

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث 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 du MASTER

Filière Télécommunication

Spécialité Réseaux et télécommunications

Présenté par :

DAOUDI MERIEM

&

MOKHBAT MERIEM

Réduction du PAPR dans les systèmes OFDM par le Deep Learning

Proposé par :

Dr. AIT SAADI Hocine

Année Universitaire : 2020-2021

On tient à exprimer nos plus vifs remerciement à notre promoteur **Monsieur Pit Saadi Kocine** pour son aide, pour ses précieux conseils constructifs et pour le concours qu'il nous a prêté, ainsi que ses critiques et orientations qu'il nous a prodigués en contribuant à l'élaboration de ce modeste travail.

Nous tenons à remercier les membres du jury, **Madame Zine Leila** et **Monsieur Kebib Sami** pour l'honneur qu'ils ont bien voulu nous faire par leur présence.

Nous exprimons notre profonde reconnaissance à nos parents, à nos frères et sœurs qui nous ont soutenus et aidés durant la période de notre formation et surtout à la réalisation de ce modeste travail.

Que nos amis de la promotion, trouvent l'expression de notre profonde reconnaissance pour leurs aides qui nous a été très utiles.

Enfin, nous remercions nos enseignant de **l'Université Saad Dahleb Blida (USDB)** pour la formation qu'ils nous ont dispensée.

Je dédie ce modeste travail

- *À la mémoire de mes grands-parents ;*
- *À ma grand-mère maternelle ;*
- *À mes chers parents ;*
- *À ma chère sœur Selma ;*
- *À mes tantes et oncles ;*
- *À mes cousines et cousins ;*
- *À mon binôme DAOUDI MERIEM ;*
- *À toutes mes amies, surtout SARAH, ABIR, ASMA, HADIL, YASMINE, MANEL et FELLA ;*
- *À tous ceux qui me sont chers.*

MOKHBAT Meriem

مخباط مريم

Je dédie ce modeste travail

- *À mes parents pour l'éducation exemplaire qu'ils ont su me donner, et qui m'a permis avec l'aide de DIEU d'atteindre mes objectifs ;*
- *À mes deux frères Sadek et Aymen, et ma sœur Manar ;*
- *À mes tantes et oncles ;*
- *À mes cousines et cousins ;*
- *À mon binôme MOKHBAT Meriem ;*
- *À tous mes amis.*

DAOUDI Meriem

داودي مريم

ملخص

OFDM "متعددة الترددات متعامد شعبة" هو تقنية متطورة متعددة الموجات التشكيل التي أثبتت نفسها في مجال الاتصالات اللاسلكية مما يسمح للتكنولوجيات الحالية لتحقيق معدل نقل عالية. يتم استخدامه في معظم معايير الاتصال، وهذه التقنية لها مزايا وعيوب. ومن المساوئ الرئيسية أن إشارات OFDM لها تقلبات قوية في المغلف تتميز بارتفاع "عامل الذروة". "عامل الذروة" هو ما يعادل PAPR "نسبة الذروة إلى المتوسط السلطة"، والهدف من هذه الأطروحة هو جعل الحد من PAPR عن طريق التعلم العميق باستخدام طريقة TRnet وإجراء مقارنة بين طريقة TR الكلاسيكية وطريقة التعلم العميق.

كلمات المفتاحية: OFDM ,PAPR ,TRnet ,TR

Résumé

L'OFDM "Orthogonal Frequency Division Multiplexing" est une technique de modulation multiporteuse sophistiquée qui a fait ses preuves dans le domaine de la communication sans fil permettant aux technologies actuelles d'atteindre un débit de transmission élevé. Elle est utilisée dans la plupart des standards de communication, cette technique présente des avantages ainsi que des inconvénients. Un des principaux inconvénients est que les signaux OFDM ont une forte fluctuation d'enveloppe qui est caractérisée par un "facteur de crête" élevé. Le "facteur de crête" est équivalent au PAPR « Peak-to-Average Power Ratio », L'objectif de ce mémoire est de faire la réduction de PAPR par la méthode tone reservation en utilisant l'apprentissage profond.

Mots clés : OFDM, PAPR, TRnet, TR

Abstract

OFDM "Orthogonal Frequency Division Multiplexing" is a sophisticated multicarrier modulation technique that has proven it self in the field of wireless communication allowing current technologies to achieve a high transmission rate. It is used in most communication standards, this technique has advantages as well as disadvantages. One of the main disadvantages is that OFDM signals have a strong envelope fluctuation, which is characterized by a high "peak factor". The "peak factor" is equivalent to the PAPR "Peak-to-Average Power Ratio", The objective of this master dissertation is to make a PAPR reduction with Tone Reservation method by using a deep learning.

Keywords : OFDM, PAPR, TRnet,TR

Liste des abréviations

IA	:	Intelligence artificielle.
CNN	:	Convolutional Neural Network.
RNN	:	Recurrent Neural Network.
ReLU	:	Rectifiedlinear Unit.
Tanh	:	Tangente hyperbolique.
OFDM	:	Orthogonal Frequency Division Multiplexing.
PAPR	:	Peak-to-Average Power Ratio.
HPA	:	Amplificateur a haute puissance.
CP	:	Préfixe cyclique.
IES	:	interférence entre symboles.
IEP	:	interférence entre porteuses.
TFD	:	transformée de fourier discrète directe.
TFDi	:	Transformée de fourier discrète indirecte.
FFT	:	FastFourier Transform.
IFFT	:	inverse Fast Fourier Transform.
SNR	:	Rapport Signal à bruit.
QAM	:	Modulation d'amplitude en quadrature.
QPASK	:	Quadrature Phase Shift Keying.
CDF	:	Cumulative Distribution Function.
CCDF	:	Complementary Cumulative Distribution Function.
TEB	:	Taux d'erreurs binaire.
TR	:	Tone Reservation.
ACE	:	Active Constellation Extension.
TI	:	Tone Injection.
TRnet	:	Tone Reservation network.
FFNN	:	FeedForward Neural Network.
DNN	:	Deep Neural Network.
SGD	:	Descent de fradiant stochastique.

Table des matières

Introduction Générale	1
Chapitre 1 Deep Learning.....	2
1.1 Introduction :	4
1.2 L'intelligence artificielle :	4
1.2.1 Test de Turing :	5
1.2.2 Une petite histoire sur L'intelligence artificielle (IA) :	5
1.2.3 Sous domaine de l'intelligence artificielle(IA) :	7
Représentation des connaissances et Raisonnement Automatique :	7
Résolution de problèmes généraux :	7
Traitement du langage naturel :	8
Vision artificielle :	8
Robotique :	8
Apprentissage automatique :	8
1.2.4 État de l'art de l'intelligence artificielle :	8
1.3 Machine Learning :	9
1.3.1 Type d'apprentissage :	9
1.3.2 Le fonctionnement du machine learning :	10
1.4 Faut-il avoir peur de l'intelligence artificielle ?	10
1.5 Les réseaux de neurones :	11
1.5.1 MODELE BIOLOGIQUE :	11
1.5.2 Principe de base :	13
La fonction d'activation :	14
1.5.3 Les types de réseaux neuronaux :	15
Réseaux de neurones convolutifs CNN :	15
Réseaux de neurones récurrents(RNN) :	15
Les réseaux de neurones à propagation directe (feed-forwarded) :	16
1.5.3.1.1 Exemple de réseau à deux couches :	17
1.5.4 Apprentissage d'un réseau de neurones :	19
1.5.5 Applications :	19
1.6 Deep learning :	20
1.6.1 Définition :	20
1.6.2 Pourquoi deep learning et pourquoi nous étudions cela maintenant ?	20
1.6.3 Les applications du Deep Learning :	21
La reconnaissance faciale :	21
Le traitement automatique de langage naturel :	21
Voitures autonomes :	21

Table des matières

Recherche vocale et assistants à commande vocale :	21
Ajout automatique de sons à des films muets :	22
Traduction automatique :	22
Génération automatique de texte :	22
Reconnaissance d'image :	23
La description automatique d'image :	23
Colorisation automatique :	23
La détection du cancer du cerveau :	24
Analyse des sentiments du texte :	24
Recherche en marketing :	24
1.6.4 Les applications de Deep learning en télécom :	24
1.7 Conclusion	25
Chapitre 2 System OFDM et PAPR	26
2.1 Introduction :	27
2.2 La modulation OFDM :	27
2.3 Principe de l'OFDM :	28
2.3.1 La chaîne de transmission OFDM :	29
2.3.2 Principes du modulateur et du démodulateur utilisant les transformées de Fourier discrètes :	30
2.3.3.1 La transformée de Fourier discrète :	30
2.3.3 L'orthogonalité :	31
2.3.4 L'interférence entre symboles (IES) :	32
2.3.5 L'interférence entre porteuses (IEP) :	33
2.4 Intervalle de garde :	33
2.5 les avantages et les inconvénients de l'OFDM :	34
2.5.1 Les avantages :	34
2.5.2 Les inconvénients :	35
Le décalage fréquentiel :	35
Le PAPR :	35
2.6 Problème du PAPR lors de l'amplification de puissance :	36
2.7 Fonction de répartition du PAPR pour l'OFDM :	36
2.8 Les techniques de réduction de PAPR :	37
2.9 Critère d'évaluation des techniques de réduction du PAPR :	37
Compatibilité descendante :	37
Dégradation du taux d'erreurs binaire (TEB) :	37
Diminution du débit utile :	37

Table des matières

Complexité :	37
2.10 La réduction de PAPR :	38
2.11 Technique TR :	38
Avantage :	40
Inconvénients :	40
2.12 Conclusion :	40
Chapitre 3 Réduction du PAPR avec la méthode	41
3.1 Introduction :	42
3.2 Les modèles de système et FFNN :	42
3.2.1 Modèle du système :	42
3.2.2 FFNN :	43
3.2.2.1 La fonction de perte :	43
3.2.2.2 Les fonctions d'optimisation :	44
3.2.2.3 Descente de gradient :	44
3.2.2.1.1 Les algorithmes d'optimisation de la descente de gradient :	45
3.3 TR-NET :	49
3.4 Structure du TR-NET :	50
La couche FC :	50
La couche BN :	50
L'algorithme dropout :	52
La fonction tanh :	52
3.5 Prétraitement des données :	53
3.6 Formation de TR-NET :	54
3.7 Conclusion :	54
Chapitre 4 Paramètres et résultats des simulations	55
4.1 Introduction :	56
4.2 Les effets du non linéarité de l'amplificateur sur le signal OFDM :	56
4.3 Technique de réduction :	58
4.4 La technique TR :	58
4.5 La méthode TR avec l'algorithme de programmation linéaire (TR-LP) :	58
4.6 Méthode TR-NET	60
4.7 Conclusion :	62
Conclusion Générale :	63
Bibliographie :	64

Liste des Figure :

Figure 1. 1: Deep Learning	5
Figure 1. 2: Schéma d'un neurone	12
Figure 1. 3 : Architecture d'un réseau de neurone	12
Figure 1. 4: Modélisation d'un neurone artificiel	13
Figure 1. 5: Les classifieurs	16
Figure 1. 6: Deux types de perceptrons	17
Figure 1. 7 : Exemple de réseau à deux couches	18
Figure 1. 8: La fonction ReLU	18
Figure 1. 9 : La différence entre réseau de neurone simple et profond	20
Figure 2.1 : Les blocs d'un système OFDM	28
Figure 2.2 : Schéma de fonctionnement de l'OFDM	29
Figure 2.3 : Forme d'un spectre du signal OFDM	32
Figure 2.4 : Intervalle de garde pour un système OFDM	34
Figure 2.5 : Principe de fonctionnement de TR	39
Figure 3. 1 : Le schéma de principe du codeur FFNN	42
Figure 3. 2 : La representation de la fonction de perte	44
Figure 3. 3 : Courbe montrant les positions des minima [15]	48
Figure 3. 4 : Vue d'ensemble du TRNet	49
Figure 4. 1 : spectre sans et avec amplification par un modèle SSPA pour différente valeur de C	56
Figure 4. 2 : La fonction CCDF d'un signal OFDM	57
Figure 4. 3 : Les fonctions CCDF du PAPR des signaux OFDM avec la méthode TR-LP	59
Figure 4. 4: CCDF de la méthode TR-NET avec $u=8$, $N=128$	62

Liste des tableaux :

Tableau 4. 1: Paramètres de simulation	58
--	----

Introduction générale :

Dans le contexte actuel et futur des systèmes de communications mobiles, la transmission de données à très haut débit est essentielle pour de nombreux services comme la vidéo, l'audio haute qualité et mobile numérique à intégration de service réseau. Lorsque les données sont transmises à très haut débit, sur des canaux de radio mobiles, le canal à réponse impulsionnelle peut s'étendre de nombreux symboles périodes, ce qui conduit à des interférences entre les symboles (ISI).

L'OFDM (Orthogonal Frequency Division Multiplex) est l'un des candidats promoteurs pour atténuer l'ISI. Dans un signal OFDM la bande passante est divisée en plusieurs sous-porteuses étroites, qui sont transmises en parallèle. Chaque sous-porteuse généralement choisie, est suffisamment étroite pour éliminer l'effet de retard de la propagation.

En revanche, un des principaux inconvénients est que les signaux OFDM ont une forte fluctuation d'enveloppe qui est caractérisée par un "facteur de crête" élevé. Le "facteur de crête" est équivalent au PAPR ("Peak-to-Average Power Ratio"). Le PAPR rend les signaux OFDM très sensibles aux non-linéarités des composants analogiques, en particulier celles de l'amplificateur de puissance.

Pour résoudre le problème du PAPR plusieurs techniques ont été proposées, à savoir la technique selective mapping, la technique de codage reed-muller et la technique tone réservation. Cette dernière a fait l'objet d'une attention considérable.

Dans notre cas la réduction du PAPR va se faire en utilisant l'apprentissage profond par un algorithme de type non supervisé. Cette technique se nomme le TR-NET (Tone Reservation Network)

L'apprentissage profond a montré de bonnes performances dans :

- L'estimation du canal ;
- La détection du signal ;
- Le décodage du canal ;
- La reconnaissance de la modulation ;
- Les systèmes sans fil de bout en bout.

De ce fait le but de notre travail est de régler la distorsion du signal dans le cas où, ce dernier dépasse le seuil (le pic), c'est-à-dire diminuer le facteur de crête.

Pour cerner ce rôle, notre approche va tourner autour des questions suivantes :

- Qu'est-ce que le deep learning ?
- Qu'elle est la différence entre la méthode TR et la méthode TR-NET ?

Pour ce faire, ce travail est divisé en quatre chapitres :

Dans le 1er chapitre, nous décrivons brièvement l'apprentissage profond, l'intelligence artificielle, ainsi que les types de réseaux neuronaux.

Dans le 2eme chapitre, nous présentons le principe de modulations multi-porteuses OFDM ainsi que sur l'effet des blocs tels que l'intervalle de garde et l'IFFT le problème du facteur de crête (PAPR) et la méthode de réduction TR.

Dans le 3ème chapitre, on parle sur les étapes de la réduction du PAPR avec la méthode TRnet ainsi que les caractéristiques du FFNN.

Dans le dernier chapitre, nous présentons les résultats de nos simulations et la comparaison entre les deux méthodes, à savoir, la méthode classique TR et la méthode de deep learning.

Nous clôturons ce mémoire par une conclusion générale.

Chapitre 1

Deep Learning

1.1 Introduction :

Depuis quelques années, on associe presque toujours l'intelligence aux capacités d'apprentissage. C'est grâce à l'apprentissage qu'un système intelligent capable d'exécuter une tâche peut améliorer ses performances avec l'expérience. C'est grâce à l'apprentissage qu'il pourra apprendre à exécuter de nouvelles tâches et acquérir de nouvelles compétences.

1.2 L'intelligence artificielle :

L'intelligence artificielle (IA) est la pratique qui tend à faire penser les machines, c'est un ensemble de technique visant à tenter d'approcher le raisonnement humain. À l'avenir c'est elle qui nous permettra de nous déplacer dans des voitures autonomes et d'avoir des robots domestiques [1-2]. Elle repose sur des algorithmes pour obtenir un résultat

On peut catégoriser l'IA de quatre façons [2] :

- **Se comporter comme un être humain** : elle reflète mieux le « test de Turing » selon lequel une machine est considérée comme intelligente si elle peut converser de telle manière que les interrogateurs.
- **Penser comme un être humain** : si l'on attache l'importance à cette définition, cela implique que l'IA est une science expérimentale, car pour déterminer si un programme pense comme un humain on devra d'abord avoir une méthode sur comment les humains pensent.
- **Penser rationnellement** : dans ce cas, les systèmes doivent raisonner d'une manière rationnelle, c'est à dire en suivant les lois de la logique. Cette approche peut être critiquée car il semble que certaines capacités (la perception, par exemple) ne sont pas facilement exprimables en logique. De plus, ce standard de rationalité ne peut pas être atteint en pratique car la technologie actuelle ne permet pas de réaliser des calculs aussi complexes.
- **Se comporter rationnellement** : concerne le développement des agents qui agissent pour mieux satisfaire leurs objectifs. On remarque que cette définition est plus générale que la précédente, car raisonner logiquement peut être une façon d'agir rationnellement.

