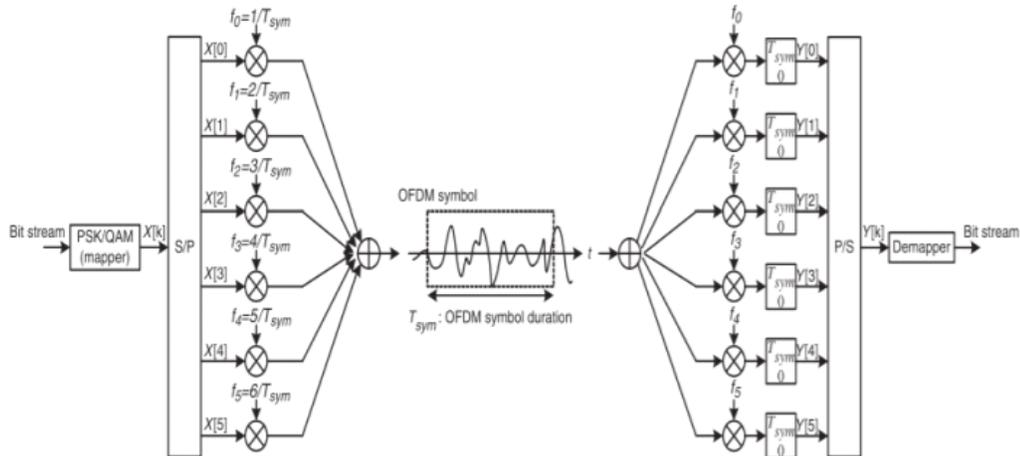


Figure II.9: Schéma fonctionnel de la modulation OFDM et de la démodulation

II.3.11. Les avantages de la technique de modulation OFDM

a. Immunité à la décoloration sélective de fréquence

L'un des principaux avantages de l'OFDM est qu'il est plus résistant à la décoloration à fréquence sélective que les systèmes à bande unique parce qu'elle divise le canal global en plusieurs signaux à bande étroite qui sont affectés individuellement comme plate fading sous-canaux

b. Résilience aux interférences

Les interférences survenant dans la couche physique peuvent être à bande limitée donc il n'affecterait pas tous les sous-canaux.

c. Efficacité spectrale

Les sous-canaux sont étroitement espacés et se chevauchent ainsi l'utilisation totale de la bande passante est élevée par rapport à un système de transporteur simple.

d. Efficacité Contre Le Bruit Impulsif

Le bruit impulsif peut être défini comme l'interférence qui résulte des phénomènes naturels tels que la foudre. Cela se produit dans les communications qui utilisent des canaux sans fil ou à paire torsadée. Habituellement, la durée du symbole est bien inférieure à la longueur de la forme d'onde d'interférence. Afin de clarifier cela, nous pouvons imaginer avoir un système de 10 Mbps qui a $0,1 \mu\text{s}$ comme durée de symbole. Une erreur de rafale qui ne peut pas être inversée à l'aide de codes correcteurs d'erreurs classiques se produit si la forme d'onde de bruit est supérieure à deux microsecondes [21].

II.3.12. Inconvénients de l'OFDM :

Bien que l'OFDM ait apporté de nombreuses fonctionnalités aux systèmes de communication, il présente certains inconvénients. Ces inconvénients sont étudiés partout dans le monde afin de fabriquer des systèmes OFDM [24].

a. Rapport de Puissance de Crête à Moyenne (PAPR)

Le PAPR élevé est considéré comme l'un des principaux inconvénients de l'OFDM. La raison derrière cela pourrait être comprise si chaque sous-porteuse modulée est considérée comme un signal complexe à sa propre fréquence. Cela est clair puisque les signaux OFDM ne sont rien d'autre que l'addition de N variables aléatoires complexes. Par conséquent, PAPR est considéré comme un problème d'émetteur. La forme d'onde transmise doit contenir tous les pics. Par conséquent, l'amplificateur de puissance de l'émetteur doit avoir une large plage linéaire afin d'éviter l'écrêtage de l'onde à transmettre. De tels amplificateurs sont très coûteux à construire et consomment une grande quantité d'énergie [25][26].

b. Sensibilité aux erreurs de fréquence

Les erreurs de fréquence peuvent entraîner une dégradation des performances d'un système OFDM. Ici, l'ICI provoque le BER dans les signaux reçus qui pourraient être affectés par les décalages de fréquence Doppler et les erreurs de synchronisation de porteuse [21].

II.4. Conclusion

En conclusion, dans ce chapitre, le principe de base des modulations multi-porteuses à fréquences orthogonales a été expliqué. Nous avons vu les problèmes d'interférence entre porteuses (IEP), et entre symboles (IES); et comment les éliminer grâce à la notion d'orthogonalité qui offre de meilleures performances, et l'intervalle de garde. Par la suite on a vu qu'une réalisation en terme de la transformée de Fourier discrète permet de surpasser les difficultés liées à la mise en œuvre des systèmes à porteuses multiples. Et on a terminé par discuter les avantages et inconvénients de l'OFDM.

Chapitre III : l'estimation et la détection

III.1. Introduction

Le multiplexage par répartition en fréquence orthogonale (OFDM) est un schéma de modulation populaire qui a été largement adopté dans les systèmes sans fil à large bande pour combattre l'évanouissement fréquence et dans les canaux sans fil. L'information sur l'état du canal (CSI) est essentielle à la détection et au décodage cohérent dans les systèmes OFDM. Habituellement, le CSI peut être estimé au moyen de pilotes avant la détection des données transmises. Avec le CSI estimé les symboles transmis peuvent être récupérés au niveau du récepteur.

Les méthodes d'estimation traditionnelles, c'est-à-dire les moindres carrés (LS) et l'erreur moyenne quadratique minimale (MMSE), ont été utilisées et optimisées dans diverses conditions. La méthode d'estimation LS ne nécessite pas de statistiques préalables sur les canaux, mais ses performances peuvent être insuffisantes. L'estimation MMSE en général conduit à de bien meilleures performances de détection.

Dans ce chapitre, nous introduisons une approche d'apprentissage profond pour l'estimation du canal et la détection des symboles dans un système OFDM. L'estimation des canaux et la détection des symboles dans un système OFDM.

III.2. Méthode d'apprentissage

Le modèle DNN (deep learning) prend en entrée les données reçues (composées d'un bloc pilote et d'un bloc de données) et restaure les données transmises de bout en bout. L'apprentissage profond a été appliqué avec succès dans une large gamme de domaines présentant une amélioration significative des performances. La structure d'un modèle DNN est illustrée à la Fig 3.1. Généralement, les DNN sont des versions plus profondes des ANN en augmentant le nombre de couches cachées afin d'améliorer la capacité en représentation ou en reconnaissance. Chaque couche du réseau se compose de plusieurs neurones, dont chacun a une sortie qui est une fonction non linéaire d'une somme pondérée de neurones de la couche précédente, comme le montre la Fig. 3.1.

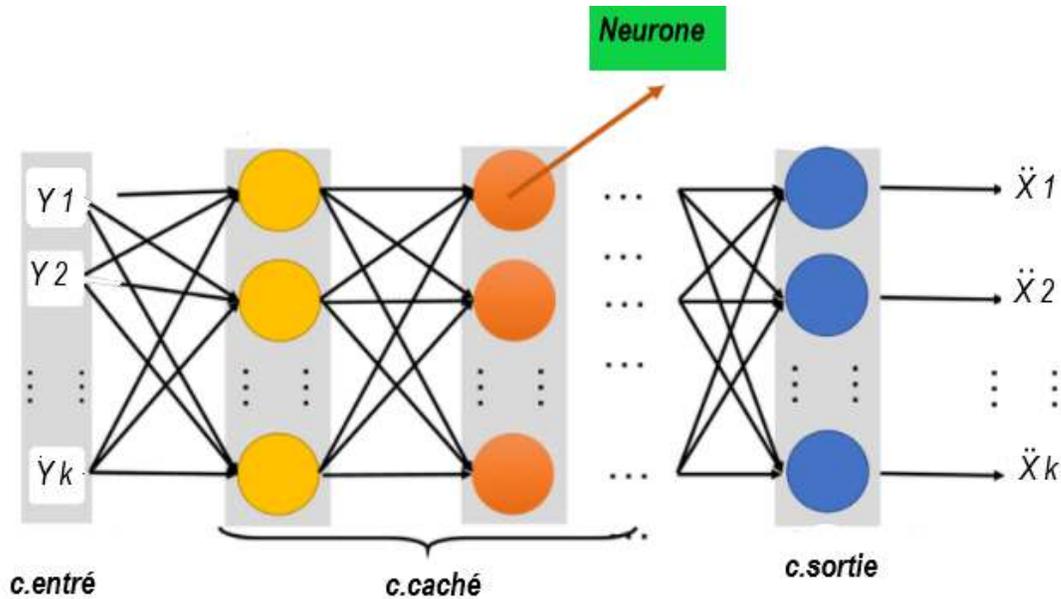


Figure III.1 : Modèle de DNN

La fonction non linéaire peut être la fonction Sigmoidale, ou la fonction Relu, Par conséquent, la sortie du réseau z est une cascade de transformation des données d'entrée I , exprimées mathématiquement en

$$z = f(I, \theta) = f^{(L-1)}(f^{(L-2)}(\dots f^1(I)))$$

Où L représente le nombre de couches, et θ représente le poids du réseau neurone.

III.2.1. Fonction activation

La fonction d'activation sert avant tout à modifier de manière non-linéaire les données. Cette non-linéarité permet de modifier spatialement la représentation des données [27].

Dit simplement, la fonction d'activation permet de changer notre manière de voir une donnée

a. Relu

La fonction RectifiedLinear Unit (ReLU) est la fonction d'activation la plus simple et la plus utilisée.

Chapitre III : l'estimation et la détection

Elle donne x si x est supérieur à 0, 0 sinon. Autrement dit, c'est le maximum entre x et 0:

Fonction $\text{Relu}(x) = \max(x, 0)$

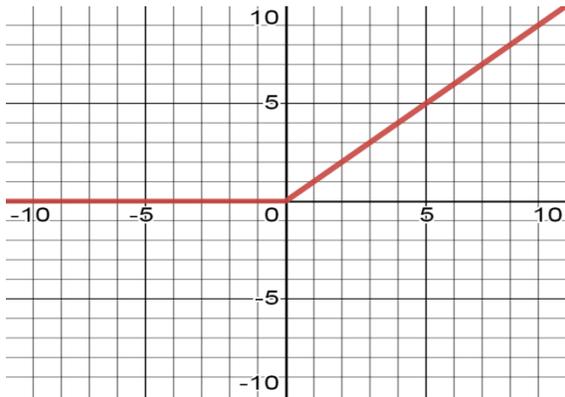


Figure III.2 : Fonction Relu

Cette fonction permet d'effectuer un filtre sur nos données. Elle laisse passer les valeurs positives ($x > 0$) dans les couches suivantes du réseau de neurones. Elle est utilisée presque partout mais surtout pas dans la couche finale, elle est utilisée dans les couches intermédiaires (cachée).

b. Sigmoid

La fonction sigmoïde donne une valeur entre 0 et 1, une probabilité. Elle est donc très utilisée pour la classification binaire lorsqu'un modèle doit déterminer seulement deux labels.

$$\text{fonction_Sigmoid}(x) = \frac{1}{1 + e^{(-x)}}$$

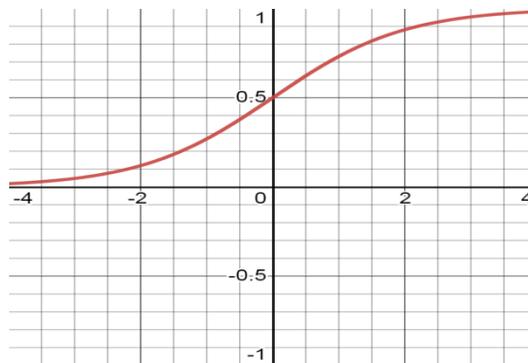


Figure III.3 : fonction Sigmoid

La fonction Sigmoidé est très simple a appliqué.

III.3. Architecture de system

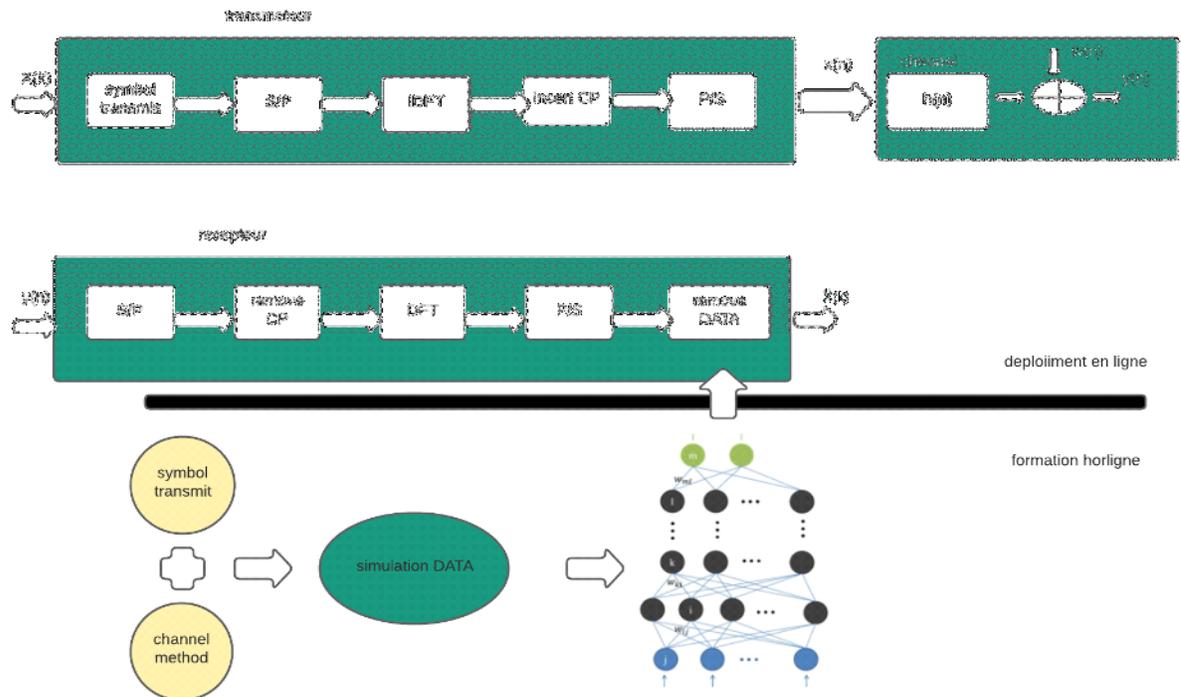


Figure III.4 : Architecture de système

L'architecture du système OFDM avec estimation de canal basée sur l'apprentissage profond et détection de signal est illustrée à la Fig 3.4. Le système OFDM en bande de base est le même que les systèmes conventionnels. Côté émetteur, les symboles transmis insérés avec des pilotes sont d'abord convertis en un flux de données parallélisé, puis la transformée de Fourier discrète inverse (IDFT) est utilisée pour convertir les signaux du domaine fréquentiel au domaine temporel. Après cela, un préfixe cyclique (CP) est inséré pour atténuer l'interférence inter-symboles (ISI). La longueur du CP ne doit pas être inférieure à la propagation de retard maximale du canal [28].

Nous considérons un canal multi-chemin espacé d'échantillon décrit par variables aléatoires complexes $\{h(n)\}_{n=0}^{N-1}$ Ainsi le signal reçu, $\mathbf{y}(n)$, peut être exprimé comme

$$y(n) = x(n) \otimes h(n) + w(n)$$

où \otimes désigne la convolution circulaire tandis que $x(n)$ et $w(n)$ représentent le signal transmis et Bruit blanc gaussien additif (AWGN), respectivement. Après avoir retiré le CP et la réalisation DFT, le signal de domaine fréquentiel reçu est :

$$Y(k) = X(k)H(k) + W(k)$$

Où $Y(k)$, $X(k)$, $H(k)$ et $W(k)$ sont les DFT de $y(n)$, $x(n)$, $h(n)$ et $w(n)$ respectivement. Nous supposons que les symboles pilotes sont dans le premier bloc OFDM alors que les blocs OFDM suivants sont constitués des données transmises. Le DNN modèle prend en entrée les données reçues consistant en un bloc pilote et un bloc de données dans notre étude initiale, et récupère la donnée transmise de bout en bout.

Comme le montre la Fig 3. 1, pour obtenir un modèle efficace de DNN pour estimation commune de canal et détection de symbole, deux étapes sont incluses. Dans la phase de formation hors ligne, le modèle est formé avec les échantillons OFDM reçus qui sont générés avec différentes séquences d'information et dans diverses conditions de canal avec certaines propriétés statistiques, telles que urbain ou profil vallonné de retard de terrain. Dans la phase de déploiement en ligne, le modèle DNN génère la sortie qui récupère les données transmises sans estimer explicitement le canal sans fil.

III.4. Estimation et détection basées sur l'apprentissage profond

III.4.1. Principe de fonctionnement

Notre objectif est de remplacer l'estimation de canaux et la détection de signaux par des réseaux de neurones. Et la conception d'un réseau de neurones, en fait, consiste essentiellement à concevoir trois choses :

- Architecture spécifique au réseau.
- Progrès de la formation
- Quelle est l'entrée et quelle est la sortie de réseaux.

A. Architecture spécifique au réseau

Notre réseau de neurones est composé de 5 couches, trois couches cachées, utilisant respectivement 256, 500, 250, 120 et 16 neurones. Chaque donnée transmise de 16 bits est regroupée et prédite sur la base d'un seul modèle formé indépendamment, puis connectée au modèle final pour la sortie. À l'exception de la dernière couche qui utilise la fonction Sigmoidale pour mapper la sortie sur l'intervalle [0,1], les autres couches utilisent la fonction Relu comme fonction d'activation.

B. Progrès de la formation

Les critères utilisés pour calculer la mise à jour du poids, Δw_{ij} sont introduits dans cette section.

1. La Descente de gradient stochastique (SGD)

La Descente de gradient stochastique (SGD) calcule l'erreur pour chaque entraînement donné et ajuste les poids immédiatement. Si nous avons 100 points de données d'entraînement, le SGD ajuste les poids 100 fois. La mise à jour du poids de la SGD est liée à l'ensemble des données de formation [30].

Lorsque le SGD ajuste le poids pour chaque point de données, les performances du réseau neuronal est tordu pendant le processus de formation. L'adjectif "stochastique" implique le comportement aléatoire du processus d'entraînement. Le SGD calcule les mises à jour de poids comme :

$$\Delta w_{ij} = \alpha \delta_i x_j$$

2. Batch (lots)

Dans la méthode batch, chaque mise à jour de poids est calculée pour toutes les erreurs de la formation données, et la moyenne des mises à jour de poids est utilisée pour ajuster les poids. Cette méthode utilise toutes les données de formation et ne les met à jour qu'une seule fois.

La méthode batch calcule la mise à jour du poids comme suit :

$$\Delta w_{ij} = \frac{1}{N} \sum_{k=1}^N \Delta w_{ij}(k)$$

Où Δw_{ij} est la mise à jour de poids pour les k données d'entraînement et N est le nombre total des données d'entraînement.