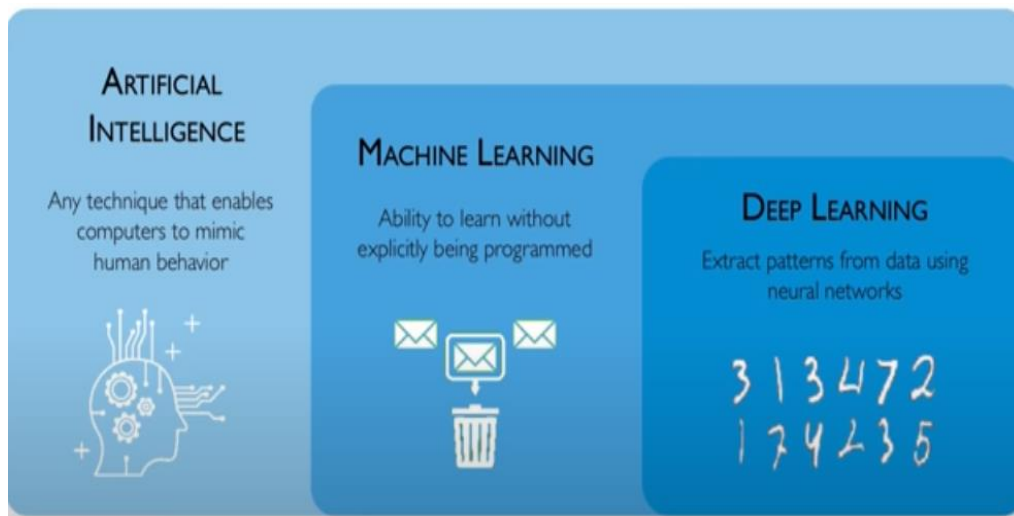


Figure 1. 1: Deep Learning

1.2.1 Test de Turing :

Son principe est simple, un évaluateur humain consiste à tester si une machine peut être au moins aussi intelligente qu'un humain. Si l'évaluateur ne sait pas distinguer l'homme de la machine alors la machine a passé le test avec succès [2].

Ce teste fut imaginé par le mathématicien et cryptologue « Alan Turing » dans une publication de 1950 Ce test fut à la fois très critiqué et très influent. Au fil des années, il est devenu un concept très important dans le domaine de la philosophie de l'intelligence artificielle

1.2.2 Une petite histoire sur L'intelligence artificielle (IA) :

L'histoire de L'intelligence artificielle (IA) est passée par des étapes importantes, en voici quelques une [1] :

➤ La période 1943-1955, appelée gestation de l'IA, était considérée comme les débuts de l'intelligence artificielle (même si le terme n'existait pas encore). Les travaux de Mcculloch et pitts ont introduit un modèle de neurones artificiels en 1943. Quelques années après, Hebb proposa une règle pour modifier des connections entre neurones. Minsky et Edmonds construisirent le premier réseau de neurones. Ce fut aussi durant cette période que Turing publia son fameux article dans lequel introduit le test de Turing.

- Durant l'année 1956, qui est considérée comme l'année de naissance de l'IA, a permis à un petit groupe d'informaticiens intéressés par l'étude de l'IA, de se réunir pour une conférence sur ce thème. Cette conférence permit de poser les fondements de l'IA.
- La période 1952-1969, fut une période très active pour le domaine de l'IA, car un grand nombre de programmes furent développés pour résoudre des problèmes d'une grande diversité. McCarthy publia un article devenu célèbre dans lequel, il traite des programmes qui ont du sens commun. La recherche sur les réseaux de neurones fut également poursuivie. Ce fut aussi l'époque du Shakey, le premier robot à être capable de raisonner sur ses propres actions.
- La période 1966-1973, de grandes déceptions se produisirent durant ces années. Des chercheurs réalisèrent un traducteur automatique, mais se sont vite rendu compte que leur approche purement syntaxique n'était pas suffisante (pour bien traduire un texte, il faut d'abord le comprendre). Cet échec a provoqué l'annulation en 1966 de tout le financement du gouvernement américain 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, et découvrirent alors qu'ils ne fonctionnaient pas, par manque de mémoire et de puissance de calcul. Ce fut une des critiques adressée à l'IA dans le rapport de Lighthill de 1973, qui provoqua l'arrêt du financement de la quasi-totalité des projets en IA de Grande Bretagne. Et comme si cela ne suffisait pas, Minsky et Papert prouvèrent dans leur livre « Perceptrons » de 1969 que les réseaux de neurones de l'époque ne pouvaient pas calculer certaines fonctions pourtant très simples, ce qui mit en cause toute la recherche en apprentissage automatique, entraînant une crise dans cette branche de l'IA.
- La période 1969-1979, fut la période des Systèmes Experts. Le premier système expert, appelé DENDRAL, fut créé en 1969 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 élaborées par des experts humains. Après le succès du DENDRAL, d'autres Systèmes Experts furent créés, notamment le système MYCIN, qui réalisait un diagnostic des

infections sanguines. Avec 450 règles, MYCIN réussissait à diagnostiquer à un niveau proche des experts humains et considérablement meilleur que celui des jeunes médecins.

- Au début des années 80 du siècle dernier, l'IA est rentrée dans l'industrie. En effet 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. Beaucoup de grandes entreprises commencèrent alors à s'intéresser à l'IA et à former leurs propres équipes de recherche. Les Etats-Unis et le Japon financèrent de gros projets en IA, et la Grande Bretagne relança son programme de financement.
- Au milieu des années 80 du siècle dernier, fut le retour des réseaux de neurones. Effectivement quatre groupes de chercheurs découvrirent indépendamment la règle d'apprentissage « back-propagation » qui permit le développement de réseaux de neurones capables d'apprendre des fonctions très complexes. Depuis, l'apprentissage automatique est devenu l'un des domaines les plus actifs de l'IA, et a été appliqué avec succès à de nombreux problèmes pratiques (comme par exemple la fouille de données).
- À partir de 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 ou des études expérimentales plutôt que sur l'intuition, et sont appliquées plus souvent aux problèmes issus du monde réel.

1.2.3 Sous domaine de l'intelligence artificielle(IA) :

L'IA est divisée en de nombreuses sous-disciplines qui essaient chacune d'elle de traiter une partie du problème. Voici les principales sous domaines [1]:

- **Représentation des connaissances et Raisonnement Automatique :**

Comme son nom le suggère, cette branche de l'IA traite le problème de la représentation des connaissances (qui peuvent être incomplètes, incertaines, ou incohérentes) et de la mise en œuvre du raisonnement.

- **Résolution de problèmes généraux :**

L'objectif est de créer des algorithmes généraux pour résoudre des problèmes concrets.

- **Traitement du langage naturel :**

Ce sous-domaine vise à la compréhension, la traduction, ou la production du langage (écrit ou parlé).

- **Vision artificielle :**

Le but de cette discipline est de permettre aux ordinateurs de comprendre les images et la vidéo (par exemple, de reconnaître des visages ou des chiffres).

- **Robotique :**

Cette discipline vise à réaliser des agents physiques qui peuvent agir dans le monde (pour voir les robots humanoïdes les plus avancés aujourd'hui).

- **Apprentissage automatique :**

Dans cette branche de l'IA, on essaie de concevoir des programmes qui peuvent s'auto-modifier en fonction de leur expérience.

1.2.4 État de l'art de l'intelligence artificielle :

Avant de clore ce chapitre, on peut citer quelques exemples qui illustrent l'état de l'art aujourd'hui [1] :

- **Les Jeux :** En 1997, Deep Blue devient le premier programme à battre un champion du monde d'échecs en titre, ce qui fit sensation. Aujourd'hui, des programmes informatiques peuvent aussi jouer à un niveau expert aux dames, au backgammon, ou encore au bridge. En revanche, le jeu de go s'avère être très résistant, et aucun programme informatique ne dépasse aujourd'hui le niveau d'un joueur de club moyen.
- **Vision artificielle :** Le système ALVINN a conduit une voiture à travers les Etats-Unis pendant plus de 4000 kilomètres (avec un peu d'aide pour les moments difficiles comme les sorties d'autoroutes).

- **planification et Ordonnancement** : Un système de planification et d'ordonnancement a contrôlé sans la moindre intervention humaine une navette spatiale pendant 2 jours, planifiant ses actions, détectant des problèmes, et modifiant ses trajectoires au besoin.
- **Systèmes Experts** : Les programmes de diagnostic médical sont aujourd'hui capables de réaliser des diagnostics tout aussi fiables que les experts humains dans plusieurs spécialités médicales. On a même vu un expert humain rejeter le diagnostic proposé par un système expert sur un cas difficile, avant de reconnaître son erreur après que le programme eût expliqué son jugement.
- **Logistique** : Un programme de planification logistique fût utilisé par l'armée américaine pour coordonner ses véhicules, son équipement et ses soldats pendant la guerre du golfe en 1991. Grace à ce système, des problèmes qui dans le passé auraient pris des semaines pour être résolus le furent en quelques heures. On estime que l'économie réalisée par l'armée américaine à l'aide de ce programme fut supérieure au montant qu'elle avait investi dans la recherche en IA pendant 30 ans.
- **Robotique** : Aujourd'hui, des robots sont régulièrement utilisés pour réaliser plusieurs types d'interventions chirurgicales, aidant les médecins à effectuer des manipulations plus précises et ouvrant la possibilité de faire de la chirurgie à distance.

1.3 Machine Learning :

Est un sous domaines de l'intelligence artificielle. Ce qui veut dire donner à une machine la capacité d'apprendre sans la programmer de façon explicite. Les scientifiques font souvent référence à la technologie, afin d'utiliser les données, pour mettre en œuvre le Machine Learning comme des algorithmes. Les algorithmes sont des séries d'instructions à suivre dans un certain ordre pour effectuer une tâche de manière automatique, c'est-à-dire sans avoir besoin de comprendre ou de réfléchir [2].

1.3.1 Type d'apprentissage :

Le machine learning possède une famille composée de trois types d'algorithmes d'apprentissages, notamment les algorithmes d'apprentissages Supervisés, les algorithmes d'apprentissages non supervisé et les algorithmes d'apprentissages par renforcement [2].

- **La méthode supervisé** : c'est-à-dire ici on a des étiquettes à fournir à chaque donnée, par exemple on va montrer à une machine ou un ordinateur plusieurs images d'un chat et on va lui dire que sur cette image il y'a un chat l'image est labélisé par un chat
- **La méthode non supervisé** : c'est-à-dire on ne va pas aider la machine à apprendre. La méthode non supervisé permet de faire ce qu'on appelle du clustering, qui veut dire rassembler plusieurs objets qui ont les mêmes caractéristiques sous une même catégorie, ce qui permet donc de catégoriser des objets sans forcément leur mettre une étiquette dessus.
- **L'apprentissage par renforcement** : c'est-à-dire on teste plusieurs combinaisons d'actions, et plusieurs solutions. De ces tests, on déduit les solutions qui nous permettent d'obtenir les résultats souhaités.

1.3.2 Le fonctionnement du machine learning :

Il faut introduire les données pour que les machines puissent maîtriser le « machine learning », en anglais on appelle les données « Data » (Data c'est le pluriel du mot latin qui s'appelle datum), les Data c'est un ensemble de faits qui vont permettre d'arriver à une conclusion, il existe plusieurs types de données tel que :

➤ **Les données structurés** : ce sont des informations (un mot, un signe, un chiffre) représentés dans des cases. Ex : base de données, tableur (Excel)

➤ **Les données non structurés** : comme son nom l'indique sont des données non organisés, comme par exemple des images, des vidéos, et audio

1.4 Faut-il avoir peur de l'intelligence artificielle ?

L'IA n'éliminera donc pas l'humanité de sa propre initiative. Mais comme toute technologie puissante, l'IA peut être utilisée pour le bénéfice de l'humanité entière ou pour le bénéfice d'un petit nombre aux dépens du plus grand nombre. L'émergence de l'IA va sans doute déplacer des métiers. Mais elle va aussi sauver des vies (par la sécurité routière et la médecine). Elle va très probablement s'accompagner d'une croissance de la production de richesses par habitant.

La question pour les instances dirigeantes est comment distribuer ces nouvelles richesses ? Et comment former les travailleurs déplacés aux nouveaux métiers créés par le progrès technologique ? C'est une question politique et non technologique. C'est une question qui n'est pas nouvelle : l'effet du progrès technologique sur le marché du travail

Existe depuis la révolution industrielle. L'émergence de l'IA n'est qu'un symptôme de l'accélération du progrès technologique [3].

1.5 Les réseaux de neurones :

1.5.1 MODELE BIOLOGIQUE :

Beaucoup d'idées en ingénierie se sont passés, comme par exemple la réalisation d'une machine qui remplacera l'homme. Comment réaliser une telle machine ? Les ingénieurs se sont inspirés de la vraie vie, en basant leurs études sur l'apprentissage chez les êtres vivants. Ils s'intéressent sur la façon dont le cerveau fonctionne : Comment intégrer les informations ? Comment contrôler la motricité ? Comment assurer les fonctions cognitives ?

L'élément central du cerveau est le neurone qui transmet l'information sous forme d'impulsions électrique, chaque neurone est composé d'un corps cellulaire appelé soma, contenant le noyau, et de deux types de prolongements qui assurent la connexion entre les neurones :

- Les dendrites qui recueillent l'information et l'acheminent vers le corps de la cellule.
- L'axone est un élément très long et unique qui conduit l'information du corps cellulaire vers d'autres neurones, avec qui, il fait des connexions appelées synapses.

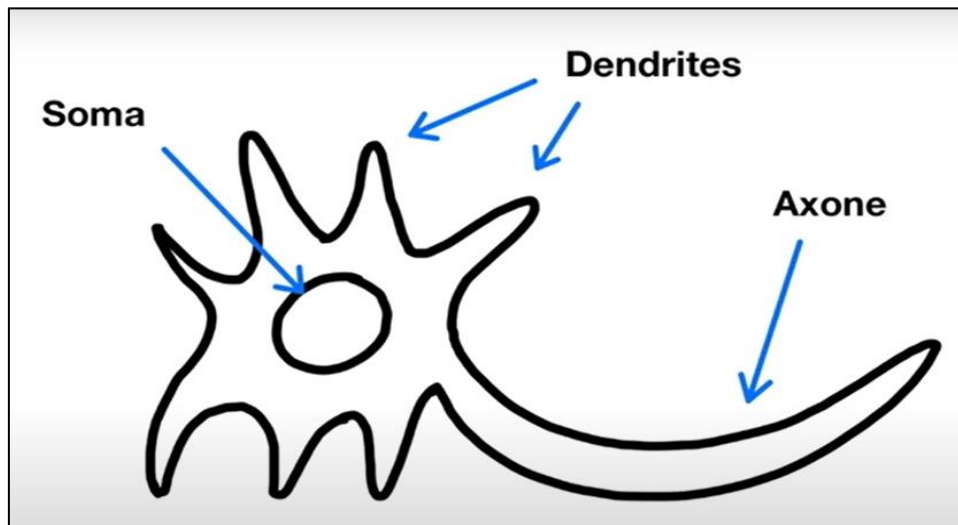


Figure 1. 2: Schéma d'un neurone

Comme le cerveau est extrêmement complexe, les ingénieurs ont donc programmé une abstraction de son fonctionnement, qu'on appelle « **un réseau de neurones** ». Le but des neurones est juste une construction mathématique qui s'inspire du modèle de notre cerveau. Ce neurone appartient à des familles différentes. La famille des entrées « **input** », la famille des couches cachés « **hidden layers** » et la famille des sorties « **output** »

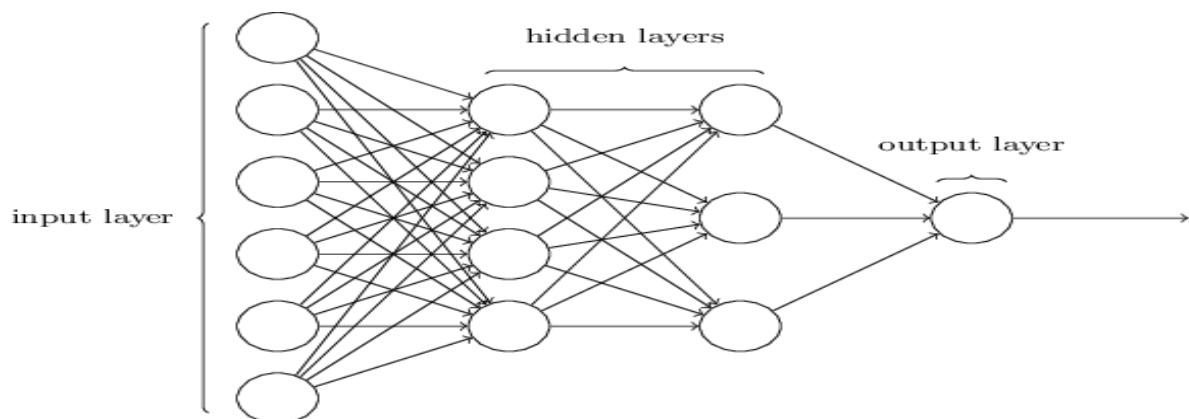


Figure 1. 3 : Architecture d'un réseau de neurone

- **La famille des entrées** : les neurones d'entrée fournissent des informations du monde extérieur au réseau et construisent ensemble une « couche d'entrée ». Aucun calcul n'est effectué dans les nœuds d'entrée. Ils transmettent simplement les informations aux nœuds cachés [4].