3. Mini batch

La méthode mini batch est un mélange des méthodes SGD et batch. Il sélectionne une pièce de l'ensemble de données de formation et les utilise pour la formation à la méthode par lots. Donc, il calcule les mises à jour de poids des données sélectionnées et entraîne le réseau neuronal avec la mise à jour du poids moyen. Par exemple, si 20 points de données arbitraires sont sélectionnés parmi 100 points de données d'apprentissage, la méthode par lots est appliquée aux 20 données point. Dans ce cas, un total de cinq ajustements de poids sont effectués pour compléter le processus de formation pour tous les points de données ($5 = 100/20$).

La méthode mini batch, lorsqu'elle sélectionne un nombre approprié de données points, obtient les avantages des deux méthodes : vitesse de la SGD et stabilité du lot. Pour cette raison, il est souvent utilisé dans l'apprentissage profond, qui manipule une quantité importante de données.

Maintenant, approfondissons un peu le SGD, le lot et le mini lot en termes de durées "epoch". La durée est brièvement introduite dans la " Formation d'un neurone monocouche Réseau : Règle Delta " section. En résumé, la durée est le nombre de cycles de formation pour toutes les données de formation. Dans la méthode par lots, le nombre des cycles d'entraînement du réseau neuronal sont égaux à une époque [30].

4. Rétro propagation (Back propagation)

Dans la pratique, une fonction de réseau de neurones est constituée de nombreuses opérations de tenseurs, chacune d'entre elles ayant une dérivée simple et connue.

Chapitre III : l'estimation et la détection

Ces opérations sont composées, et reliées entre elles pour former une fonction de réseau de neurones. C'est ce qu'on appelle une fonction composée. Cette fonction composée peut être dérivée grâce à la règle de la chaîne, Appliquer la règle de la chaîne permet d'obtenir le gradient de fonction de perte par rapport à chaque poids. On calcule le gradient de chaque couche en commençant par la dernière couche et en finissant par la première. Cette application est appelée la Backpropagation, elle est définie comme suit :

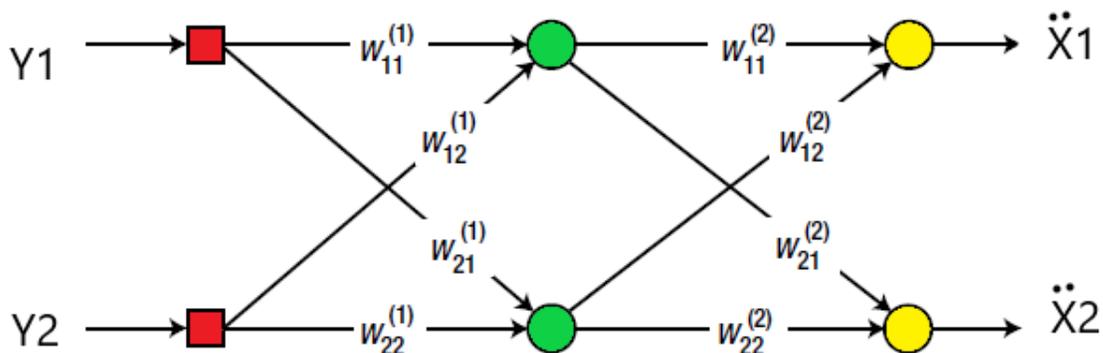


Figure III.5 : Réseau neurone qui se compose de deux nœuds pour l'entrée et la sortie et une couche cachée, qui a deux nœuds [30]

Afin d'obtenir l'erreur de sortie, nous avons d'abord besoin de la sortie du réseau neuronal à partir des données d'entrée. Essayons. Comme l'exemple de réseau à une seule couche cachée, nous avons besoin de deux manipulations de données d'entrée avant que le calcul de sortie soit traité. Tout d'abord, la somme pondérée du nœud caché est calculée comme suit [30]:

$$\begin{pmatrix} v_1^{(1)} \\ v_2^{(1)} \end{pmatrix} = \begin{pmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \quad (3.1)$$

Lorsque nous mettons cette somme pondérée, équation 3.1, dans la fonction d'activation, nous obtenons la sortie des nœuds cachés [30].

$$\begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} = \begin{pmatrix} \varphi(v_1^{(1)}) \\ \varphi(v_2^{(1)}) \end{pmatrix}$$

Chapitre III : l'estimation et la détection

Où $x_1^{(1)}$ et $x_2^{(1)}$ sont des sorties des nœuds cachés correspondants. Dans un de manière similaire, la somme pondérée des nœuds de sortie est calculée comme suit :

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} w_{11}^{(2)} & w_{12}^{(2)} \\ w_{21}^{(2)} & w_{22}^{(2)} \end{pmatrix} \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} \quad (3.2)$$

Lorsque nous mettons cette somme pondérée dans la fonction d'activation, le réseau neuronal produit la sortie [30].

$$\begin{pmatrix} \ddot{X}_1 \\ \ddot{X}_2 \end{pmatrix} = \begin{pmatrix} \varphi(v_1) \\ \varphi(v_2) \end{pmatrix}$$

Maintenant, nous allons former le réseau neuronal en utilisant l'algorithme de rétro propagation [30].

La première chose à calculer est delta δ (la règle delta ajuste le poids), de chaque nœud.

La **Figure III.6** a été redessinée avec les connexions inutiles atténuées.

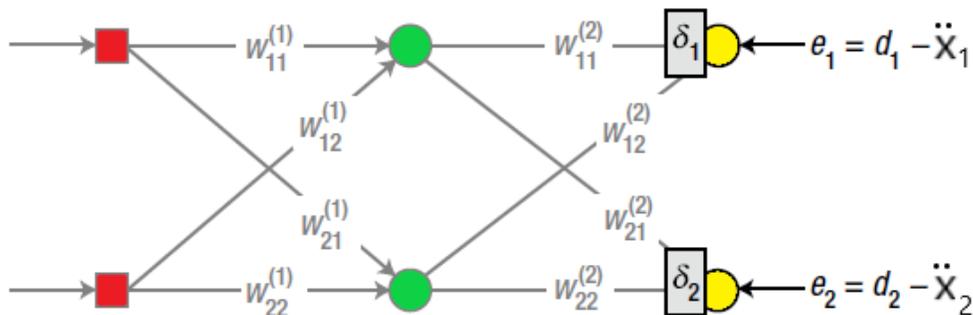


Figure III.6 : Former le réseau neuronal à l'aide de l'algorithme de rétro propagation.

$$w_{ij} + \alpha e_i y_j \rightarrow w_{ij}$$

y_j = La sortie du nœud d'entrée j , ($j=1, 2, 3$)

e_i = L'erreur du nœud de sortie i

Chapitre III : l'estimation et la détection

w_{ij} = Le poids entre le nœud de sortie i et l'entrée nœud j

α = Taux d'apprentissage ($0 < \alpha < 1$)

Dans l'algorithme de rétro propagation, le delta du nœud de sortie est défini identique à la règle delta

$$\left. \begin{aligned}
 e_1 &= d_1 - \ddot{X}_1 \\
 \delta_1 &= \varphi'(v_1)e_1 \\
 (3.3) \\
 e_2 &= d_2 - \ddot{X}_2 \\
 \delta_2 &= \varphi'(v_2)e_2
 \end{aligned} \right\}$$

Où $\varphi'(v)$ est la dérivée de la fonction d'activation du nœud de sortie, \ddot{X}_k est la sortie du nœud de sortie, d_i est la sortie correcte de la formation donnée, et v_1 est la somme pondérée du nœud correspondant. Puisque nous avons le delta pour chaque nœud de sortie, procédons vers la gauche aux nœuds cachés et calculons le delta (**Figure III.7**).

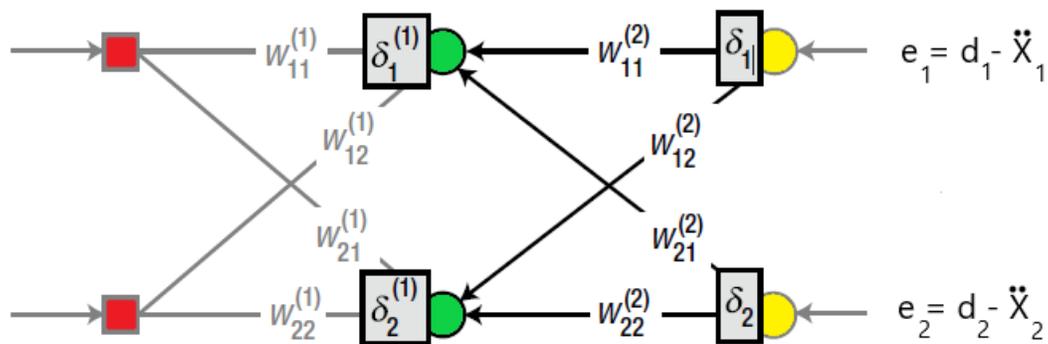


Figure III.7 : Continuez vers la gauche jusqu'aux nœuds cachés et calculez le delta.

Le problème du nœud caché est comment définir l'erreur. Dans l'algorithme de rétro propagation, l'erreur du nœud est définie comme la somme pondérée des deltas rétro

Chapitre III : l'estimation et la détection

propagés de la couche sur le droit immédiat (dans ce cas, la couche de sortie). Une fois l'erreur obtenue, le calcul du delta à partir du nœud est le même que celui de l'équation 3.3. Ce processus peut être exprimé comme suit :

$$\begin{aligned}
 e_1^{(1)} &= w_{11}^{(2)} \delta_1 + w_{21}^{(2)} \delta_2 \\
 \delta_1^{(1)} &= \varphi'(v_1^{(1)}) e_1^{(1)} \\
 e_2^{(1)} &= w_{12}^{(2)} \delta_1 + w_{22}^{(2)} \delta_2 \\
 \delta_2^{(1)} &= \varphi'(v_2^{(1)}) e_2^{(1)}
 \end{aligned}
 \tag{3.4}$$

Où $v_1^{(1)}$ et $v_2^{(1)}$ sont les sommes pondérales des signaux directs aux nœuds respectifs. On remarque à partir de cette équation que l'avant, la retro propagation est appliqués de manière identique aux nœuds cachés ainsi qu'aux nœuds de sortie. Cela implique que la sortie et les nœuds cachés subissent le même retro propagation. La seule différence est le calcul de l'erreur (**Figure III.8**).