- **La famille des couches cachés** : les neurones cachés n'ont pas de connexion directe avec le monde extérieur (d'où le nom "caché"). Ils effectuent des calculs et transfèrent les informations des neurones d'entrée aux neurones de sortie. Un ensemble de neurones cachés forme une « couche cachée » [4].
- **La famille des sorties** : les neurones de sortie construisent ensemble une « couche de sortie ». Ils sont responsables des calculs et du transfert des informations du réseau vers le monde extérieur [4].

1.5.2 Principe de base :

Un neurone artificiel reçoit des entrées de certains autres neurones ou d'une source externe ayant des valeurs numériques : « x_1, x_2, \dots, x_n » auquel il est connecté par des synapses pour calculer en suite la sortie « y ». Chaque entrée « x_i » a un poids associé « w_i », qui est attribué en fonction de son importance relative par rapport aux autres entrées. La valeur d'entrée « x » du neurone correspond à la somme pondérée de ses entrées en ajoutant une autre entrée ayant un poids « b », appelé « **biais** ». Ensuite, le neurone applique une fonction « f » sur cette somme [4].

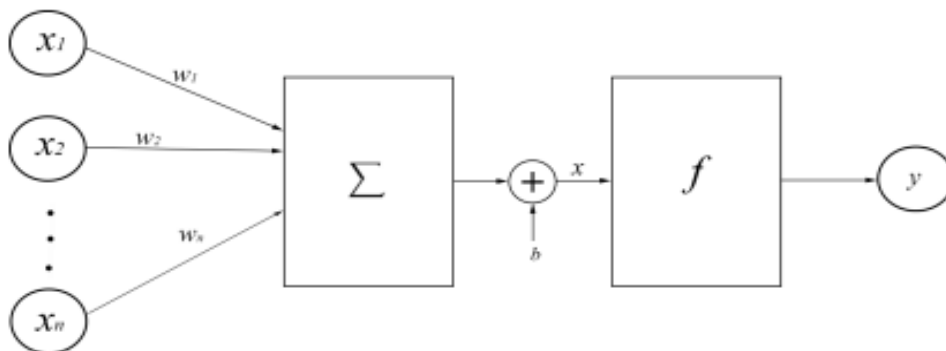


Figure 1. 4: Modélisation d'un neurone artificiel

La sortie du neurone « y », appelée activation de sortie, est calculée selon la formule suivante :

$$y = f \left(\sum_{i=1}^n w_i x_i + b \right) \quad (1.1)$$

Où « f », est la fonction d'activation.

- **La fonction d'activation :**

La fonction d'activation, est une fonction non linéaire. Elle a pour but d'introduire la non-linéarité à la sortie d'un neurone. Ceci est important car la plupart des données du monde réel sont non linéaires, et donc les neurones apprendront ces représentations non-linéaires [5]. Elle devrait être aussi différentiable pour pouvoir appliquer ensuite l'algorithme de rétro-propagation du gradient. Chaque fonction d'activation prend un nombre unique et effectue une certaine opération mathématique sur ce nombre.

Il existe plusieurs fonctions d'activation utilisées dans la pratique :

- ✓ **La fonction logistique :** prend une entrée réelle, ensuite elle la réduit entre 0 et 1

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}} \quad (1.2)$$

- ✓ **La fonction Tangente hyperbolique :** prend une entrée de valeur réelle, ensuite elle la réduit à une valeur dans l'intervalle : $[-1, 1]$

$$f(x) = \tanh(x) = 2 \sigma(2x) - 1 \quad (1.3)$$

- ✓ **La fonction valeur absolue ($|x|$) :** est utilisée lorsqu'on souhaite modéliser une invariance par changement de signe (comme par exemple la reconnaissance d'objets, pour approcher une invariance en fonction des conditions d'illumination).
- ✓ **La fonction ReLU :** la fonction $\text{ReLU}(x) = \max(x, 0)$ est aujourd'hui considérée comme la fonction d'activation standard, à la fois dans l'industrie et dans la recherche académique, mais de très nombreux autres choix sont possibles et

fonctionnent bien selon les Applications. L'idée étant de garder des fonctions suffisamment simples pour pouvoir optimiser le réseau rapidement, cette fonction est une opération élémentaire (appliquée par élément) et remplace toutes les valeurs négatives par zéro [5].

1.5.3 Les types de réseaux neuronaux :

- **Réseaux de neurones convolutifs CNN :**

Couche de convolution : Un réseau de neurones convolutifs est un réseau de neurones qui utilise une opération mathématique, appelée convolution ou produit de convolution. Il s'agit d'une opération linéaire. Chaque réseau de neurones convolutifs contient au moins une couche de convolution. Soient f et g deux fonctions définies sur \mathbf{R} , le produit de convolution entre f et g est généralement noté $(f * g)$; il est défini par l'équation suivante :

$$s(x) = (f * g)(x) = \int_{-\infty}^{+\infty} f(t)g(x - t) dt, \quad (1.4)$$

- **Réseaux de neurones récurrents(RNN) :**

Les réseaux récurrents (ou RNN pour Récurrent Neural Network) sont des réseaux de neurones dans lesquels l'information peut se propager dans les deux sens, y compris des couches profondes aux premières couches. En cela, ils sont plus proches du vrai fonctionnement du système nerveux, qui n'est pas à sens unique. Ces réseaux possèdent des connexions récurrentes au sens où elles conservent des informations en mémoire : ils peuvent prendre en compte à un instant « t » un certain nombre d'états passés. Pour cette raison, les RNN 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 [6].

Un réseau récurrent est principalement utilisé dans la reconnaissance vocale et le traitement automatique du langage naturel (TAL, NLP, TNL). Les RNN sont conçus de manière à reconnaître les caractéristiques séquentielles et les modèles d'utilisation des données requis pour prédire le scénario suivant le plus probable.

- **Les réseaux de neurones à propagation directe (feed-forwarded) :**

Le réseau de neurones à propagation directe (ou feedforward neural network) était le premier et le plus simple réseau de neurones artificiel mis au point. Il contient plusieurs neurones disposés en couches.

Les neurones des couches adjacentes ont des connexions entre eux. Toutes ces connexions ont des poids qui leur sont associés.

Le perceptron est un algorithme d'apprentissage supervisé de classifieurs binaires (séparant en deux classes) et linéaire (classe séparée par une droite), il est constitué d'un seul neurone ou bien d'un seul bloc de construction d'un réseau de neurones.

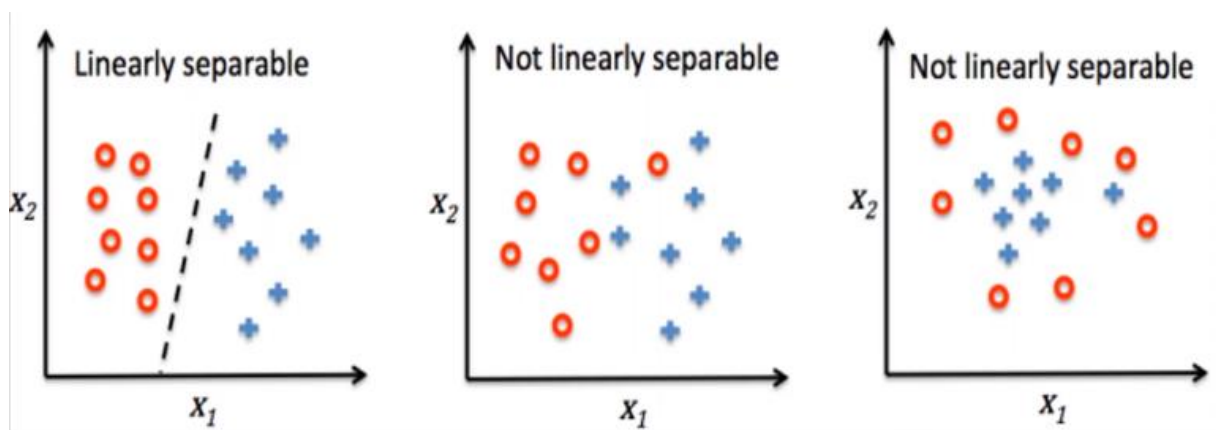


Figure 1. 5: Les classifieurs

L'algorithme du perceptron permet de classer les ensembles de données comme celui illustré à gauche de la figure 1.5, lorsque les données sont séparées de manière linéaire, ils ont tendance à être moins performant pour les ensembles de données comme les deux présenté à droite de la figure 1.5. Cet exemple montre la limitation inhérente à la modélisation d'un perceptron, qui nécessite l'utilisation d'architectures neurales plus complexes.

L'algorithme du perceptron n'est pas garanti d'être convergent dans le cas où les données ne sont pas séparables de manière linéaire. Le perceptron peut parfois aboutir à une

très mauvaise solution avec des données qui ne sont pas séparables linéairement (en comparaison avec de nombreux autres algorithmes d'apprentissage).

Dans la famille des réseaux à propagation directe, on distingue :

a- Les réseaux monocouche (**perceptron** simple)

b- Les réseaux multicouches (**perceptron** multicouches)

- **perceptron monocouche** : Il s'agit d'un réseau de neurones à propagation directe. Il est simple, car il contient qu'une seule couche cachée [4].

- **perceptron multicouches** : Il s'agit donc d'un réseau à propagation directe. Il comporte une ou plusieurs couches cachées. Il est plus utile que le perceptron monocouche dans les applications actuelles [4].

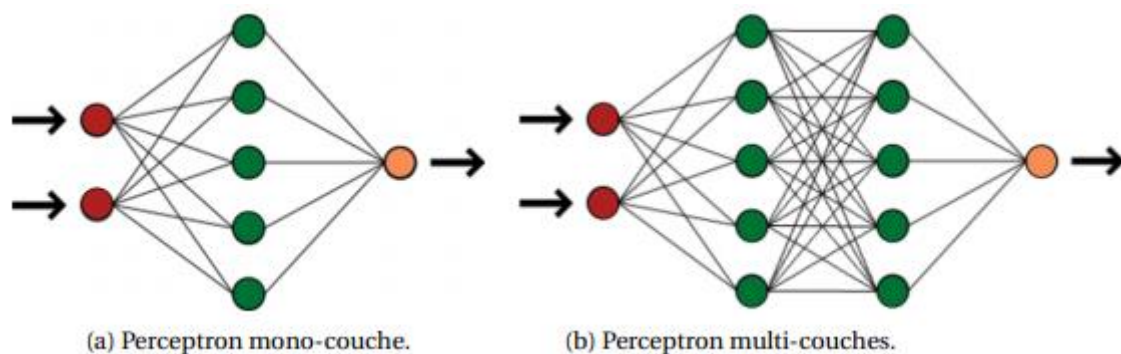


Figure 1. 6: Deux types de perceptrons

1.5.3.1.1 Exemple de réseau à deux couches :

Commençons par un exemple très simple de réseau feedforward à deux couches. Ce réseau à une unique couche intermédiaire [5], on va supposer typiquement que ces valeurs s'écrivent :

$$a_i^1 = g \left(W_{i1}^{[1]} x_1 W_{i2}^{[1]} x_2 \right), i = 1, 2 \quad (1.5)$$

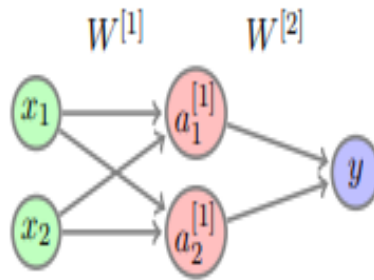


Figure 1. 7 : Exemple de réseau à deux couches

Avec g une fonction non linéaire, par exemple la fonction **RectifiedLinear Unit** ou **ReLU**

$$ReLU(x) = \text{Max}(0, x)$$

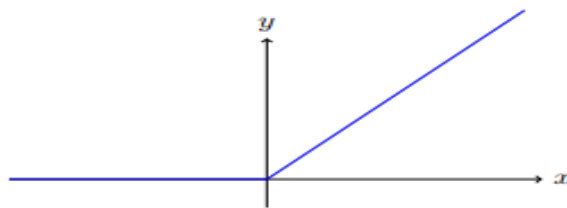


Figure 1. 8: La fonction ReLU

On suppose que la sortie y est calculée à partir de la couche intermédiaire comme

$$y = W_1^{[2]} a_1^{[1]} + W_2^{[2]} a_2^{[1]} \quad , \quad (1.6)$$

La fonction complète représentée par ce réseau est donc la suivante :

$$y'(x) = \sum_{i=1}^2 w_i^{[2]} g\left(\sum_{k=1}^2 w_{ik}^{[2]} x_k\right) = W^{[2]} \text{max}(0, W^{[1]}x) \quad , \quad (1.7)$$

Elle dépend des paramètres :

$$W^{[1]} = \begin{pmatrix} W_{11}^{[1]} & W_{12}^{[1]} \\ W_{21}^{[1]} & W_{22}^{[1]} \end{pmatrix} \quad , \quad (1.8)$$

$$W^{[2]} = (W_1^{[2]} W_2^{[2]}) , \quad (1.9)$$

Le numéro entre crochets dans la notation $W^{[1]}$ désigne la couche sur laquelle on se trouve.

1.5.4 Apprentissage d'un réseau de neurones :

La phase d'apprentissage consiste à ajuster les poids d'un réseau de neurones afin d'obtenir les meilleurs résultats de régression ou de classification. L'objectif de cette optimisation est de minimiser la fonction de coût « ψ » qui calcule l'erreur entre les données de la base d'apprentissage « y » et les données prédites par le réseau de neurones « y' », qui sert de critère pour déterminer la meilleure solution à un problème d'optimisation.

$$\psi(p) = \frac{1}{N} \sum_{i=1}^N (y_i - y'_i)^2 , \quad (1.10)$$

L'un des algorithmes d'optimisation proposés est l'algorithme de rétropropagation qui se base sur la descente de gradient. Il est efficacement utilisée dans des réseaux de neurones multicouches comme les perceptrons multicouches. L'algorithme du gradient a pour but de converger de manière itérative vers une configuration optimisée des poids synaptiques.

1.5.5 Applications :

- statistiques : analyse de données, prévision, classification ;
- robotique : contrôle et guidage de robots ou de véhicules autonomes ;
- imagerie et reconnaissance de formes ;
- traitement du signal ;
- simulation de l'apprentissage.

1.6 Deep learning :

1.6.1 Définition :

L'apprentissage profond (ou deep learning en anglais), est une classe de méthodes dont les principes sont connus depuis la fin des années 80, mais dont l'utilisation ne s'est vraiment généralisée que depuis 2012, environ.

Le Deep Learning, est un terme qui apparait partout, qui fait partie de l'intelligence artificielle et de l'apprentissage automatique (machine learning). L'apprentissage profond est un ensemble de méthodes d'apprentissage qui tentent de modéliser des données avec des architectures complexes combinant différentes transformations non linéaires. Les briques élémentaires de l'apprentissage profond sont les réseaux neuronaux profonds.