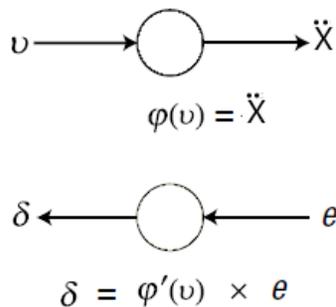


Figure III.8 : Le calcul de l'erreur est la seule différence

En résumé, l'erreur du nœud caché est calculée en tant que la somme pondérée du delta, et le delta du nœud est le produit de l'erreur et de la dérivée de la fonction d'activation. Ce processus commence à la sortie couche et répétitions pour tous les couches cachés. Cela explique à peu près ce que le l'algorithme de rétro propagationil est. Les deux

Chapitre III : l'estimation et la détection

formules de calcul d'erreur de l'équation 3.4 sont combinées dans une matrice équation comme suit :

$$\begin{pmatrix} e_1^{(1)} \\ e_2^{(1)} \end{pmatrix} = \begin{pmatrix} w_{11}^{(2)} & w_{12}^{(2)} \\ w_{21}^{(2)} & w_{22}^{(2)} \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix} \quad (3.5)$$

Comparez cette équation avec la sortie du réseau neuronal de l'équation 3.2. La matrice de l'équation 3.5 est le résultat de la transposition de la matrice de poids, W , de l'équation 3.2 Par conséquent, l'équation 3.5 peut être réécrite comme suit :

$$\begin{pmatrix} e_1^{(1)} \\ e_2^{(1)} \end{pmatrix} = W_2^T \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix} \quad (3.6)$$

Cette équation indique que nous pouvons obtenir l'erreur comme le produit de la matrice de poids transposée et le vecteur delta. Cet attribut très utile permet une mise en œuvre plus facile de l'algorithme. Si nous avons des couches cachées supplémentaires, nous répéterons simplement la même chose en arrière traiter pour chaque couche cachée et calculer tous les deltas. Une fois tous les deltas ont été calculés, nous serons prêts à former le réseau de neurones. Il suffit d'utiliser l'équation suivante pour ajuster les poids des couches respectives.

$$\Delta w_{ij} = \alpha \delta_i y_j \quad (3.7)$$

$$w_{ij} + \Delta w_{ij} \rightarrow w_{ij}$$

Où y_j est le signal d'entrée pour le poids correspondant. Pour plus de commodité, nous omettons l'indicateur de couche de cette équation. Équation identique à celle de la règle delta. La seule différence est les deltas des nœuds cachés, qui sont obtenus à partir du calcul en arrière en utilisant l'erreur de sortie du réseau de neurones. Nous allons procéder un peu plus loin et dériver l'équation pour ajuster le poids en utilisant l'équation 3.7. Considérons le poids $w_{21}^{(2)}$ par exemple. Le poids $w_{21}^{(2)}$ de la **Figure III.9** peut être ajusté en utilisant l'équation 3.7 comme :

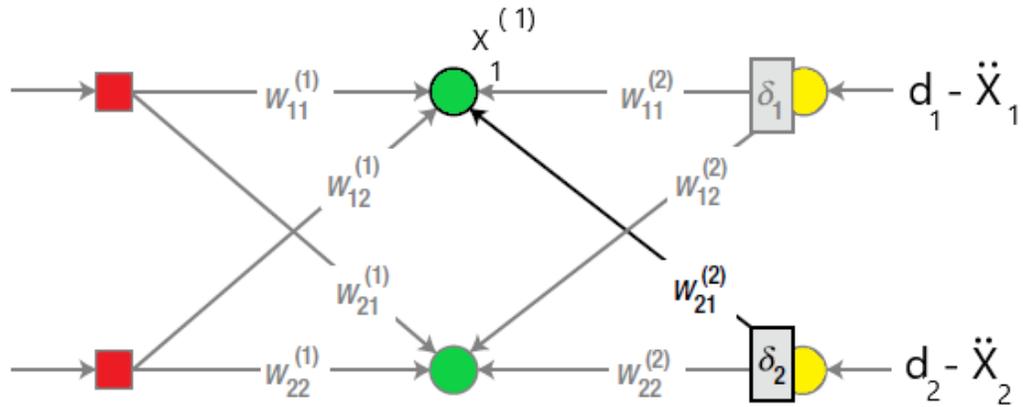


Figure III.8 : Dérivez l'équation pour ajuster le poids

$$w_{21}^{(2)} + \alpha \delta_2 x_1^{(1)} \rightarrow w_{21}^{(2)}$$

Où $x_1^{(1)}$ est la sortie du premier nœud caché. Voici un autre exemple. Le poids $w_{11}^{(1)}$ de la **Figure III.10** est ajusté à l'aide de l'équation 3.7 comme :

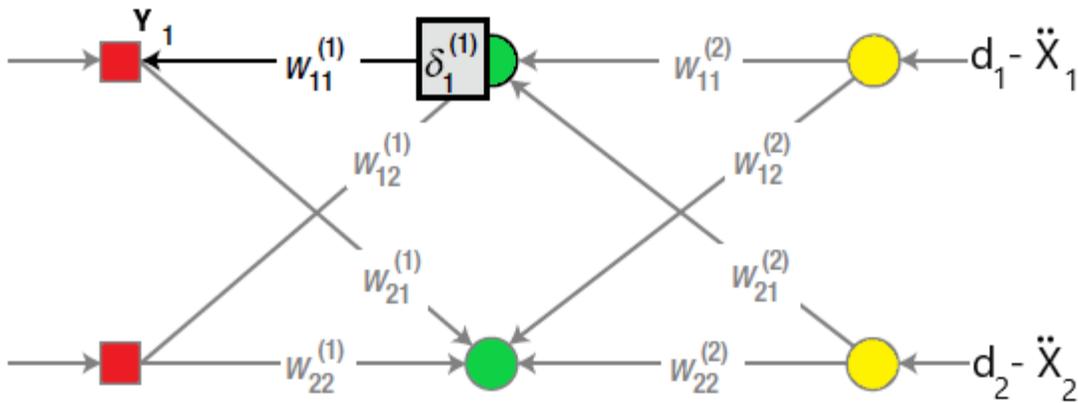


Figure III.9 : Dérivez l'équation pour ajuster le poids, encore une fois.

$$w_{11}^{(1)} + \alpha \delta_1^{(1)} y_1 \rightarrow w_{11}^{(1)}$$

Chapitre III : l'estimation et la détection

Où y_1 est la sortie du premier nœud d'entrée, c'est-à-dire la première entrée du réseau neuronal. Organisons le processus pour former le réseau de neurones en utilisant l'algorithme de propagation.

1. Initialisez les poids avec des valeurs adéquates.
2. Entrez des données d'entraînement {input, correctoutput} et obtenir la sortie du réseau neuronal. Calculer l'erreur de la sortie à la sortie exacte et le delta, δ , des nœuds de sortie.

$$e = d - \check{X}$$

$$\delta = \varphi'(v)e$$

3. Propager le nœud de sortie delta, δ , en arrière, et calculez les deltas des nœuds suivants immédiats (à gauche).

$$e^k = W^T \delta$$

$$\delta^{(k)} = \varphi'(v^{(k)})e^{(k)}$$

4. Répétez l'étape 3 jusqu'à ce qu'elle atteigne le couche caché le droit immédiat de la couche d'entrée.
5. Ajustez les poids selon la règle d'apprentissage suivante.

$$\Delta w_{ij} = \alpha \delta_i y_j$$

$$w_{ij} + \Delta w_{ij} \rightarrow w_{ij}$$

6. Répétez les étapes 2 à 5 pour chaque point de données d'entraînement.
7. Répétez les étapes 2 à 6 jusqu'à ce que le réseau neuronal soit correctement formé.

Autres que les étapes 3 et 4, dans lesquelles le delta de sortie se propage vers l'arrière pour obtenir le delta du nœud caché, ce processus est fondamentalement le même que celui de la règle delta, qui a déjà été discutée. Bien que cet exemple n'ait qu'une couche cachée,

Chapitre III : l'estimation et la détection

l'algorithme de rétro-propagation est applicable pour la formation de nombreuses couches cachées. Il suffit de répéter l'étape 3 de l'algorithme précédent pour chaque couche cachée.

Ainsi on peut calculer avec précision l'influence de chaque paramètre dans le résultat de fonction perte et les modifier en conséquence.

Finalement, cela nous permet d'optimiser nos poids, minimiser la perte de fonction de perte et on peut enfin utiliser notre modèle de Deep Learning comme il se doit [29].

5. Fonction de loss :

La fonction de perte est l'une des composantes importantes des réseaux neuronaux. La perte n'est rien d'autre qu'une erreur de prédiction du réseau neuronal. Et la méthode pour calculer la perte est appelée Fonction de perte. En termes simples, la perte est utilisée pour calculer les gradients. Et les gradients sont utilisés pour mettre à jour les poids du réseau neuronal.

a. Erreur quadratique moyenne (MSE) :

C'est la fonction de perte préférée dans le cadre d'inférence du maximum de vraisemblance si la distribution de la variable cible est gaussienne.

L'erreur quadratique moyenne est calculée comme la moyenne des différences au carré entre les valeurs prévues et réelles. Le résultat est toujours positif quel que soit le signe des valeurs prédites et réelles et une valeur parfaite est 0,0. La quadrature signifie que les erreurs plus importantes entraînent plus d'erreurs que les erreurs plus petites, ce qui signifie que le modèle est corrigé pour avoir commis des erreurs plus importantes.

$$L_2 = \frac{1}{N} \sum_k (\hat{X}(k) - X(k))^2$$

$\hat{X}(k)$: la valeur prédit.

$X(k)$: la valeur d'entre.

b. L'entrée et la sortie de réseau :

La sortie est facile à comprendre, puisque notre objectif est de restaurer le signal d'origine, alors notre valeur de sortie est le signal que nous avons restauré. Alors, quelle est l'entrée ? Évidemment, nous devrions donner toutes les informations disponibles au réseau

de neurones. L'information disponible est la donnée reçue y , qui est l'entrée du réseau. L'information de h doit être obtenue par estimation de canal, afin d'obtenir x à partir de y . Donc, pour notre réseau de neurones, devrions-nous utiliser l'estimation de canal traditionnelle pour estimer h d'abord, puis entrer h et y dans le réseau de neurones ? Nous donnons toutes les informations utilisées dans la méthode traditionnelle au réseau, Par conséquent, l'entrée du réseau neuronal est le signal de réception pilote et le signal de réception de données.

III.4.2. Détails d'implémentation

A. Système OFDM

Dans le système OFDM réel, le signal est transmis trame par trame. Par exemple, 64 bits forment une trame, puis passent par FFT, l'IFFT Attende que l'opération atteigne le domaine fréquentiel. La station de base envoie d'abord une trame de $64 * 1$ signal pilote, puis envoie une trame de $64 * 1$ signal de données, puis l'extrémité réceptrice reçoit un total de $128 * 1$ vecteur. Notez qu'en communication, parce que le signal est divisé en canal I et canal Q pour la transmission, chaque élément du vecteur est un nombre complexe. Les cadres d'apprentissage profond traditionnels ne prennent souvent en charge que les opérations de nombres réels, de sorte que l'opération utilisée dans notre mémoire consiste à diviser les parties réelles et imaginaires du vecteur, puis à les réunir en un vecteur réel $256 * 1$, c'est-à-dire que le vecteur Les 128 premiers éléments est la partie réelle du vecteur d'origine, et les 128 derniers éléments sont la partie imaginaire. Du point de vue de l'information, l'information du vecteur de nombres complexes d'origine est entrée dans le réseau sans aucune perte.

B. Données pilotes

On a envoyé à la station de base deux trames de signaux, la première trame est le pilote, la deuxième trame est les données. Ensuite, le point important à noter est - cette séquence pilote peut être arbitraire (1 0 0 1... ou 0 0 1 1 n'est pas important), mais elle doit être uniforme. Autrement dit, la séquence pilote dans la formation et les tests doit toujours être cohérente. Ceci est facile à comprendre-utilisez la même séquence pilote pour entraîner

Chapitre III : l'estimation et la détection

le réseau, puis dans le processus de formation, le réseau apprendra progressivement de cette séquence pilote et utilisera cette séquence pilote pour le traitement du signal.

Dans notre situation, la station de base envoie deux trames de données, d'un vecteur complexe 64×1 de donnée, c'est-à-dire 128 bits d'information. Ensuite, le plus simple est de spécifier que la sortie du réseau est un vecteur de nombre réel 128×1 , chaque élément correspondant à chaque bit restauré. L'approche du mémoire est qu'en utilisant 8 réseaux parallèles, l'entrée et la structure de chaque réseau soient exactement les mêmes, et ils sont responsables de la détection de 16 bits. Au total, les 128 bits sont détectés. Par exemple, le premier réseau détecte 1-16 bits, le second est 17-32 bits... Il y a deux points principaux à noter ici :

- a.** Bien que chaque réseau ne détecte que 16 bits, il entrera tous les signaux reçus. En effet, chaque bit du système OFDM est lié à tous les signaux reçus. Même si nous ne le considérons pas du point de vue de la communication, nous devrions donner toutes les informations disponibles au réseau pour traitement.
- b.** En fait, il est possible de détecter directement 128 bits avec un seul réseau, mais selon la pratique, la performance d'un réseau est le taux d'erreur binaire, ainsi que la performance sont bien pire que lorsqu'il est divisé en huit réseaux. Par conséquent, bien que la division en 8 réseaux augmente la complexité, les performances sont beaucoup améliorées.

Dans le code, un seul réseau doit être simulé au lieu de 8 réseaux parce que notre objectif est de calculer le taux d'erreur symbole, c'est la valeur moyenne des symboles, donc l'erreur de 1 réseau Le taux d'erreur de code est le même que 8 réseaux, donc le résultat de la simulation d'un réseau est le même que celui de 8 réseaux.

III.5. Les critères d'estimation

III.5.1. L'estimateur LS

L'estimateur de canal à moindres carrés LS est le modèle le plus simple puisqu'il consiste en une division du signal reçu sur le signal d'entrée, qui devrait être des symboles pilotes connus. On rappelle qu'au récepteur, les symboles observés à la sortie de la DFT sont :

$$Y = X * H + N$$

Où la matrice diagonale X contient les symboles transmis sur sa diagonale, et le vecteur H contient la réponse fréquentielle du canal.

L'estimateur LS minimise le paramètre J_{LS} . $(y - XF\hat{g}_{LS})^H (y - XF\hat{g}_{LS})$

Comme suit :

$$J_{LS} = (y - XF\hat{g}_{LS})^H (y - XF\hat{g}_{LS}) \quad (3.8)$$

$$J_{LS} = (y^H - \hat{g}_{LS}^H F^H X^H) (y - XF\hat{g}_{LS})$$

$$J_{LS} = (y^H y - \hat{g}_{LS}^H F^H X^H y - y^H X F \hat{g}_{LS} + \hat{g}_{LS}^H F^H X^H X F \hat{g}_{LS}) \quad (3.9)$$

Pour minimiser J_{LS} , nous devons faire sa dérivée de l'équation (3.9) par rapport à \hat{g}_{LS}^H , soit :

$$\partial J_{LS} / \partial \hat{g}_{LS}^H = [0 - F^H X^H y - 0 + F^H X^H X F \hat{g}_{LS}]$$

$$F^H X^H X F \hat{g}_{LS} = F^H X^H y$$

$$\hat{g}_{LS} = (F^H X^H X F)^{-1} F^H X^H y \quad (3.10)$$

Puisque les colonnes de F sont orthonormées, \hat{g}_{LS} génère l'estimation du canal \hat{H}_{LS} dans le domaine fréquentiel.

$$\hat{H}_{LS} = F \hat{g}_{LS} = F (F^H X^H X F)^{-1} F^H X^H y$$

$$\hat{H}_{LS} = F (F^H)^{-1} (X^H)^{-1} X^{-1} F^{-1} F^H X^H y$$

$$\hat{H}_{LS} = X^{-1} y \quad (3.11)$$

Les coefficients de l'estimateur LS sont donnés par la formule suivante :

$$\hat{H}_{LS} = \left[\frac{y_0}{X_0}, \frac{y_1}{X_1}, \dots, \frac{y_{N-1}}{X_{N-1}} \right]^T$$

L'avantage principal de cet estimateur est sa simplicité. Il exige seulement une simple division par sous-porteuse. L'inconvénient principal est qu'il a une grande erreur quadratique moyenne. Ceci est dû à l'utilisation d'un modèle de canal trop simplifié et que cet algorithme ne se sert pas des corrélations fréquentielle et temporelle du canal [10].

III.5.2. L'estimateur MMSE

L'estimateur MMSE utilise des statistiques des seconds ordres qui impliquent l'utilisation de la covariance automatique du canal dans afin de minimiser l'erreur carrée. Le critère du minimum d'erreur quadratique moyenne vise à minimiser la fonction de cout J_{MMSE} définie comme l'erreur quadratique moyenne entre le canal réel et estimé en utilisant la corrélation fréquentielle du canal :

$$J_{MMSE} = E[\|e\|^2] = E[\|(g - \hat{g}_{MMSE})\|^2] \quad (3.12)$$

$$J_{MMSE} = E[\|g - Ay\|^2]$$

$$J_{MMSE} = E[(g - Ay)(g - Ay)^H]$$

$$J_{MMSE} = E[(g - Ay)(g^H - y^H A^H)]$$

$$J_{MMSE} = E[gg^H - gy^H A^H - Ayg^H + Ayy^H A^H] \quad (3.13)$$

Pour minimiser J_{MMSE} on doit faire la dérivée de (3.13) par rapport à A^H , soit :

$$\partial J_{MMSE} / \partial A^H = 0 - E[gy^H] - 0 + A * E[yy^H] = 0$$

$$-R_{gy} + AR_{yy} = 0$$

$$A = R_{gy} R_{yy}^{-1}$$

$$\hat{g}_{MMSE} = Ay = R_{gy} R_{yy}^{-1} y \quad (3.14)$$

Et définie par :

$$R_{gy} = E[gy^H] = E[g(XFg + n)^H]$$

$$R_{gy} = E[gg^H F^H X^H] + E[gn^H] = E[gg^H F^H X^H] + 0$$

$$R_{gy} = R_{gg} F^H X^H \quad (3.15)$$

Et :

$$R_{yy} = E[yy^H] = E[(XFg + n)(XFg + n)^H]$$

$$R_{yy} = E[XFgg^H F^H X^H] + E[nn^H]$$

$$R_{yy} = XFR_{gg}F^H X^H + \sigma_n^2 I_N \quad (3.16)$$

$$R_{hh} = E[hh^H] = E[(Fg)(Fg)^H] = FE[gg^H]F^H$$

$$R_{hh} = FR_{gg}F^H$$

R_{gy} et R_{yy} sont respectivement la matrice de covariance entre g et y et la matrice d'auto covariance de y .