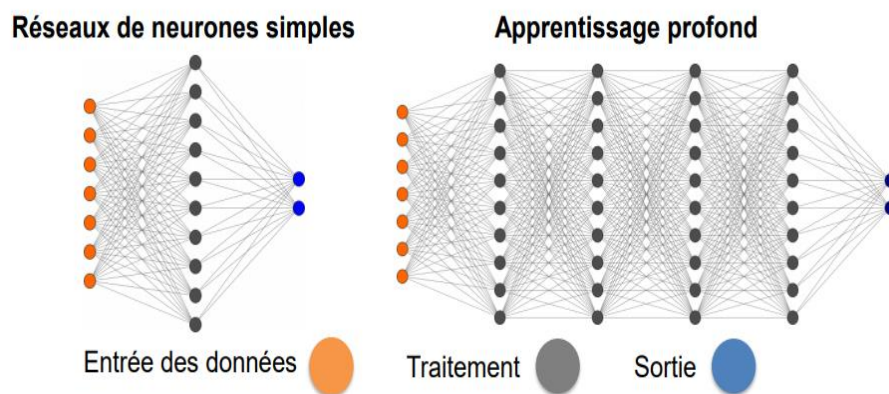


Figure 1.9 : La différence entre réseau de neurone simple et profond

1.6.2 Pourquoi deep learning et pourquoi nous étudions cela maintenant ?

La plupart des entreprises du monde numérique, font de plus en plus appel au « Deep Learning » qui booste l'intelligence artificielle (IA). Cet outil d'apprentissage basé sur des

« Réseaux de neurones artificiels », est utilisé par Google Now, pour la reconnaissance vocale, ou Cortana pour reconnaître des visages. Aussi pour les véhicules autonomes, les systèmes de recommandations d'achats sur internet, BigData qui est devenues beaucoup plus omniprésentes et boîtes à outils open-source comme le flux de tenseurs.

1.6.3 Les applications du Deep Learning :

- **La reconnaissance faciale :**

Dans la reconnaissance faciale, les yeux, le nez, et la bouche, tout autant de caractéristiques qu'un algorithme de Deep Learning va apprendre à détecter sur une photo. Il va s'agir en premier lieu de donner un certain nombre d'images à l'algorithme, puis à force d'entraînement, l'algorithme va être en mesure de détecter un visage sur une image.

- **Le traitement automatique de langage naturel :**

Le traitement automatique de langage naturel est une autre application du Deep Learning. Son but étant d'extraire le sens des mots, voire des phrases pour faire de l'analyse de sentiments. L'algorithme va par exemple comprendre ce qui est dit dans un avis Google, où il va communiquer avec des personnes via des chatbots. La lecture et l'analyse automatique de textes est aussi un des champs d'application du Deep Learning avec le Topic Modeling : tel texte aborde tel sujet.

- **Voitures autonomes :**

Les voitures autonomes, ayant des services d'aide à la conduite, doivent apprendre à un ordinateur à maîtriser certaines parties essentielles de la conduite à l'aide de systèmes de capteurs numériques. Pour ce faire, les entreprises qui construisent de tels types de services commencent généralement par entraîner des algorithmes utilisant une grande quantité de données. Ces nouveaux services pourraient fournir des modèles commerciaux inattendus aux entreprises.

- **Recherche vocale et assistants à commande vocale :**

L'un des domaines d'utilisation les plus populaires de Deep Learning est la recherche vocale ainsi que les assistants intelligents à commande vocale. Les géants de la technologie

ont déjà fait d'importants investissements dans ce domaine. Des assistants à commande vocale peuvent être trouvés sur presque tous les smartphones. Le « Siri » d'Apple est sur le marché depuis octobre 2011. « Google voice » aujourd'hui, est l'assistant à

commande vocale pour Androïde, il a été lancé moins d'une année après « Siri ». Le plus récent des assistants intelligents à commande vocale est Microsoft Cortana.

- **Ajout automatique de sons à des films muets :**

Dans cette tâche, le système doit synthétiser des sons pour correspondre à une vidéo silencieuse. Le système est formé à l'aide de 1 000 exemples de vidéos avec le son d'une baguette frappant différentes surfaces et créant différents sons. Pour associer les images vidéo à une base de données de sons préenregistrés, le modèle Deep learning est le mieux indiqué, afin de sélectionner le son à reproduire qui correspond le mieux à ce qui se passe

Dans la scène. Le système a ensuite été évalué à l'aide d'un test de contrôle, comme une configuration dans laquelle les humains devaient déterminer quelle vidéo comportait le son réel ou le son factice (synthétisé). Ceci utilise à la fois les réseaux de neurones convolutionnels et les réseaux de neurones récurrents à mémoire à court terme.

- **Traduction automatique :**

Il s'agit d'une tâche dans laquelle des mots, expressions ou phrases donnés dans une langue sont automatiquement traduits dans une autre langue. La traduction automatique existe depuis longtemps, mais Deep Learning permet d'obtenir les meilleurs résultats dans deux domaines spécifiques :

- Traduction automatique de texte.
- Traduction automatique d'images.

La traduction de texte peut être effectuée sans aucun traitement préalable de la séquence, ce qui permet à l'algorithme d'apprendre les dépendances entre les mots et leur correspondance avec une nouvelle langue.

- **Génération automatique de texte :**

C'est une tâche intéressante, où un corpus de texte est appris et à partir de ce modèle, un nouveau texte est généré, mot par mot ou caractère par caractère. Le modèle est capable d'apprendre comment épeler, ponctuer, former des phrases et même capturer le style du texte dans le corpus. Les grands réseaux de neurones récurrents sont utilisés pour apprendre

la relation entre les éléments dans les séquences de chaînes d'entrée, puis pour générer du texte.

- **Reconnaissance d'image :**

Un autre domaine populaire en matière de Deep Learning est la reconnaissance d'image. Son objectif est de reconnaître et d'identifier les personnes et les objets dans les images, ainsi que de comprendre le contenu et le contexte. La reconnaissance d'image est déjà utilisée dans plusieurs secteurs tels que les jeux, les médias sociaux, la vente au détail, le tourisme, etc.

Cette tâche nécessite la classification des objets d'une photo parmi un ensemble d'objets connus auparavant. Une variante plus complexe de cette tâche, appelée détection d'objet, consiste à identifier spécifiquement un ou plusieurs objets dans la scène de la photo et à dessiner un cadre autour d'eux.

- **La description automatique d'image :**

Le sous-titrage automatique des images est la tâche pour laquelle le système doit générer une légende décrivant le contenu de l'image. Une fois que vous pouvez détecter des objets sur des photographies et générer des étiquettes pour ces objets, vous pouvez voir que l'étape suivante consiste à transformer ces étiquettes en description de phrase cohérente. Généralement, les systèmes impliquent l'utilisation de très grands réseaux de neurones convolutifs pour la détection d'objets sur les photographies, puis d'un réseau de neurones récurrent comme une mémoire à court terme à long terme pour transformer les étiquettes en une phrase cohérente.

- **Colorisation automatique :**

La colorisation de l'image pose le problème de l'ajout de couleurs aux photographies noir et blanc. Le deep learning peut être utilisé pour utiliser les objets et leur contexte dans la photographie pour colorer l'image, un peu comme un opérateur humain pourrait aborder le problème. Cette capacité tire parti des réseaux de neurones de convolution de grande qualité et de très grande taille formés pour ImageNet et cooptés pour le problème de la colorisation

de l'image. Généralement, l'approche implique l'utilisation de très grands réseaux de neurones convolutifs et de couches supervisées qui recréent l'image avec l'ajout de couleurs.

- **La détection du cancer du cerveau :**

Une équipe de chercheurs français a noté qu'il était difficile de détecter les cellules cancéreuses du cerveau invasives au cours d'une intervention chirurgicale, en partie à cause des effets de l'éclairage dans les salles d'opération. Ils ont découvert que l'utilisation de réseaux de neurones conjointement avec la spectroscopie Raman pendant les opérations leur permettait de détecter les cellules cancéreuses plus facilement et de réduire le cancer résiduel après l'opération.

- **Analyse des sentiments du texte :**

De nombreuses applications ont des commentaires ou des systèmes de révision basés sur des commentaires intégrés à leurs applications. La recherche sur le traitement du langage naturel et les réseaux de neurones récurrents ont parcouru un long chemin et il est maintenant tout à fait possible de déployer ces modèles sur le texte de votre application pour extraire des informations de niveau supérieur. Cela peut être très utile pour évaluer la polarité sentimentale dans les sections de commentaires ou pour extraire des sujets significatifs à l'aide de modèles de reconnaissance d'entités nommées.

- **Recherche en marketing :**

En plus de rechercher de nouvelles fonctionnalités susceptibles d'améliorer votre application, Deep Learning peut également être utile en arrière-plan. La segmentation du marché, l'analyse des campagnes marketing et bien d'autres peuvent être améliorés à l'aide

de modèles de régression et de classification Deep Learning. Cela vous aidera vraiment beaucoup si vous avez une grande quantité de données. Sinon, vous ferez probablement mieux d'utiliser des algorithmes traditionnels d'apprentissage automatique pour ces tâches plutôt que Deep learning.

1.6.4 Les applications de Deep learning en télécom :

- Sélection automatique des fonctionnalités et prédiction du taux de désabonnement.

- Une étude comparative de réduction de la dimensionnalité dans la segmentation des clients télécoms.
- Effectuer des prédictions très précises via des réseaux convolutifs pour les défis de télécommunication réels.
- Utiliser l'apprentissage en profondeur pour prédire le taux de désabonnement des clients dans un réseau de télécommunications mobiles.
- Apprentissage en profondeur appliqué aux données de téléphonie mobile pour la classification des revenus individuels.
- Prédiction du sexe des abonnés dans les télécommunications.
- L'affectation de ressources radio pour les réseaux ultra denses 5G.
- Prévision du trafic dans les systèmes de télécommunications.
- Détection d'anomalies assistée par Big Data basée sur l'apprentissage en profondeur dans les réseaux cellulaires.
- Les données de télédétection : un tutoriel technique sur l'état de l'art.

1.7 Conclusion

Nous pensons que l'apprentissage profond aura beaucoup plus de succès dans un proche avenir parce qu'il nécessite très peu d'ingénierie à la main, de sorte qu'il peut facilement tirer parti de l'augmentation de la quantité de calcul et de données disponibles. De nouveaux algorithmes et architectures d'apprentissage actuellement en cours de développement pour les réseaux neuronaux profonds ne feront qu'accélérer ce progrès.

Chapitre 2

System OFDM

et PAPR

2.1 Introduction :

Dans ce chapitre, nous nous intéressons à une technique de modulation multi-porteuse particulière, qui est le multiplexage par division en fréquences orthogonales (OFDM). L'OFDM est une technologie ancienne au succès récent qui consiste à transmettre des données en parallèle sur plusieurs fréquences orthogonales, son avantage concernant principalement la robustesse du signal vis-à-vis du canal multi-trajet avec évanouissement et l'encombrement spectral optimal. Cependant, cette technique représente le désavantage d'avoir de fortes fluctuations en amplitude de l'enveloppe du signal modulé et donc par des variations importantes en puissance instantanée.

Le «Peak-to-Average Power Ratio» (PAPR) prend en compte ces variations en puissances. Il fait entrer le signal transmis dans la région non linéaire de l'amplificateur à haute puissance (HPA), ce qui génère une distorsion du signal. Pour réduire le PAPR, il existe plusieurs techniques, certaines techniques agissent sur l'amplificateur afin d'éviter la saturation du signal d'entrée et d'autres techniques se basent plutôt sur un traitement réalisé directement au niveau du signal [8].

2.2 La modulation OFDM :

L'OFDM (Orthogonal Frequency Division Multiplexing) est une technique qui divise une large bande de fréquence en plusieurs sous-bandes ou sous-porteuses. L'orthogonalité qui la caractérise, autorise un certain recouvrement spectral entre les sous-porteuses, ce qui augmente l'efficacité spectrale du système. La figure 2.1 ci-dessus résume les différentes étapes d'une transmission OFDM par le schéma bloc, où **CP** est l'indicateur de préfixe cyclique, **S/P** et **P/S** qui représentent les conversions série-parallèle et parallèle-série, respectivement [7].

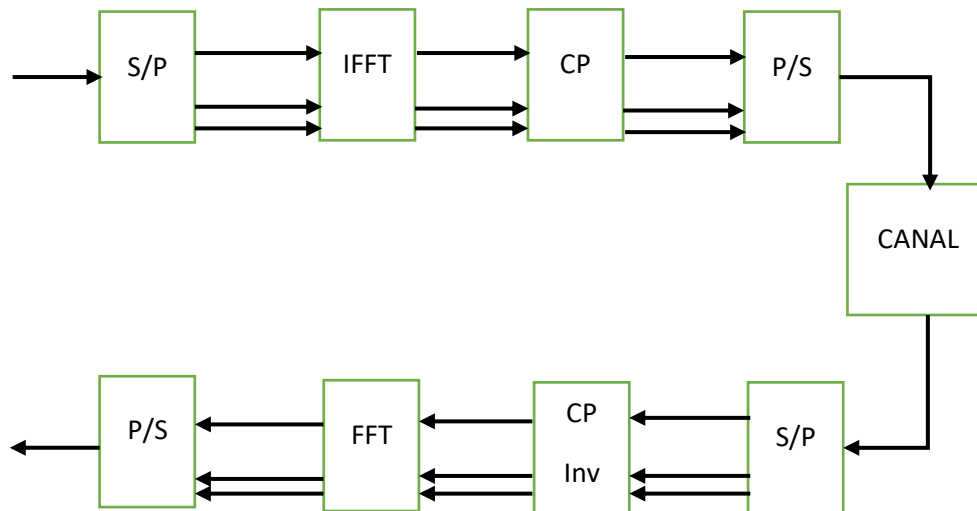


Figure 2.1 : Les blocs d'un système OFDM

2.3 Principe de l'OFDM :

Le principe de la modulation OFDM est de paralléliser la transmission des symboles afin d'utiliser le plus efficacement possible la largeur de bande disponible. La bande totale disponible est divisée en N sous-porteuses (ou sous-canaux) uniformes et orthogonales. Les symboles de la constellation sont insérés dans chaque sous-porteuse.

À la suite de l'interférence entre symboles (IES) et l'interférence entre porteuses (IEP), résultant des trajets multiples du canal de transmission, La condition d'orthogonalité n'est plus maintenue à l'entrée du récepteur. L'implémentation d'un intervalle de garde entre chaque symbole OFDM transmis, résout ce problème avec une petite perte d'énergie de transmission.

La modulation OFDM est un bon candidat pour les transmissions à haut débit binaire, car l'égalisation est très simple et les coûts en complexité d'implantation sont réduits.

2.3.1 La chaîne de transmission OFDM :

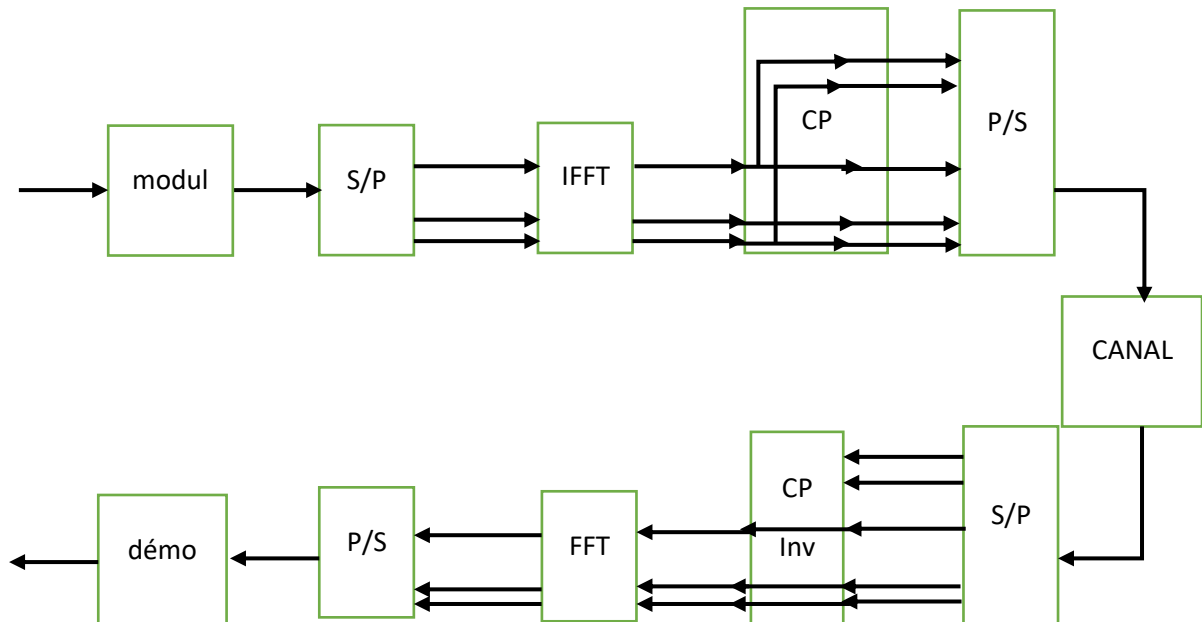


Figure 2.2 : Schéma de fonctionnement de l'OFDM

- Le schéma synoptique de la **figure 2.2** illustre les différents modules qui composent la chaîne de transmission OFDM.
- Le modulateur QAM transforme les données binaires \mathbf{b}_i , de durée T_i , en symboles complexes \mathbf{X}_k de durée $T_q = \log_2 M T_b$, où M est la taille de la constellation de la modulation QAM utilisée.
- L'IFFT est utilisée afin de transformer le spectre du signal OFDM au domaine temporel pour la transmission à travers le canal.
- À la réception, les opérations inverses sont réalisées, commençant par la suppression du préfixe cyclique, la décomposition spectrale des échantillons reçus est calculée en utilisant l'**algorithme FFT**.
- En dernier ressort, la phase de démodulation, permet de retrouver les données binaires transmises.