R_{gg} et R_{hh} sont respectivement les matrices d'auto covariance de g et h , et σ_n^2 est la variance du bruit. Puisque les colonnes de F sont orthonormées, \hat{g}_{MMSE} génère l'estimation du canal \hat{H}_{MMSE} dans le domaine fréquentiel.

$$\hat{H}_{MMSE} = F\hat{g}_{MMSE}$$

On trouve :

$$\hat{H}_{MMSE} = FR_{gy}R_{yy}^{-1}y \quad (3.17)$$

En remplaçant les équations (3.15) et (3.16) dans (3.17) on obtient :

$$\hat{H}_{MMSE} = FR_{gg}F^H X^H (XFR_{gg}F^H X^H + \sigma_n^2 I_N)^{-1}y$$

$$\hat{H}_{MMSE} = FR_{gg}F^H X^H (XFR_{gg}F^H + (\sigma_n^2 X^H)^{-1})^{-1}y$$

$$\begin{aligned} &= FR_{gg}F^H(XFR_{gg}F^H + \sigma_n^2(X^H)^{-1})^{-1}y \\ &= FR_{gg}F^H(FR_{gg}F^H + \sigma_n^2(X^H)^{-1}X^{-1})^{-1}X^{-1} \end{aligned}$$

Donc :

$$\hat{H}_{MMSE} = R_{hh}(R_{hh} + \sigma_n^2(XX^H)^{-1})^{-1}\hat{H}_{LS} \quad (3.18)$$

Comme on peut le voir, l'estimateur MMSE est beaucoup plus complexe que l'estimateur LS. Cela implique plus de multiplications et d'inversion matricielle. Le MMSE modifié réduit la complexité de l'estimateur MMSE en utilisant des méthodes telles que la décomposition en valeurs singulières.

III.6. Conclusion

Dans ce chapitre, nous avons étudié trois méthodes d'estimation de canal (MMSE, LS et DNN), dans les systèmes OFDM. La méthode d'apprentissage profond présente des avantages lorsque les canaux sans fil sont compliqués par une distorsion et une interférence car, le principe du modèle est que le nombre d'entrées correspond au nombre de données réelles et imaginaires des deux blocs OFDM qui contiennent des signaux pilotes et sont envoyés, chaque 16 bits des données transmises sont regroupés et prédits sur la base d'un seul modèle d'entraînement indépendant, puis ils sont connectés en série en tant que sortie finale. Ce qui prouve que les DNNs ont la capacité de se souvenir et d'analyser les caractéristiques complexes des canaux sans fil.

Chapitre IV :
Paramètres et résultats des
simulations

IV.1. Introduction

Ce chapitre est consacré à la partie simulation (paramètres de simulation, résultats et interprétation). Le logiciel de simulation utilisé est **Matlab**, les programmes réalisés peuvent être modifiés pour faire les tests et les comparaisons pour différents paramètres. Le but principal est d'examiner les performances des estimateurs du canal en termes d'erreur quadratique moyenne, le taux d'erreurs symboles et la complexité nécessaire. Pour cela une chaîne de transmission utilisant la modulation OFDM est simulée.

IV.2. Processus global

Dans le processus d'apprentissage hors ligne, on utilise d'abord la simulation pour obtenir un tas de données de formation. En fait, la simulation génère différents canaux et données de transmission (y compris pilote constant et données changeantes x), puis ajoute du bruit aux données de transmission à travers le canal pour obtenir le signal reçu y . Le signal reçu y est le signal d'entrée et les données x sont l'étiquette. $\{x, y\}$ peut être considéré comme un échantillon. Lors de la formation hors ligne, nous générons des milliers d'échantillons pour former le réseau. Avec l'itération de la formation, la fonction de perte devient de plus en plus petite, c'est-à-dire que les données récupérées par le réseau se rapprochent de plus en plus des données d'origine. Une fois la formation terminée, nous fixons les poids des paramètres dans le réseau et ne serons plus. Lors de l'utilisation en ligne, il nous suffit de faire passer le signal reçu à travers le réseau pour obtenir le signal récupéré.

Remarque : À l'heure actuelle, le poids du réseau a été fixé, de sorte que la complexité ne soit que de quelques opérations de multiplication matricielle, ce qui est très faible (la partie vraiment fastidieuse de la formation réseau peut être complétée à l'avance hors ligne, par exemple, vous pouvez passer du temps avant de l'utiliser. Le modèle est formé en une journée, et le réseau n'a pas besoin d'être formé lorsqu'il est réellement utilisé).

IV.3. Paramètres de la simulation

Nous avons mené plusieurs expériences pour démontrer la performance des méthodes d'apprentissage profond pour le canal commun d'estimation et de détection de symboles dans les systèmes de communication sans fil OFDM. Un modèle DNN est formé sur la base de la simulation donnée, et est comparé avec les méthodes traditionnelles en termes de taux d'erreur binaire (SER) sous différents rapports signal sur bruit (SNRs). Dans les expériences suivantes, l'approche basée sur le Deep Learning s'est avérée plus robuste que LS et MMSE dans les scénarios où moins de pilotes d'entraînement sont utilisés, le Préfix cyclique est omis, ou il y a un bruit d'écrêtage non linéaire. Dans notre expérience, un système OFDM avec 64 sous-porteuses et le CP de longueur 16 est considéré. Le canal sans fil suit la norme mondiale sans fil pour un nouveau modèle de radio (WINNER II) [9], où la fréquence porteuse est de 2,6 GHz, le nombre des chemins est 24, et les canaux urbains typiques avec le retard maximum 16 périodes d'échantillonnage sont utilisés. La QPSK est utilisé comme méthode de modulation.

IV.4. Le système OFDM

Il est utile de préciser les paramètres concernant le système OFDM que nous avons considéré dans ce travail:

a. Nombre de sous-porteuses :

Le système OFDM transmet les données sur plusieurs sous-porteuses. Plusieurs paramètres déterminent ou bien limitent le nombre des sous-porteuses, comme la largeur de bande, le débit binaire ainsi que la durée des symboles. On a choisi $N = 64$ (*porteuses*). Ce nombre dépend généralement de la bande de cohérence du canal, il faut diviser le canal en N sous-canaux de largeur inférieure à la bande de cohérence.

b. Modulation :

Plusieurs types de modulation peuvent être utilisés tels que, PSK, FSK et QAM. Pour le système OFDM, dans notre simulation on a utilisé la modulation QPSK. Le signal modulé est

Chapitre IV : Paramètres et résultats des simulations

représenté par un nombre complexe $C_k = a_k + jb_k$. Les deux valeurs a_k et b_k appartiennent à des ensembles bien déterminés.

Pour le cas du QPSK $a_k, b_k \in [1-1j; 1+1j; -1+1j; -1-1j]$. Chaque symbole est un regroupement de $\log_2(M)$ bits. Ce nombre varie selon la modulation et comme la modulation consiste à assigner à des mots binaires d'une taille $\log_2(M)$ des symboles complexes, l'utilisation du codage de Gray positionne les symboles d'une façon qu'à la réception, l'erreur détectée sur un symbole ne produit qu'une erreur d'un bit

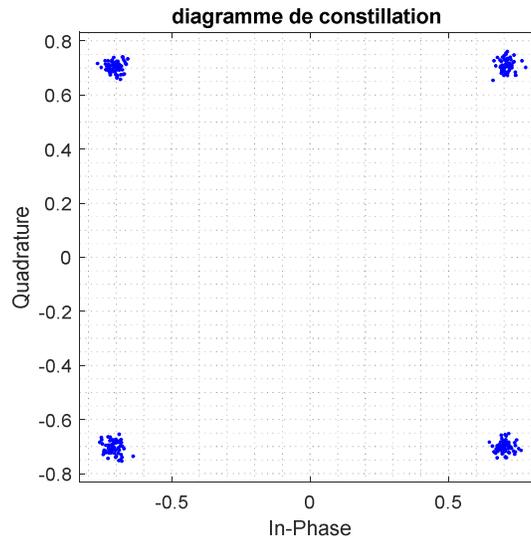


Figure IV.1 : Diagramme de constellation QPSK

c. Intervalle de garde :

L'intervalle de garde est choisi pour être plus grand que l'étalement du retard de canal, $T_g > \tau_{max}$ Pour éviter l'IEP et L'IES. Nous avons utilisé l'extension cyclique comme critère pour remplir cet intervalle de garde.

Dans ce travail, nous avons considéré aussi que :

- L'extension cyclique des symboles OFDM est utilisée comme intervalle de garde.
- La réponse impulsionnelle de canal est plus courte que la longueur du préfixe cyclique pour éviter IES.
- L'émetteur et le récepteur sont parfaitement synchronisés.

Chapitre IV : Paramètres et résultats des simulations

- On suppose que le canal demeure sans changement pendant la durée d'un symbole OFDM.

Le tableau reprend les paramètres de la chaîne de transmission OFDM que nous avons utilisés dans nos simulations

Paramètres	Spécifications
Nombre de sous-porteuses	64
Taille de l'FFT/IFFT	64
Modulation numérique	QPSK
Type d'intervalle de garde	Extension cyclique
Intervalle de garde	$N/4 = 16$
Nombre de trajets	20
Algorithme d'estimation	LS, MMSE, DNN

Tableau IV-1: Paramètres utilisés dans nos simulations.

IV.5. Estimation du canal

L'estimation du canal est basée sur des symboles pilotes transmis à certaines positions dans la grille temps-fréquence du système OFDM. Les atténuations du canal sont estimées à partir de ces pilotes. Les atténuations du canal dans les points voisins dans la grille temps-fréquence sont fortement corrélées, une propriété qui peut être utilisée par l'estimateur.

Le canal peut être estimé aux fréquences pilotes avec l'utilisation de l'un des algorithmes suivants :

- Moindres carrés (LS).
- L'estimateur linéaire à erreur quadratique moyenne minimale (MMSE).
- Deep Learning (DNN).

IV.6. Critères d'évaluation

Pour évaluer les performances de la chaîne OFDM et l'estimation du canal, nous avons utilisé les critères d'évaluation suivants :

a. **Le taux d'erreur par bit (BER) :**

Le *BER* est calculé à partir de l'équation suivante :

$$BER = \frac{\text{nombre total de bits erroné}}{\text{nombre total de bits transmis}}$$

b. **Le taux d'erreur par symbole (SER) :**

On peut aussi, calculer le *SER*, le taux d'erreur par symbole.

$$SER = \frac{\text{nombre total de symboles erroné}}{\text{nombre total de symboles transmis}}$$

IV.7. Arrangement des pilotes

Dans nos simulations, on a utilisé un arrangement de pilotes type blocs qui sont approprié au modèle de canal statique sélectif en fréquence.

IV.8. Paramètre de formation

Paramètre	Spécification
Temps de d'apprentissage	50s
Durée	100
Itérations par durée	8
Itération	800
Max itération	800
Taux d'apprentissage	0.01

Tableau IV-2 : Paramètres utilisés dans la formation

Chapitre IV : Paramètres et résultats des simulations

*Itérations = Itérations par époque * Nombre d'époques*

IV.9. Condition d'arrêt dans l'apprentissage :

- Après un certain nombre d'itération car :
L'erreur de sortie totale *devient* inférieure à un seuil acceptable [0,0.2]
Si le nombre est trop faible, l'erreur n'est pas suffisamment réduite.
Si le nombre est grand, surapprentissage
- Lorsque les poids se stabilisent
- Lorsque l'erreur descend en dessous d'une certaine borne

IV.9. Résultats des simulations

IV.9.1. Impact de CP

Le CP est nécessaire pour convertir la convolution linéaire du canal physique en convolution circulaire et atténuer l'ISI. Mais cela coûte du temps et de l'énergie pour la transmission. Dans cette expérience, nous allons étudier les performances avec CP.

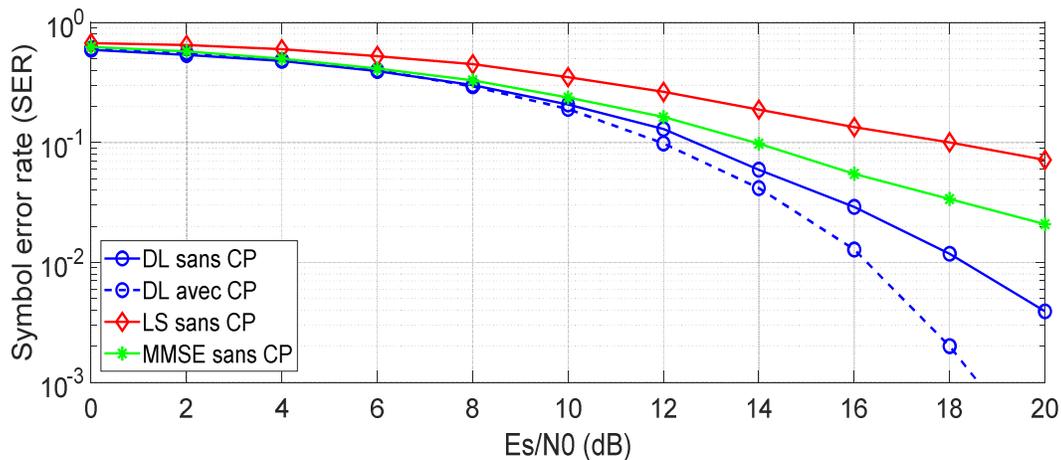


Figure IV.2 : Courbes SER SANS CP

Chapitre IV : Paramètres et résultats des simulations

La figure IV.2 illustre les courbes SER pour un système OFDM sans CP. À partir de cette figure, ni MMSE ni LS ne peuvent estimer efficacement le canal. La précision a tendance à être saturée lorsque le SNR est supérieur à 14 dB. Cependant, la méthode d'apprentissage profond fonctionne toujours bien. Ce résultat montre à nouveau que les caractéristiques du canal sans fil ont été révélées et peuvent être apprises au stade de l'apprentissage par les DNN.

IV.9.2. Impact des numéros pilotes

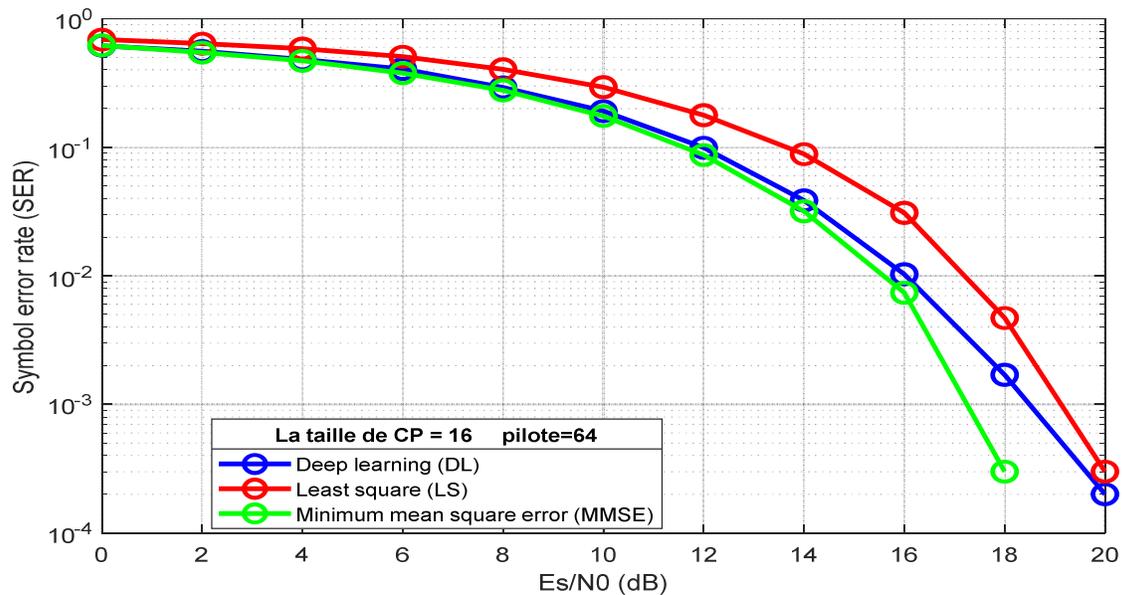


Figure IV.3 : Courbes SER avec 64 pilotes

La méthode proposée est d'abord comparée aux méthodes LS et MMSE pour l'estimation et la détection des canaux, lorsque 64 pilotes sont utilisés pour l'estimation des canaux dans chaque trame. De la Fig IV.3, la méthode LS a les pires performances car aucune statistique préalable du canal n'a été utilisée dans la détection. Au contraire, la méthode MMSE présente les meilleures performances car les statistiques de second ordre des canaux sont supposées connues et utilisées pour la détection de symboles. L'approche basée sur l'apprentissage profond a de bien meilleures performances que la méthode LS et est comparable à la méthode MMSE.

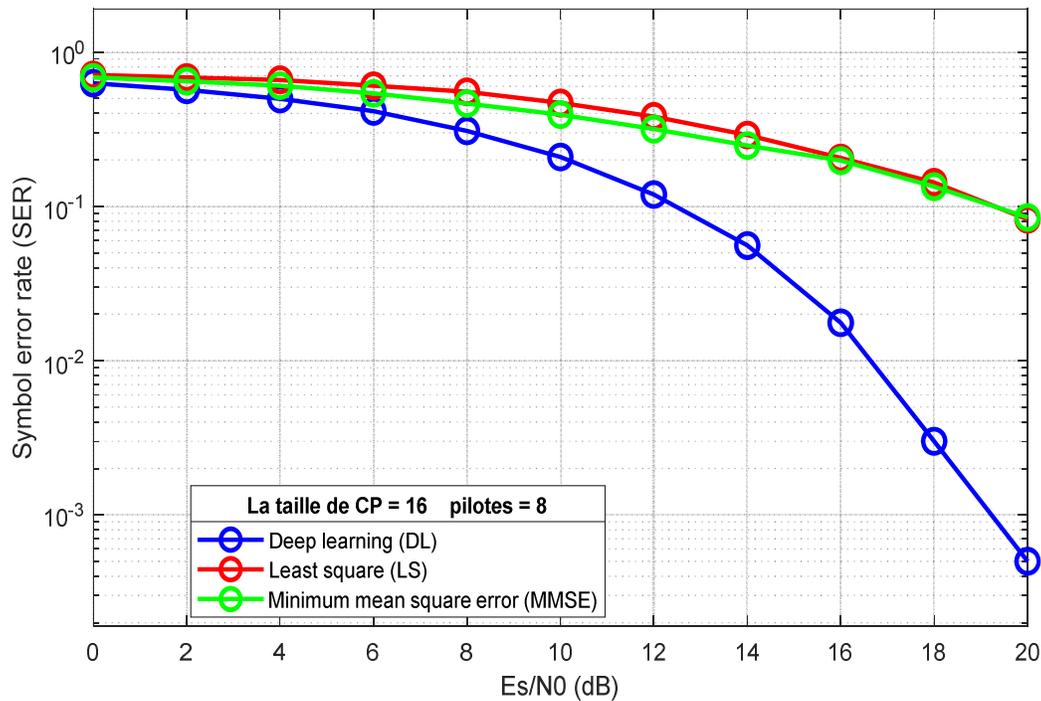


Figure IV.4 : Courbes SER avec 8 pilotes

Étant donné que le modèle de canal a un délai maximum de 16, il peut être estimé avec beaucoup moins de pilotes, ce qui conduit à une meilleure utilisation du spectre. De la Fig IV.4, lorsque seulement 8 pilotes sont utilisés, les courbes SER des méthodes LS et MMSE saturent lorsque le SNR est supérieur à 10 dB. Par contre l'apprentissage profond donne toujours des bonnes performances.