2.3.2 Principes du modulateur et du démodulateur utilisant les transformées de Fourier discrètes :

2.3.3.1 La transformée de Fourier discrète :

La réalisation d'un modulateur/démodulateur nécessite un circuit complexe, c'est pour cette raison qu'Elbert et Weinstein ont conçu dans les années 70 du siècle dernier, un système parallèle en utilisant la transformée de Fourier discrète. Cette méthode a éliminé la complexité du système OFDM pour les processus de modulation et de démodulation [10].

La transformée de Fourier discrète (DFT) est définie comme suit :

Considérons maintenant un échantillonnage du signal $S(t)$ obtenu dans l'équation ci-dessus :

$$s(t) = e^{2j\pi f_0 t} \sum_{k=0}^{N-1} C_k e^{2j\pi \frac{kt}{T_u}} , \quad (2.7)$$

Avec une période $T_e = T_u/N$. À chaque instant $t = nT_e$ avec $n \in [1:N]$, le signal échantillonné $S(nT_e)$, que l'on notera $S(n)$ pour simplifier s'écrit :

$$S(n) = \sum_{k=0}^{N-1} C_k e^{2j\pi \frac{k}{T_u} n T_e} , \quad (2.8)$$

En tenant compte de la relation liant T_u et T_e , la relation précédente devient:

$$S(n) = \sum_{k=0}^{N-1} C_k e^{2j\pi \frac{k}{N} n} \quad 1 \leq n \leq N , \quad (2.9)$$

Le terme $S(n)$ sera appelé symbole OFDM, l'ensemble de ces symboles, pour n variant de 1 à N et noté $\{S\}_{1:N}$, constituera la trame OFDM.

La transformée inverse de Fourier discrète associe aux symboles X_k pour $k \in [1:N]$ les symboles $x(n)$ avec $n \in [1:N]$ de la façon suivante:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{2j\pi \frac{k}{N} n} \quad 1 \leq n \leq N , \quad (2.10)$$

Au facteur N près, le signal $\mathcal{S}(n)$ de l'expression (2.9) apparaît comme la transformée de Fourier inverse et discrète (TFDI) des symboles \mathcal{C}_k , l'indice k faisant référence à la k -ième sous-porteuse. L'ensemble des symboles complexes \mathcal{S}_n du signal en bande de base sont obtenus à la sortie en introduisant ces symboles associés à la modulation numérique dans un module de TFDi, c'est-à-dire la trame OFDM. À la réception, il suffira d'effectuer la transformée de Fourier discrète directe (TFD) sur les symboles reçus pour restituer le message désiré.

N : représente la taille de la **DFT/IDFT**.

La matrice de la **DFT** est composée d'exponentielles complexes. Elle est définie comme suit :

$$\mathbf{F} = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \dots & \mathbf{1} \\ \mathbf{1} & e^{-j2\pi/N} & e^{-j2\pi2/N} & \dots & e^{-j2\pi(N-1)/N} \\ \mathbf{1} & e^{-j2\pi2/N} & e^{-j2\pi4/N} & \dots & e^{-j2\pi2(N-1)/N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{1} & e^{-j2\pi(N-1)/N} & e^{-j2\pi2(N-1)/N} & \dots & e^{-j2\pi(N-1)(N-1)/N} \end{pmatrix}, \quad (2.3)$$

Cependant, une autre forme de la DFT est utilisée pour les systèmes OFDM, appelée la transformée de Fourier rapide (FFT), qui est un algorithme de DFT développé en 1965. Cette transformée rapide permet de réduire considérablement le temps de calcul [7].

2.3.3 L'orthogonalité :

La différence fondamentale entre les différentes techniques classiques de modulation multi-porteuses et l'OFDM, est que cette dernière autorise un recouvrement spectral entre ces sous-porteuses, ce qui permet d'augmenter sensiblement leur nombre. Les porteuses doivent respecter une contrainte d'orthogonalité, à la fois dans le domaine temporel et fréquentiel pour que le recouvrement n'aura pas d'effet néfaste.

L'utilisation d'un très grand nombre de porteuse nécessite beaucoup de modulateur/démodulateur, des filtres et de largeur de bande.

Pour résoudre ces deux problème, on spécifie un espacement rigoureusement régulier de $\frac{1}{T_u}$ entre les sous porteuses, où « T_u » est la période du symbole OFDM.

La figure ci-dessus présente les porteuses qui forment un ensemble orthogonal.

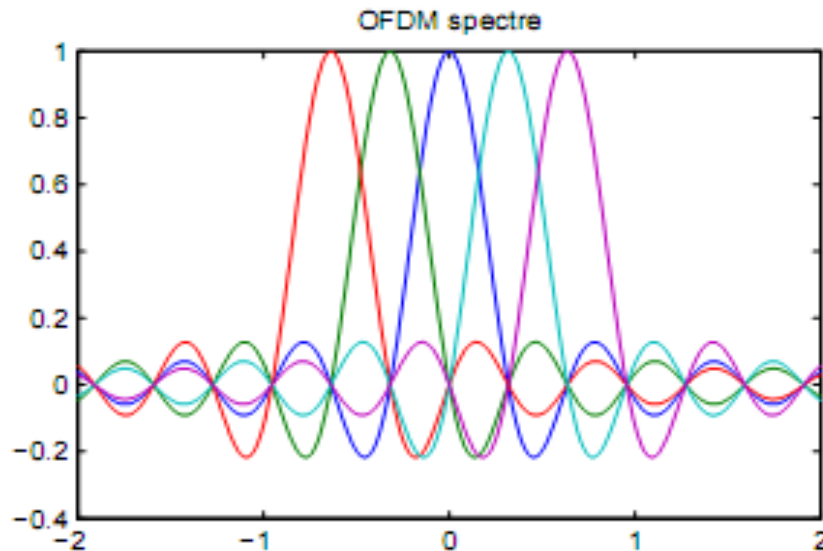


Figure 2.3 : Forme d'un spectre du signal OFDM

Un signal OFDM peut s'écrire sous la forme :

$$x_n(\mathbf{n}) = \sum_{k=0}^{N-1} X_{n,k} e^{-j2\pi f_k t} \quad , \quad (2.4)$$

Où :

$X_{n,k}$: sont les symboles à émettre à l'instant n sur la $k^{\text{ème}}$ sous-porteuse.

$$f_k = f_0 + k\Delta f = f_0 + k \frac{1}{T_s} \quad \text{Avec}$$

- f_0 représente la fréquence de la sous-porteuse originale.
- f_k représente la fréquence de la sous-porteuse d'indice k .
- Δf représente l'écart fréquentiel entre deux sous-porteuses voisines et N indique le nombre de sous-porteuses.

Les fréquences des sous porteuses doivent être toujours des multiples entiers de la fréquence fondamentale pour que la condition d'orthogonalité soit respectée [7]

2.3.4 L'interférence entre symboles (IES) :

Les signaux OFDM sont transmis à des intervalles égaux, ils doivent parcourir un certain trajet pour atteindre le récepteur. Dans le cas d'un canal multi-trajets, un symbole transmis prend différents retards pour arriver au récepteur à cause des différents chemins de propagation, ce qui va engendrer par la suite des étalements temporels. La prolongation de la durée des

symboles fait chevaucher ces derniers entre eux, d'où la naissance du phénomène des interférences entre symboles (IES) [10]. Pour surmonter ce problème l'OFDM utilise un intervalle de garde.

2.3.5 L'interférence entre porteuses (IEP) :

L'orthogonalité dans un système OFDM signifie qu'au maximum de chaque spectre d'une sous-porteuse, tous les spectres des autres sous-porteuses sont nuls [11]. Les spectres des sous-porteuses se recouvrent mais demeurent orthogonaux entre eux. L'interférence entre porteuses (IEP) est causée par la perte d'orthogonalité et donc par la présence des symboles de données d'une sous-porteuse sur les sous-porteuses adjacentes.

2.4 Intervalle de garde :

L'intervalle de garde permet d'éviter l'utilisation d'égalisateurs de canaux et d'éviter la présence des interférences, et ce en l'ajoutant entre deux symboles OFDM transmis successivement. De plus la durée utile d'un symbole sera choisie suffisamment par rapport à l'étalement des échos, ce qui va limiter l'interférence entre symbole. La relation entre la durée de symbole, la durée utile et l'intervalle de garde nous donne la relation suivante :

$$T_{OFDM} = T_u + T_g \quad (2.5)$$

T_u : est la période utile pendant laquelle est émise l'information.

T_g : est le temps de garde.

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).

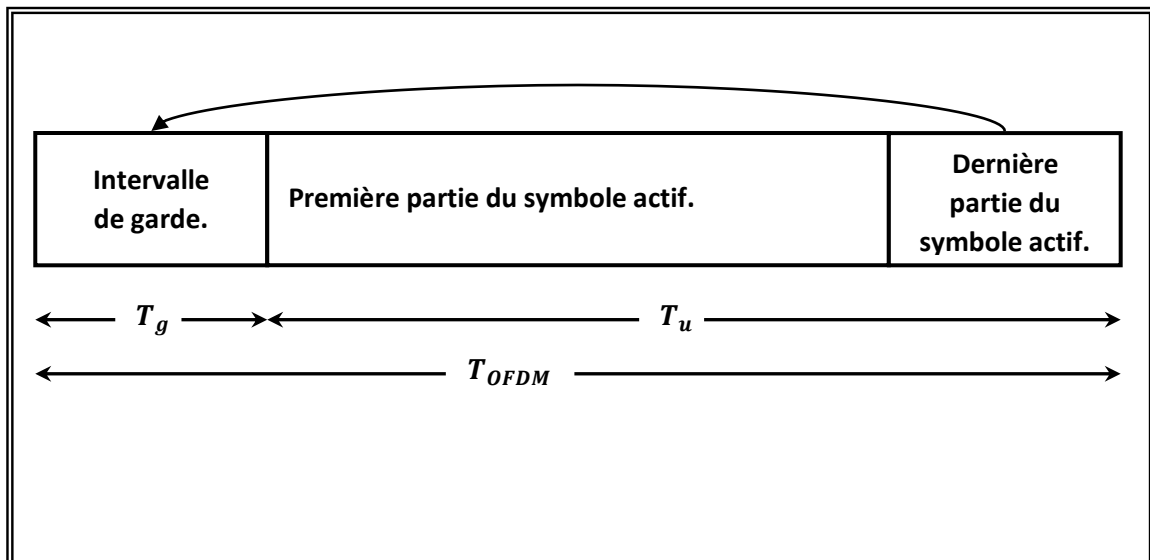


Figure 2.4 : Intervalle de garde pour un système OFDM

Dans cette figure, on illustre l'ajout d'un intervalle de garde :

- L'intervalle de garde est formé, lorsqu'on ajoute un segment au début du symbole qui doit être identique au segment de même longueur à la fin du symbole.
- La période du symbole est prolongée de manière à être supérieure à la période utile.
- 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.
- Les brouillages IEP et IES ne se produisent que lorsque le retard relatif est plus long que l'intervalle de garde.

2.5 les avantages et les inconvénients de l'OFDM :

2.5.1 Les avantages :

- Elle permet une utilisation efficace des ressources fréquentielles contrairement aux autres solutions classiques de multiplexage fréquentiel. Ceci est dû au fait, que dans l'OFDM, les canaux se chevauchent tout en gardant une orthogonalité parfaite.

- Une égalisation numérique et un décodage simple est optimal grâce à l'utilisation de l'intervalle de garde et de l'algorithme de Viterbe.
- Les techniques multi-porteuses sont robustes au bruit impulsif puisque chaque porteuse est affectée d'un bruit indépendant des autres porteuses.

2.5.2 Les inconvénients :

Le système OFDM présente certains désavantages importants tels que :

➤ Le décalage fréquentiel :

Le système OFDM est sensible au décalage fréquentiel. Les auteurs ont démontré que les systèmes multi-porteuses sont beaucoup plus sensibles aux décalages fréquentiels que les systèmes mono-porteuses. Ils ont même donné une relation qui détermine la dégradation en termes de rapport signal à bruit (**SNR**) :

$$D \cong \frac{10}{\ln 10} \left(\frac{1}{3}\right) \left(\pi N \frac{\Delta f}{R}\right)^2 \frac{E_s}{N_0} , \quad (2.5)$$

Δf : Le décalage fréquentiel normalisé par l'écart entre les porteuses $\Delta f = \frac{B}{N}$.

R : Le taux de symboles qui est égal à $N = T_S$.

$\frac{E_s}{N_0}$: Est le rapport entre l'énergie par symbole et la densité spectrale de bruit.

➤ Le PAPR :

Le rapport de puissance élevé de pointe à moyen (PAPR) a été l'un des principaux inconvénients des systèmes de multiplexage orthogonal de division de fréquence (OFDM).

Le PAPR est la variation importante de l'amplitude du signal en fonction des symboles à transmettre. En effet, si l'addition des porteuses se fait de manière cohérente, le PAPR peut être très important. Il est défini comme étant le rapport de la puissance instantanée maximale d'un signal sur la puissance moyenne de ce signal.

$$PAPR(x(t)) = \frac{\max |x(t)|_{0 \leq t \leq T}^2}{E \left[\frac{1}{T} \int_0^T |x(t)|^2 dt \right]} , \quad (2.6)$$

À chaque fois qu'on aura un pic du signal, l'AP doit consommer plus d'énergie pour le transmettre. Or il est nécessaire d'optimiser la consommation d'énergie surtout dans les transmissions sans fil. Pour optimiser l'utilisation des AP, on utilise la réduction du PAPR, qui fait l'objet de notre étude.

2.6 Problème du PAPR lors de l'amplification de puissance :

Dans la modulation mono-porteuse, les données sont modulées par une porteuse occupant toute la bande de fréquence, c'est d'ailleurs pour cette raison que les variations et l'enveloppe du signal modulé ne dépendent que de la constellation, du filtre de mise en forme, par exemple : une modulation numérique 16-QAM génère un signal plus variant qu'une modulation QPSK.

Dans la modulation multi-porteuse, la bande allouée est divisée en plusieurs sous-bandes, et les données sont modulées parallèlement sur plusieurs fréquences porteuses. L'addition de ces porteuses engendre une forte dynamique de l'enveloppe du signal. La puissance instantanée varie considérablement par rapport à la puissance moyenne.

Afin d'avoir suffisamment de puissance pour lutter contre les évanouissements du canal et pour être détecté par le récepteur, Le signal émis doit s'introduire dans un amplificateur de puissance (PA), et il faut qu'il y ait un dispositif dont la caractéristique de la tension de sortie en fonction de la tension d'entrée soit non-linéaire.

2.7 Fonction de répartition du PAPR pour l'OFDM :

Le PAPR est une variable aléatoire qui mesure les variations de l'enveloppe d'un signal multi-porteuses comme l'OFDM. Le caractère aléatoire de cette variable vient du fait que les symboles d'entrées arrivent aléatoirement à l'entrée de la modulation. Chaque symbole est porté par une fonction porteuse, et l'addition de ces porteuses peut être cohérente sur certains instants, engendrant ainsi des pics d'amplitude. L'analyse du PAPR peut être conduite par une étude déterministe qui consiste à calculer sa valeur maximale, ou une borne supérieure, ou bien une étude statistique souvent réalisée, soit à l'aide de la fonction de répartition CDF (Cumulative distribution function), soit à l'aide de sa fonction complémentaire CCDF (Complementary Cumulative distribution function).

2.8 Les techniques de réduction de PAPR :

Plusieurs techniques de réduction du PAPR ont été proposées pour les systèmes OFDM, on peut citer par exemple les techniques à compatibilité descendante, tel que la technique d'écrêtage et d'ajout de signal. La notion de compatibilité descendante signifie que le récepteur n'a aucune information sur le traitement déjà fait au niveau émetteur. On trouve aussi, d'autres techniques de réduction de PAPR pour lesquelles le principe de la compatibilité descendante n'est plus respecté. Parmi ces techniques, il y a la technique de selective mapping qui a l'avantage de garantir une bonne réduction du PAPR pour n'importe quel nombre de sous-porteuses. On trouve aussi, la technique de codage Reed-Muller. Une autre technique de réduction de PAPR, est la technique Tone réservation, qui utilise quelques porteuses pour ajouter un signal réduisant les pics du signal multi-porteuses.

2.9 Critère d'évaluation des techniques de réduction du PAPR :

➤ **Compatibilité descendante :**

C'est lorsque le récepteur n'est pas modifié et n'opère pas de traitement supplémentaire.

➤ **Dégradation du taux d'erreurs binaire (TEB) :**

il existe des techniques de réduction de PAPR qui dégradent le TEB et des techniques qui ne dégradent pas le TEB, donc la dégradation du TEB est un facteur très important pour le choix des techniques. Les techniques qui dégradent le TEB sont des techniques qui modifient la structure du signal, en introduisant des distorsions. Le signal reçu est alors modifié et sa courbe de TEB est dégradée. Pour les techniques qui ne dégradent pas le TEB, on peut citer par exemple la technique TR (Tone Reservation), la technique ACE (Active Constellation Extension) et la technique TI (Tone Injection).

➤ **Diminution du débit utile :**

Pour rendre l'opération de réduction du PAPR transparente pour la démodulation, les méthodes de réduction doivent exiger la transmission d'une information supplémentaire au récepteur. Cette transmission réduit le débit utile.

➤ **Complexité :**

Si l'algorithme de réduction du PAPR est très complexe et lent à s'exécuter, cela limitera son utilisation, même si la réduction du PAPR est considérable.

2.10 La réduction de PAPR :

Pour résoudre le problème du PAPR, plusieurs schémas ont été proposés, comme la technique de réservation de tonalité (TR), dans laquelle un nombre de porteuses inutilisées sont réservées pour rajouter un signal de réduction. Cette technique est capable de réduire le PAPR sans provoquer de distorsion du signal.

2.11 Technique TR :

Comme on l'a déjà défini, c'est une technique de réduction du PAPR sans distorsion supplémentaire du signal. Cette technique réserve (L) porteuse pour générer « le signal de réduction du PAPR », et utilise le reste c'est-à-dire (N-L) porteuse pour la transmission des données.

Le signal original dans le domaine temporel, est représenté comme suit :

$$\hat{x} = x + c = Q(X + C), \quad (2.8)$$

Où :

- Q : est la matrice de transformation IFFT
- $C = [0, C_1, 0, \dots, C_L]^T$: est le vecteur de symbole réservé dans le domaine fréquentiel, c'est le signal d'annulation de crête dans le domaine temporel.

X et C , doivent toujours être orthogonaux l'un par rapport à l'autre pour éviter la distorsion du signal, C'est-à-dire :

$$X_k + C_k = \begin{cases} X_k, & K \in R \\ C_k, & k \in R^c \end{cases}, R \cap R^c = \emptyset,$$

Où :

- R : est l'ensemble des indices des porteuse de données.
- R^c : représente l'ensemble des indices de la PRT.

Le vecteur de symbole réservé optimal est obtenu par :

$$C^* = \arg \min_c \|Q(X + C)\|_\infty$$

S'il y a K valeurs pour chaque tonalité réservée, il y'aura un total de K^L vecteur de symboles réservés candidats pour un signal OFDM.

Le rapport de réservation de tonalité(TRR) peut être défini comme suit :

$$TRR = L/N$$

Dans la technique TR, la réduction du PAPR et la bande passante dépendent du TRR. Plus le TRR est élevé, plus la réduction du PAPR est efficace, mais moins la bande passante est efficace. Donc on doit chercher le meilleur compromis entre la performance du PAPR et l'efficacité de la bande passante.

À partir de la **figure 2.5**, on a $X_k^m C_k^m = 0$ où X_k^m désigne le **k-ième** symbole numérique du **m-ième** symbole OFDM et C_k^m le **k-ième** symbole numérique du **m-ième** signal ajouté.

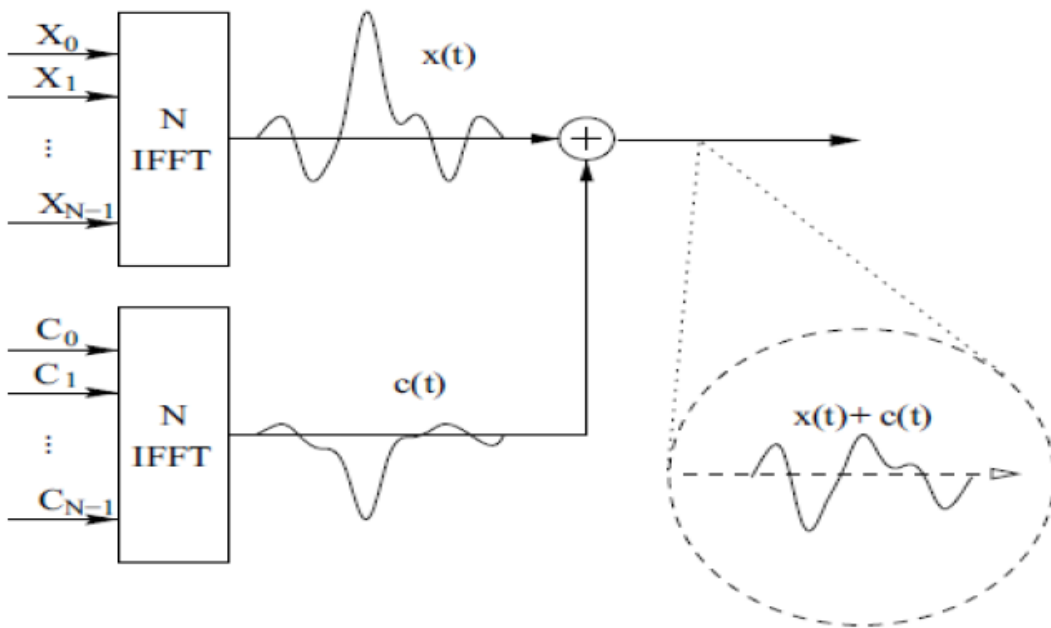


Figure 2.5 : Principe de fonctionnement de TR

Le PAPR du signal est donné par :

$$PAPR(x^m + c^m) = \frac{\|x^m + c^m\|_\infty^2}{E\left(\left|x^m \left[\frac{N}{L}\right]\right|^2\right)}, \quad (2.11)$$

➤ **Avantage :**

La technique TR permet d'obtenir une bonne réduction du PAPR.

➤ **Inconvénients :**

Complexité pour l'optimisation et pour l'implémentation. En plus, elle nécessite beaucoup d'énergie pour l'émission

2.12 Conclusion :

Nous avons vu que la technique de modulation OFDM propose l'utilisation d'un groupe de sous-porteuses pour la transmission des données en parallèle. On a aussi vu que l'OFDM est bien utile dans la transmission sans fil à haut débit binaire, puisqu'il est efficace en largeur de bande en plus d'être simple à implémenter grâce à la transformée de Fourier. Son inconvénient est le PAPR.

Chapitre 3

Réduction du

PAPR avec la

méthode

TR-NET

3.1 Introduction :

Dans nos jours, le Deep Learning, a connu un grand succès dans le monde, donc ce chapitre nous allons voir étape par étape la réduction de PAPR par le Deep Learning en utilisant la méthode TR-NET (Tone Reservation network), ainsi que les caractéristiques de base de FFNN sont également présentées.

3.2 Les modèles de système et FFNN :

3.2.1 Modèle du système :

Si l'on considère un système OFDM avec N sous porteuses, le signal OFDM transmis en temps discret peut être exprimé comme suit [15] :

$$x_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k e^{j2\pi nk/WN} \quad , n = 0, 1, \dots, WN - 1 \quad (3.1)$$

Où :

$X = [X_0, X_1, \dots, X_{n-1}]^T$: est un vecteur complexe

X_k : est une modulation par déplacement de phase (PSK) ou une modulation d'amplitude en quadrature (QAM), transmis par la Kieme sous porteuse.

W : représente le facteur d'échantillonnage.

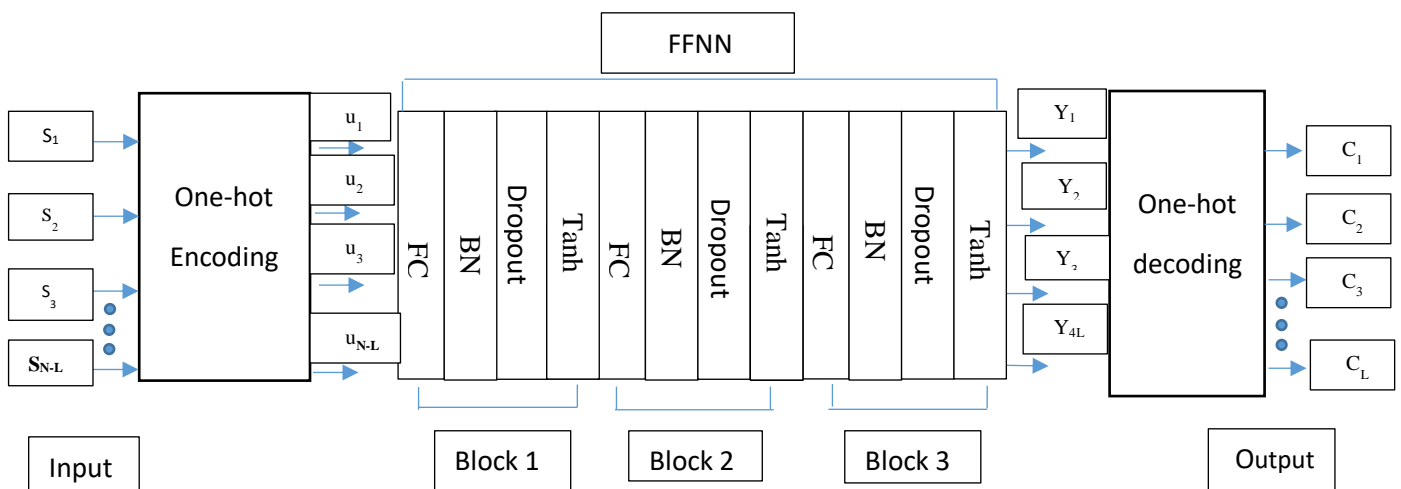


Figure 3. 1 : Le schéma de principe du codeur FFNN

Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

Afin d'obtenir un PAPR précis, le facteur d'échantillonnage doit satisfaire $W \geq 4$ [14]. Plus précisément, un échantillonnage de WN points peut être réalisé en insérant $(W - 1)N$ zéros dans le signal du domaine fréquentiel à N points, puis en faisant passer la séquence à WN points obtenue par l'unité IFFT. WN points obtenue à travers l'unité IFFT.

Le PAPR d'un symbole OFDM peut être défini comme suit :

$$PAPR = \frac{\max_{0 \leq n \leq WN-1} |x_n|^2}{\mathbb{E}[|x_n|^2]}, \quad (3.2)$$

3.2.2 FFNN :

Le FFNN est une architecture de réseau neuronal profond en anglais signifie (DNN), qui se compose d'une couche d'entrée, de plusieurs couches cachées et d'une couche de sortie. Chaque couche contient plusieurs neurones. Les neurones des couches adjacentes sont connectés les uns aux autres, tandis que les neurones d'une même couche sont indépendants. Dans le FFNN, le neurone multiplie toutes les entrées avec les poids correspondant et ajoute un biais, il sera traité ensuite par une fonction d'activation non linéaire tel que sigmoïde, tanh et ReLU. L'algorithme de rétro propagation ajuste de manière adaptative les poids entre les neurones en utilisant l'algorithme de descente de gradient stochastique (SGD) pour minimiser la fonction de perte. Il a été considéré comme un algorithme d'apprentissage valide qui accomplit les tâches telles que la classification et la régression pour le FFNN. L'algorithme SGD nécessite des stratégies de taux d'apprentissage soigneusement conçu pour obtenir des performances satisfaisantes. C'est pour cette raison que plusieurs algorithmes de taux d'apprentissages adaptatifs ont été proposés, tels que Adagrad, RMSProp, Momentum, et Adam, on utilise une couche de normalisation de lot derrière la couche entièrement connectée afin d'accélérer le processus d'apprentissage.

3.2.2.1 La fonction de perte :

La fonction de perte communique les performances d'un modèle, elle calcul le taux d'erreur en utilisant la valeur du résultat désiré et le résultat obtenu. Si les prédictions s'écartent trop des résultats réels, la fonction de perte donne un très grand nombre. Par contre, plus la valeur de la perte est proche de zéro, plus le taux d'erreur se minimise et le résultat du modèle sera plus performant et réel.

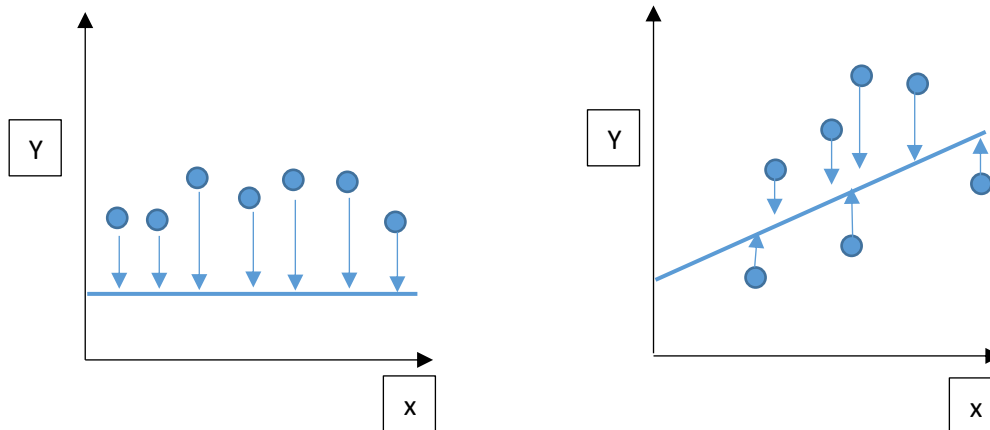


Figure 3. 2 : La representation de la fonction de perte

Il existe plusieurs fonctions de perte classées en deux catégories (les fonctions de perte de régression, et les fonctions de perte de classification).

3.2.2.2 Les fonctions d'optimisation :

Les fonctions d'optimisation ont pour le but de minimiser les couts ou le taux d'erreur par l'assurance de la mise à jour des paramètres du modèle. Par conséquent, pour trouver la valeur optimale de ces paramètres, il faut faire la mise à jour éventuellement jusqu'à ce que le taux d'erreur soit le minimum possible en se basant sur une règle de mise à jour à cet effet.

Différents algorithmes d'optimisation ont une approche différente pour calculer, mettre à jour et trouver la valeur optimale des paramètres du modèle.

3.2.2.3 Descente de gradient :

Il représente un algorithme qui assure des calculs mathématique pour qu'on puisse connaitre comment ajusté les paramètres d'un modèle. Le principe de descente gradient est de minimiser la fonction de perte en modifiant les poids et les biais par calcul des gradients (dérivés) de la fonction de cout par rapport à chaque paramètre du modèle. Ce processus se poursuit jusqu'à ce que le minimum atteigne.

Les paramètres seront ajustés comme suit :

$$W = W - e \frac{dC}{dW}$$

$$b = b - e \frac{dC}{db}$$

W : Le poids.

b : Biais.

e : Le taux d'apprentissage.

C : La fonction de cout.

3.2.2.1.1 Les algorithmes d'optimisation de la descente de gradient :

Il existe plusieurs algorithmes d'optimisation qui implémentent la descente de gradient, les plus utilisées sont [16] :

- **RMSPROP :**

RMSprop est une technique d'optimisation proposé par Geoffrey Hinton. Cet algorithme ne nécessite pas d'ajuster le taux d'apprentissage manuellement. Il choisit un taux d'apprentissage différent pour chaque paramètre automatiquement. Cela a pour effet d'équilibrer la taille de l'étape et de diminuer l'étape d'un grand gradient pour éviter d'exploser, et d'augmenter l'étape pour un petit gradient pour éviter de disparaître.