IV.9.3. Comparaison entre les estimateurs a 64 et 8 pilotes

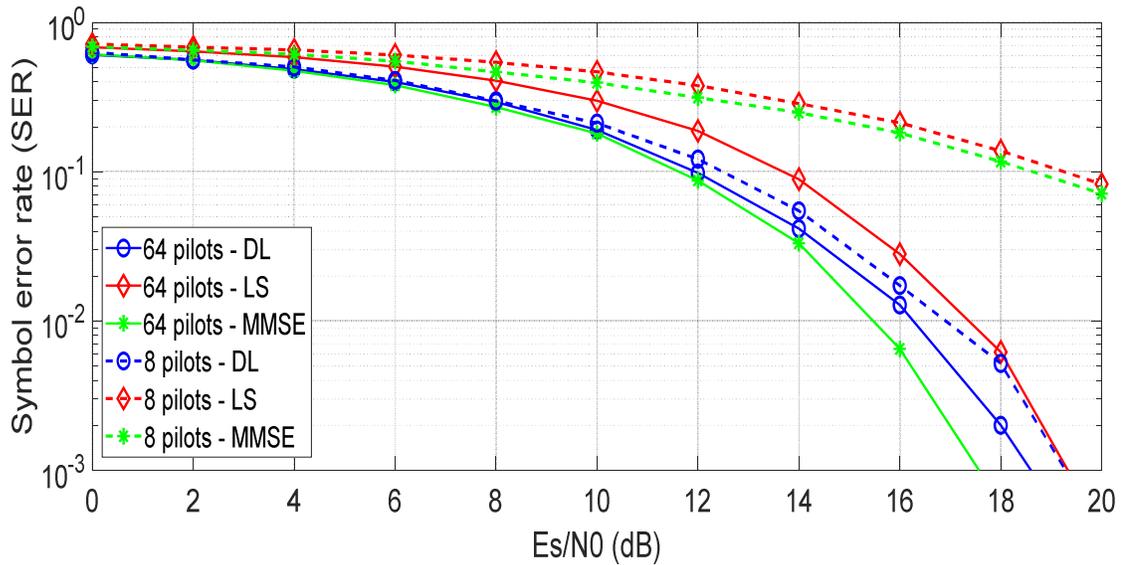


Figure IV.5 : Courbes SER basé sur l'apprentissage profond et les méthodes classiques

D'après cette figure, on constate que le nombre de pilotes a de l'impact sur les méthodes classiques (MMSE, LS) il y a une relation directe entre ces deux, car lorsque on augmente le nombre des pilotes les performances augmentent tandis que la méthode basée sur l'apprentissage profond a toujours la capacité de réduire son SER avec l'augmentation du SNR, ce qui démontre que le DNN est robuste au nombre de pilotes utilisés pour l'estimation du canal. La raison de la performance supérieure du DNN est que le CSI n'est pas distribué uniformément. Les caractéristiques des canaux sans fil peuvent être apprises sur la base des données d'apprentissage générées à partir du modèle.

IV.10. Conclusion

Dans ce dernier chapitre nous avons détaillé les paramètres et présenté les résultats de nos simulations. On a fait une comparaison entre les estimateurs classiques (LS, MMSE) et l'apprentissage profond et nous avons constaté que l'apprentissage profond est meilleur et le plus performant d'entre eux. Les résultats de nos simulations montrent que la méthode d'apprentissage profond présente des avantages lorsque les canaux sans fil sont compliqués par de graves distorsions et interférences, ce qui prouve que les DNN ont la capacité de mémoriser et d'analyser les caractéristiques compliquées des canaux sans fil.

Conclusion générale

Le but de ce mémoire était l'étude de l'estimation du canal et la détection de symbole basée sur l'apprentissage profond pour les systèmes OFDM. Nous avons donc comparé les estimateurs classiques avec l'apprentissage profond par simulations en utilisant le logiciel Matlab. On n'a remarqué que l'apprentissage profond présente de meilleures performances. Nous avons démontré nos efforts initiaux pour utiliser des DNN pour l'estimation de canal et la détection de symbole dans un système OFDM.

Le modèle est formé hors ligne sur la base des données simulées qui visualisent l'OFDM et les canaux sans-fil comme des boîtes noires. Les résultats de la simulation montrent que la méthode d'apprentissage profond présente des avantages lorsque les canaux sans fil sont compliqués par de graves distorsions et interférences, ce qui prouve que les DNN ont la capacité de mémoriser et d'analyser les caractéristiques compliquées des canaux sans fil.

Pour les applications réelles, il est important que le modèle DNN ait une bonne capacité de généralisation afin qu'il puisse toujours fonctionner efficacement lorsque les conditions de déploiement en ligne ne correspondent pas exactement aux modèles de canaux utilisés au stade de la formation.

La première expérience a été menée pour illustrer la capacité de généralisation du modèle DNN par rapport à certains paramètres du modèle de canal. Des analyses plus rigoureuses et des expériences plus complètes peuvent être faites dans des travaux futurs. En outre, pour une utilisation pratique, des échantillons générés à partir des canaux sans fil réels pourraient être collectés pour affiner le modèle et avoir de meilleures performances.

Bibliographie

- [1] Meghyn Bienvenu. Notes du cours Introduction à l'Intelligence Artificielle, Peter Norvig, 2007
- [2] <URL> : http://www.intelligenceartificielle.fr/domaines_IA.php. Consulté le mars 2021.
- [3]
<https://www.europarl.europa.eu/news/fr/headlines/society/20200827STO85804/intelligence-artificielle-definition-et-utilisation>, Intelligence artificielle créé 09/2020, consulté le mars 2021.
- [4]<URL> : <http://tazukuzanre54.vitekivpddns.com/546.html>. consulté mars 2021
- [5]<URL>: <https://openclassrooms.com/fr/courses/4011851-initiez-vous-au-machine-learning/4020611-identifiez-les-differents-types-dapprentissage-automatiques>
- [6] BITTENCOURT, Guilhermehistória da Inteligência Artificial. p. 32, 2001
- [7]<URL>: <http://neuralnetworksanddeeplearning.com/chap2.html>, How the backpropagation work , consulté mars 2021
- [8] HINTON, E .Geoffrey; SALAKHUTDINOV, Ruslan R. Reduzindo a dimensionalidade dos dados com redes neurais. ciência , v. 313, n. 5786, p. 504-507, 2006
- [9] <URL>: <http://media.nips.cc/Conferences/2009/NIPS-2009-Workshop-Book.pdf>, NIPS. 2009. Disponível em, consulté avril 2021
- [10] J. C. Rault, D. Castelain et B. Le Floch, «The Coded Orthogonal Frequency Division Multiplexing (COFDM) Technique, and its Application to Digital Radio Broadcasting Towards Mobile Receivers», IEEE Global Télécommunications Conférence, Vol. 1, pp. 428-432, novembre 1989.
- [11] Santana, Marlesson. Deep Learning: do Conceito às Aplicações. 2018

- [12] <URL>:<https://medium.com/data-hackers/deep-learning-do-conceito-%C3%A0s-aplica%C3%A7%C3%B5es-e8e91a7c7eaf>, Marlesson Santana 19 juillet 2018, consulté le mai 2021
- [13] DETTMERS, Tim. *Aprendizagem profunda e poucas palavras: Historia e Treinamento*. Disponível em, <URL>:<https://devblogs.nvidia.com/deep-learning-nutshell-history-training/>, consulté mai 2021
- [14] <URL> :<<https://www.aliger.com.br/blog/saiba-o-que-e-visao-computacional/>>. Site consulté mai 2021
- [15] Parizeau, Marc. Réseaux de Neurones (Le perceptron multicouche et son algorithme de rétropropagation des erreurs), Université Laval, 2004.
- [16] S.C.Cho, J.Kim, Y.Y.Yang, & C.G.Kang, (2010). MIMO-OFDM wireless Communications with MATLAB. Singapore: Wiley, August 2010
- [17] Alshammari, A., Albdran, S., & Matin, M. Study of bit error rate (BER) for multicarrier OFDM. Proc. SPIE. San Diego. 2012
- [18] S Alabdran, A. Alshammari, , M. Matin, Clipping and Filtering Technique for reducing PAPR In OFDM. IOSR Journal of Engineering, 2(9), pp.91-97, (September 2012).
- [19] A. Bahai, B. Saltzberg and M. Ergen, Multi carrier digital communications: Theory and applications of OFDM, Springer Science, New York, 2004.
- [20] Rappaport, T. S. Wireless communication: principles and practice. Upper Saddle River, NJ: Prentice Hall. 1996
- [21] Henrik Schulze, Christian Lüders. Scholtz, "Theory and Applications of OFDM and CDMA". Edition J. Wiley and sons, 2005.

- [22] L. Hanzo, T. Keller, « OFDM and MC-CDMA for Broadband multiuser communications, WLAN and Broadcasting », Edition John Wiley and Sons, 2nd edition, ISBN-13 978-0-470-03007-3, 2006.
- [23] W. Akmouche., « Etude et caractérisation des modulations multiporteuses OFDM », PhD thesis in electrical engineering, Université de Bretagne Occidentale, October 2000.
- [24] Langton, C. Orthogonal frequency division multiplex (OFDM) Tutorial. (2004). Retrieved November 4, 2012.
- [25] J. C. Rault, D. Castelain et B. Le Floch, « The Coded Orthogonal Frequency Division Multiplexing (COFDM) Technique, and its Application to Digital Radio Broadcasting Towards Mobile Receivers », IEEE Global Télécommunications Conférence, Vol. 1, pp. 428-432, novembre 1989
- [26] Li, X., & Cimini, L. Effects of clipping and filtering on the performance of OFDM. pp. IEEE Proceedings VTC (1634-1638. I), 1997
- [27] May, T. Rohling, H. Reducing the peak to average power ratio of OFDM Radio Transmission Systems. Proceedings of IEEE VTC '98 (pp. 2474-2478). Ottawa, Canada, 1998.
- [28] Ye Hao, Ye Li Geoffrey, and Bing-Hwang Fred Juang: power of deep learning for channel estimation and signal detection in OFDM system, pp1-3, aout 2017.
- [29] TOM KELDENICH, F. Chollet, *Deep Learning with Python*, 2017
- [30] Phil Kim Seoul, Soul-t'ukpyolsi, Korea (Republic of) MATLAB Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence, 2017.