- **ADAM**

Le nom ADAM est dérivé d'Adaptive Moment Estimation. Il représente un algorithme d'optimisation présenté en 2015. Adam est une méthode qui assure le calcul du taux d'apprentissage adaptatif pour chaque paramètre. Adam assure le stockage d'une moyenne en décomposition exponentielle des gradients carrés passés, aussi il conserve également une moyenne en décomposition exponentielle des gradients passés.

Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

- **La propagation :**

La propagation du signal d'entrée vers les sorties passant par les couches cachées s'effectue étape par étape c.à.d. d'une couche à une autre, prenons l'exemple d'une architecture à une seule couche cachée, la première étape se fera suivant les relations suivantes :

$$R_j = \sum_{i=1}^I \sum_{j=1}^J (E_i \times \omega_{ji}) , \quad (3.3)$$

$$C_j = F(R_j), \quad (3.4)$$

où :

I : Représente le nombre d'entrées.

J : Représente le nombre des neurones dans la couche cachées.

E_i : Représente l'entrée i du réseau.

ω_{ji} : Représente le poids liant le neurone de la couche cachée j à l'entrée i .

F : Représente la fonction d'activation.

C_j : Représente la sortie j de la couche cachée, et l'entrée j de la prochaine couche.

Dans notre étude, on a utilisé la fonction sigmoïde comme fonction d'activation, donc l'équation (3.4) pourra s'écrire ainsi :

$$C_j = \frac{1}{1+e^{-R_j}} , \quad (3.5)$$

On répétera la même procédure et on utilisera les mêmes formules pour la propagation de la couche cachée vers les sorties :

$$R_k = \sum_{j=1}^J \sum_{k=1}^K (C_j \times \omega_{kj}) , \quad (3.6)$$

$$S_k = \frac{1}{1 + e^{-R_k}}$$

où :

K : Représente le nombre de sorties.

C_j : Représente l'entrée j de la couche de sortie.

ω_{kj} : Représente le poids liant la sortie k au neurone de la couche cachée j .

S_k : Représente la sortie k du réseau.

- **L'erreur quadratique :**

D'une manière globale, l'erreur quadratique moyenne M.S.E (Mean Square Error) peut se définir comme étant une grandeur permettant de comparer des estimateurs entre eux, c'est exactement le même cas pour les réseaux de neurones, les estimateurs à comparer seront la sortie obtenue par le réseau après le calcul de la propagation et la sortie attendue :

$$e = \sum_{n=1}^N \sum_{k=1}^K \frac{1}{2} (D_n - S_{kn})^2 , \quad (3.7)$$

Où :

n : Représente l'indice des échantillons N .

S_{kn} : Représente la sortie obtenue par le réseau.

D_n : Représente la sortie désirée (la sortie attendue).

e : Représente l'erreur commise par le réseau.

- **La rétropropagation** : L'étape la plus importante consiste à minimiser l'erreur e , on peut écrire l'équation (3.7) de cette façon :

$$e = \sum_{n=1}^N \sum_{k=1}^K \frac{1}{2} \left(D_n - \frac{1}{1 + e^{-(C_j \times W_{kj})}} \right)^2 , \quad (3.8)$$

Cette erreur possède donc un minimum, et dépend de ω , le but est donc de mettre à jour ces pondérations connexions (ω_{ji} et ω_{kj}) afin de trouver l'erreur la plus minimale possible (minimum globale comme le montre la figure ci-dessous), ce calcul se fera inversement, c.à.d.

Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

des sorties vers les entrées, d'où l'appellation de "Rétropropagation", et afin d'y parvenir, on calculera le gradient de e par rapport à ω_{ji} et ω_{kj} suivant la relation suivante :

$$\nabla e(\vec{\omega}) = \frac{\partial e}{\partial \omega} , \quad (3.9)$$

où :

∇ : fait référence au gradient d'une fonction



Figure 3.3 : Courbe montrant les positions des minimas [15].

Le calcul direct est quasiment impossible, donc afin de simplifier l'équation on utilisera le théorème des dérivées à fonctions composées appelées aussi théorème de chaîne, par exemple pour mettre à jour les poids ω_{kj} il suffira de calculer le produit suivant :

$$\frac{\partial e}{\partial \omega_{kj}} \partial \omega_{kj} = \frac{\partial e}{\partial S_k} \times \frac{\partial S_k}{\partial R_k} \times \frac{\partial R_k}{\partial \omega_{kj}} , \quad (3.10)$$

Ce gradient sera multiplié par un facteur réel η ($0 \leq \eta \leq 1$) appelé "taux d'apprentissage", c'est un paramètre qui contrôle la vitesse à laquelle les pondérations sont ajustées, lorsque η est petit, l'algorithme convergera mais le nombre d'itérations va être beaucoup trop grand, par contre dans le cas où η est élevé, la valeur du produit $\eta \times \frac{\partial e}{\partial \omega_{kj}}$ augmentera, l'erreur risque d'osciller autour d'un minimum sans qu'il y ait convergence. L'ajustement des poids ω_{kj} se fera en effectuant la formule suivante :

$$\omega_{kj}^{p+1} = \omega_{kj}^p + \Delta \omega_{kj}^p , \quad (3.11)$$

p : représente l'indice de l'itération actuelle.

$p + 1$: représente l'indice de la prochaine itération.

Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

$$\Delta\omega_{kj} = -\eta \times \frac{\partial e}{\partial \omega_{kj}} , \quad (3.12)$$

A noter que pour éviter les problèmes liés à une stabilisation dans un minimum local, on peut ajouter un terme d'inertie β appelé "momentum", celui-ci permet de sortir des minimums locaux dans la mesure du possible et de poursuivre la descente de la fonction d'erreur. A chaque itération, le changement de poids conserve les informations des changements précédents. Cet effet de mémoire permet d'éviter les oscillations et accélère l'optimisation du réseau. L'équation III.10 devient ainsi :

$$\Delta\omega_{kj}^p = -\eta \times \frac{\partial e}{\partial \omega_{kj}} + \beta \times \Delta\omega_{kj}^{p-1} , \quad (3.13)$$

$p - 1$: représente l'indice de la précédente itération actuelle.

3.3 TR-NET :

La performance PAPR de la technique TR est considérablement limitée par l'ensemble de candidats fini. Dans cette technique le symbole réservé est choisi parmi un ensemble de candidats. Pour former le modèle qui peut générer le vecteur de symbole réservé C basé sur les caractéristiques du signal d'entrée, on utilise la capacité de mappage non linéaire du FFNN.

Le schéma TR-NET améliore l'efficacité de la bande passante et la performance de réduction du PAPR par rapport à la méthode TR conventionnelle.

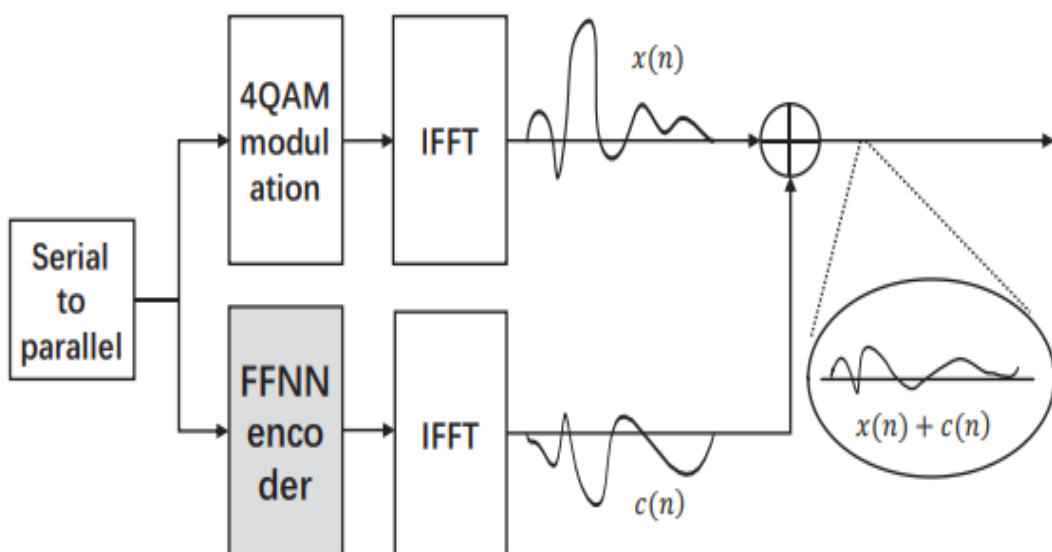


Figure 3. 4 : Vue d'ensemble du TR-NET

Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

Le signal en bande de base est copié pour obtenir deux duplicata. L'un des doubles est utilisé pour générer le signal de données dans le domaine temporel x_n en utilisant le schéma de modulation 4-QAM et l'opération IFFT, tandis que l'autre double est introduit dans le codeur FFNN pour générer le vecteur de symbole réservé C dans le domaine des fréquences. C , puis C est introduit dans le module IFFT pour obtenir le signal d'annulation de crête c_n . Le signal OFDM à PAPR réduit peut être exprimé comme suit :

$$\hat{x}_n = x_n + c_n. \quad (3.14)$$

3.4 Structure du TR-NET :

La partie réseau neuronal du TRNet se compose de cinq couches, une couche d'entrée, une couche de sortie et de trois couches cachées. Toutes les couches sont connectées côte à côte. Chaque couche cachée est constituée d'une couche entièrement connectée (**FC (fully connected)**), d'une normalisation de lot (**BN (batch normalization)**), et la fonction tangente hyperbolique (tanh).

➤ La couche FC :

Le cœur de la couche FC est la multiplication de la matrice qui est équivalente à une transformation de l'espace des caractéristiques. Plus précisément, si l'entrée de la couche FC est $(x_1, x_2, \dots, x_n) \in \mathbf{R}^n$, alors la sortie de la couche FC peut être exprimée comme suit :

$$y_j^{(k)} = \sum_{i=1}^N w_{ji}^{(k)} x_i + b_j^{(k)}, \quad (3.15)$$

Où :

$w_{ji}^{(k)}$: Est le poids de la i -ème entrée dans la k -ème couche de matrice de poids connectée au j -ème neurone de la couche cachée.

➤ La couche BN :

La couche BN normalise la sortie de la couche FC, ce qui évite le problème de la disparition du gradient et augmente la vitesse d'apprentissage. Pour créer cette couche (BN) on utilise `batchNormalizationLayer`. Plus précisément, BN peut être exprimée comme suit :

$$\hat{y}^{(k)} = \gamma \frac{y^{(k)} - E(y^{(k)})}{\sqrt{D(y^{(k)}) + \epsilon}} + \beta, \quad (3.16)$$

Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

Où :

- ϵ : est une petite valeur qui évite que le dénominateur soit nul.
 - γ et β : sont des paramètres ajustables, qui renforcent l'expressivité du réseau neuronal.
- Cela signifie que pour accélérer la convergence, le réseau neuronal doit ajuster de manière adaptative la distribution dans le processus d'apprentissage.

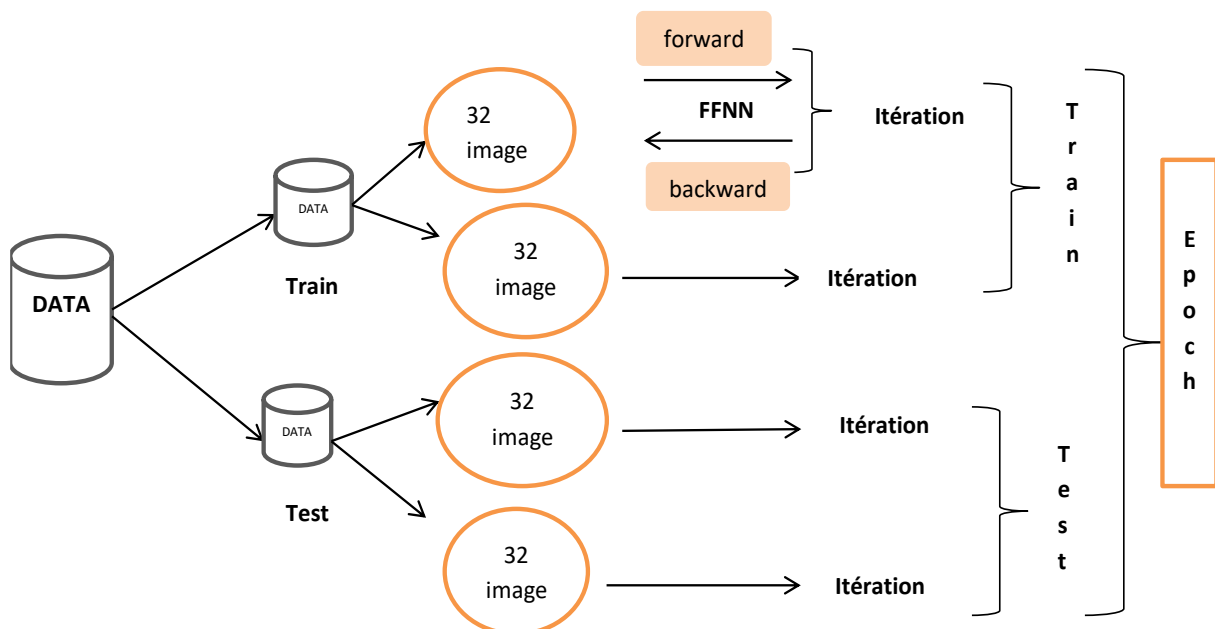
Remarque :

On a utilisé Batch parce que epoch est trop grande pour l'alimentée à l'ordinateur à la fois, nous la divisons en plusieurs petit batch. Epoch qui signifie un aller (forward) et un retour (backward) de tous les entrainements.

Batch size : est le nombre total d'exemples d'entraînement présents dans un seul batch.

Itération : est le nombre de batch nécessaires pour compléter un epoch

Voici un schéma final pour mieux comprendre :



Chapitre 3 : Réduction de PAPR avec la méthode TR-NET

➤ L'algorithme dropout :

Dans le schéma proposé, le modèle formé présente un problème de sur-ajustement. Afin de résoudre ce problème, on utilise l'algorithme d'abandon (dropout algorithm) qui réduit le sur-ajustement en ignorant la moitié des détecteurs de caractéristiques dans chaque lot d'apprentissage (la moitié des neurones de la couche cachée sont nuls). L'algorithme de décrochage (dropout algorithm) peut être exprimé comme :

$$r_j^{(k)} \sim \text{Bernoulli}(p)$$

$$\tilde{y}^{(k)} = r^{(k)} * \hat{y}^{(k)},$$

Où

r : Est un vecteur aléatoire satisfaisant à la distribution binomiale (distribution 0-1).

p : Désigne la probabilité de 1

$1 - p$: Désigne la probabilité de 0.

Cette méthode réduit les interactions entre les détecteurs de caractéristiques.

➤ La fonction tanh :

La fonction ***tanh*** est la fonction d'activation pour TR-NET, qui peut être exprimée comme :

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}},$$

Les données d'entrée de la fonction d'activation seront limitées à $[-1, 1]$ après l'opération BN. Et ***tanh*** est très proche de $y = x$ dans cet intervalle. Par conséquent, il peut directement effectuer l'opération matricielle en fonction de la nature de la fonction d'identité, ce qui peut améliorer l'efficacité de la formation.

3.5 Prétraitement des données :

Les réseaux de neurones sont incapables de traiter directement le nombre complexe. Pour résoudre ce problème, nous utilisons le vecteur de valeur réelle et le codage one-hot pour reparamétriser les données d'entrée et de sortie du réseau de neurones. Les opérations spécifiques sont les suivantes

Tout d'abord, le codage one-hot est effectué sur les données d'entrée du réseau neuronal. Nous attribuons un vecteur unitaire pour chaque donnée binaire d'entrée possible. La règle d'encodage est la suivante : nous utilisons N bits pour représenter un symbole modulé

N -aire, dans lequel un seul bit vaut 1, et les autres bits sont nuls. Par exemple, si on utilise la modulation 4-QAM, alors les codes à un coup correspondants peuvent être définis comme suit

$$s1 = 00 \leftrightarrow u1 = [1, 0, 0, 0]$$

$$s2 = 01 \leftrightarrow u2 = [0, 1, 0, 0]$$

$$s3 = 10 \leftrightarrow u3 = [0, 0, 1, 0]$$

$$s4 = 11 \leftrightarrow u4 = [0, 0, 0, 1]$$

Deuxièmement, les données de sortie du réseau neuronal sont décodées de la même manière.

Plus précisément, tous les quatre bits consécutifs des données de sortie $Y \in R^{4L}$ sont utilisés pour représenter un symbole MAQ-4. Nous pouvons représenter cette règle de mappage par la fonction $f_{onehot}(Y)$, qui peut être définie comme suit :

$$\begin{aligned} C_i &= f_{onehot}(Y) \\ &= (Y_{4i-3} - Y_{4i-1}) + j(Y_{4i-2} - Y_{4i}), \end{aligned}$$

Où

$$i = 1, 2, \dots, L$$

3.6 Formation de TR-NET :

Dans notre schéma, l'encodeur FFNN est formé pour générer le vecteur symbole réservé C et réduire le PAPR du signal composite émis. L'objectif est de minimiser la fonction de perte en mettant à jour les paramètres, qui peuvent être écrits comme suit :

$$\begin{aligned} LOSS &= PAPR\{\hat{x}\} \\ &= PAPR\{x + c\} \\ &= PAPR\{Q(X + C)\}, \end{aligned}$$

Où

\hat{x} : est le signal transmis qui est créé en additionnant le signal d'annulation de crête c généré par le codeur FFNN et le signal de données x .

3.7 Conclusion :

Dans ce chapitre, nous avons proposé un schéma de réduction du PAPR basé sur l'apprentissage profond, à savoir TRNet. TR-NET génère de manière adaptative le signal d'annulation de crête en fonction des caractéristiques du signal d'entrée.

Chapitre 4

Paramètres et

résultats des

simulations

Chapitre 4 : Paramètres et résultats des simulations

4.1 Introduction :

La simulation numérique est une étape importante dans l'étude et la mise en œuvre des systèmes réalistes de transmission. En effet, elle permet d'une part la validation des études théoriques et d'autre part l'optimisation du système par l'analyse de l'impact des divers paramètres sur les performances effectuer une comparaison et une analyse des résultats obtenus. Dans ce chapitre nous allons présenter les résultats des simulations en utilisant le logiciel « **MALTB** » et la toolbox Deep Learning. Nos simulations ont pour but de :

- Calculer les fonctions CCDF
- Réduire le PAPR avec la méthode TR classique
- Réduire le PAPR par deep learning avec la méthode TRnet

4.2 Les effets du non linéarité de l'amplificateur sur le signal OFDM :

Dans cette section, nous allons faire varier le recul à l'IBO (input back off) et observer le spectre à la sortie. Le but de cette section, est de montrer les effets du non linéarité causés par un amplificateur de puissance. La figure ci-dessous montre les spectres de puissance sans et avec amplification par un modèle SSPA (Solid State Power Amplifier) pour différentes valeurs de CR.

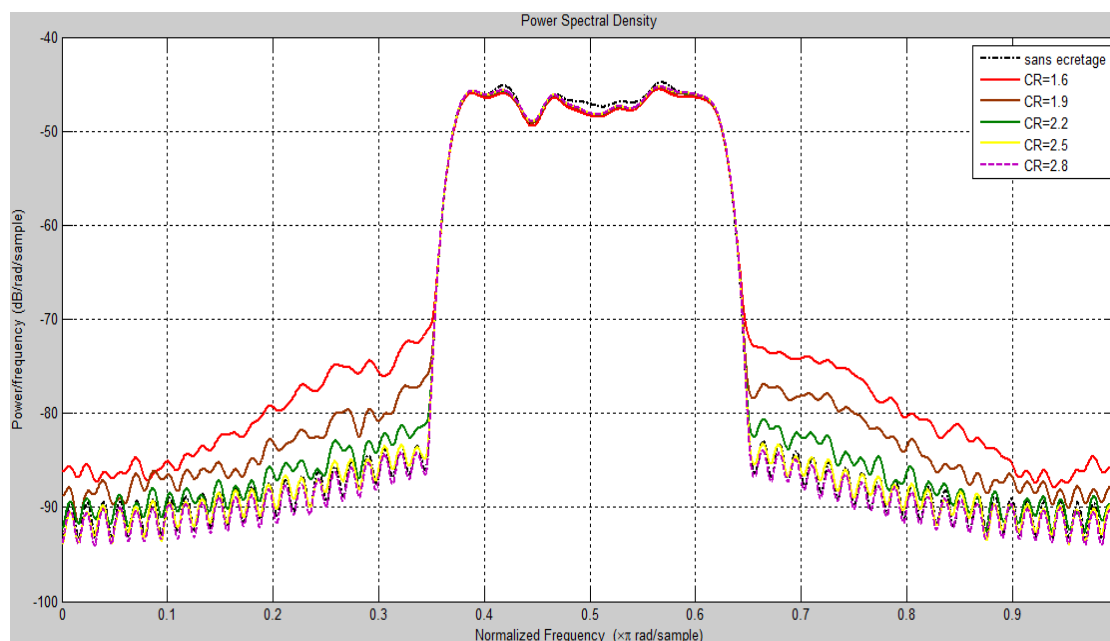


Figure 4. 1 : spectre sans et avec amplification par un modèle SSPA pour différente valeur de C

Chapitre 4 : Paramètres et résultats des simulations

D'après la figure 4.1, on remarque qu'en diminuant IBO c'est-à-dire diminution de la zone linéaire de l'amplificateur, la puissance en dehors de la bande augmente qui est due aux harmoniques, et une diminution de la puissance dans la bande causées par les intermodulations. Cependant les intermodulations sont les plus gênantes, car on ne peut pas les filtrer contrairement aux harmoniques. Ces distorsions peuvent être considérées comme un bruit supplémentaire rajouté au signal.

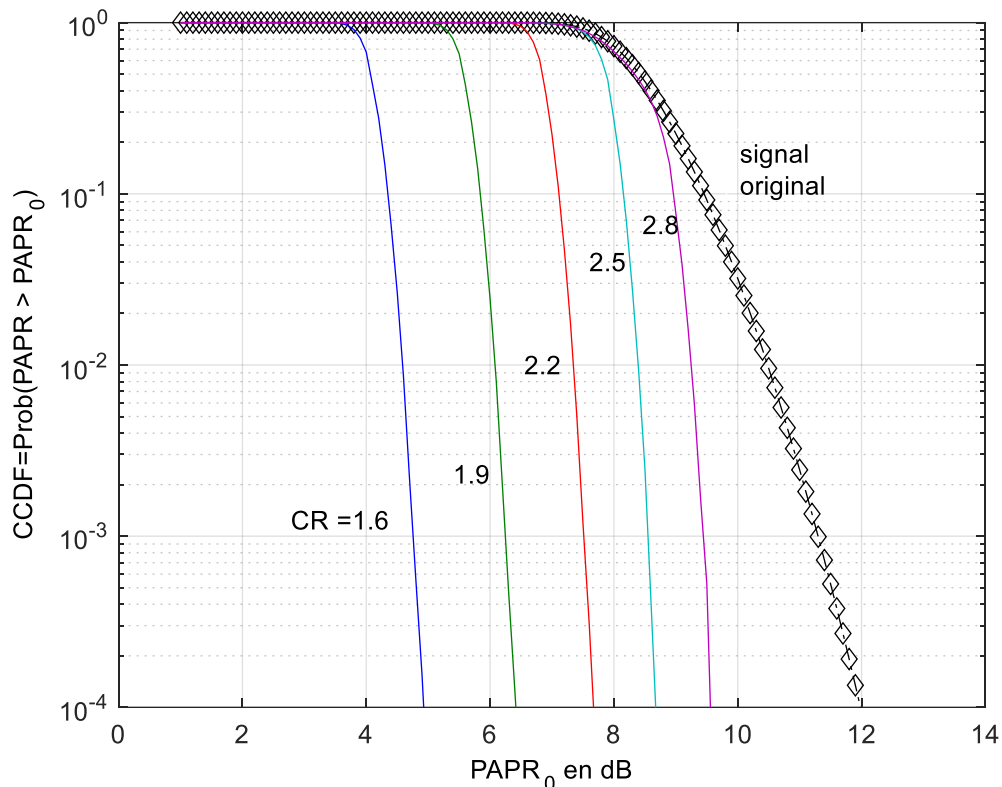


Figure 4. 2 : La fonction CCDF d'un signal OFDM

La figure 4.2, illustre les graphes de la fonction de répartition du PAPR (CCDF) d'un signal OFDM sans écrêtage et avec écrêtage pour différentes valeurs du taux d'écrêtage (CR). Le signal sans écrêtage présente un PAPR de 12 dB pour un CCDF = 10^{-4} . Le signal avec un écrêtage élevé (CR=1.6) donne le PAPR le plus réduit selon sa fonction CCDF à 10^{-4} avec une réduction d'environ 7 dB. Le signal avec l'écrêtage le plus faible (CR=2.8), possède le PAPR le plus fort parmi les signaux qui ont subi l'écrêtage. Donc si on compare les figures 4.1 et 4.2, on conclut que l'écrêtage à un niveau très fort permet de réduire le PAPR d'une manière assez importante mais il cause une distorsion du signal qui se manifeste par une remontée de la puissance en dehors de la bande.

Chapitre 4 : Paramètres et résultats des simulations

Il serait plus judicieux d'utiliser une technique de réduction sans distorsion. La méthode appelée Tone Reservation (TR), permet une réduction du PAPR sans d'effets indésirables.

4.3 Technique de réduction :

Avant de commencer les méthodes de réduction, nous allons d'abord présenter les paramètres de simulation.

Paramètres de simulation	TR-LP	TRnet
Modulation	4-QAM	4-QAM
taille des symboles non nuls k	128	128
nombre de symboles réservés u	16 32 8	8
Facteur de sur échantillonnage L	4	4

Tableau 4. 1: Paramètres de simulation

4.4 La technique TR :

Nous avons tracé dans un premier temps la fonction CCDF d'un signal OFDM original (sans aucune réduction) obtenue d'une manière analytique (théorique) et par simulation. Pour cela on va réaliser une simulation de Monte Carlo en générant 10000 symboles OFDM aléatoires. Les signaux sont modulés en 4QAM avec un $N=128$ porteuses non nulles. Parmi ces porteuses, certaines sont réservées à la réduction du PAPR en utilisant la méthode TR. L'optimisation peut se faire selon deux méthodes :

- La TR avec l'algorithme de programmation linéaire (TR-LP).
- La TR avec l'algorithme de gradient (TR-GR).

4.5 La méthode TR avec l'algorithme de programmation linéaire (TR-LP) :

La programmation linéaire est utilisée comme méthode mathématique pour déterminer et planifier les meilleurs résultats. Cette programmation fait partie d'un domaine important appelé « techniques d'optimisation » car elle est littéralement utilisée pour trouver la solution la plus optimisée à un problème donné. Cette méthode est basée sur l'algorithme du simplexe et le nombre maximum d'itération considéré est égale à 100.

La figure ci-dessus illustre les résultats des fonctions CCDF en utilisant la technique de réduction TR-LP, sachant que le nombre total des porteuses est $N= 128$, et une modulation

Chapitre 4 : Paramètres et résultats des simulations

4-QAM, avec un facteur de suréchantillonnage $L=4$. Le nombre de sous-porteuses réservées varie entre 4 et 32. Les graphes des CCDF en générant toujours 10.000 symboles OFDM sont illustrées par la figure 3.3.

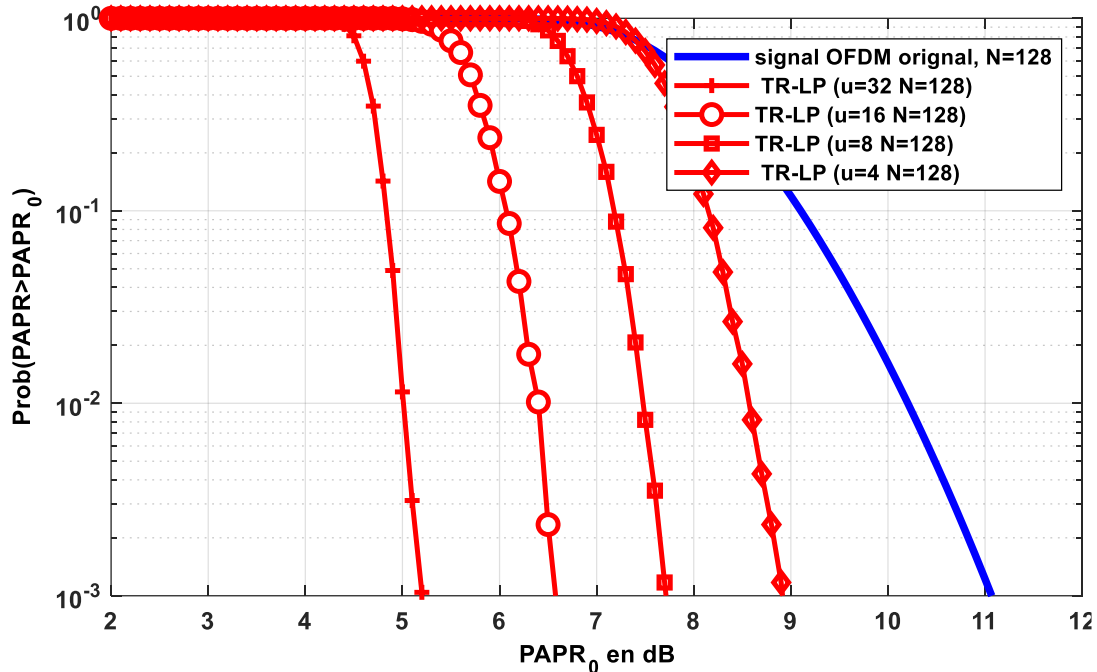


Figure 4.3 : Les fonctions CCDF du PAPR des signaux OFDM avec la méthode TR-LP

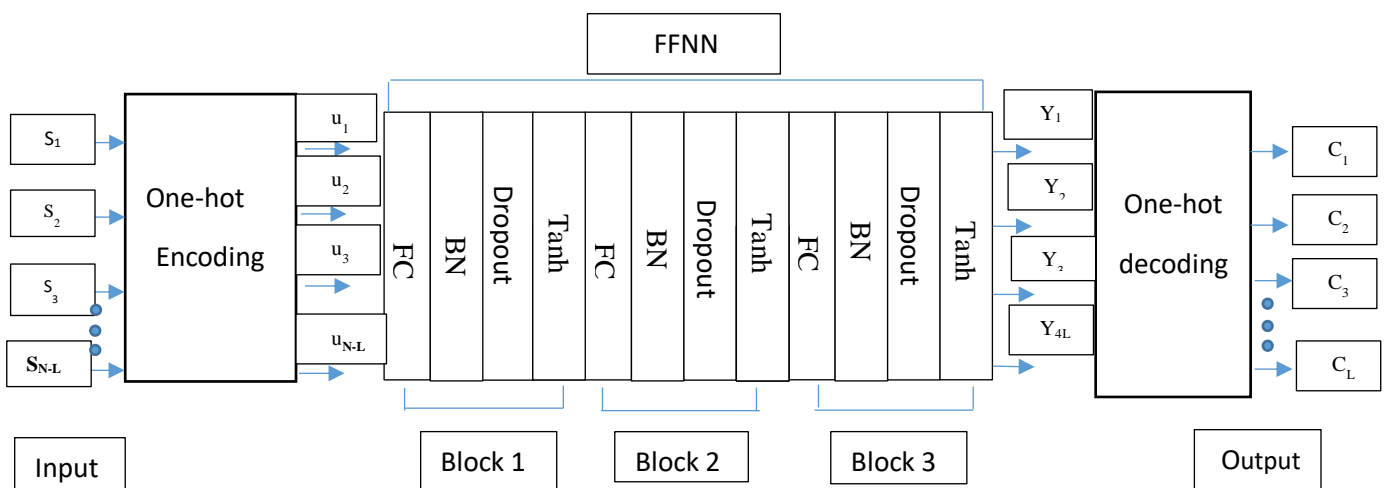
Dans cette figure, on constate, qu'en augmentant le nombre de porteuses réservées u , avec $u = [4, 8, 16, 32]$, ceci entraîne une diminution significative du PAPR. Il faut signaler que le prix de cette réduction se fait au détriment d'une diminution du débit en fonction du nombre de porteuses réservées avec les pourcentages [3.125%, 6,25%, 12.5%, 25%] respectivement. La complexité calculatoire engendrée augmente significativement, et elle est importante en fonction du nombre de sous-porteuses réservées aussi. Pour un CCDF 10^{-3} et en utilisant $u=4$ porteuses réservées parmi les 128, le PAPR de la méthode TP-LP est d'environ 8.9 dB, tandis que celui du signal OFDM c'est à dire la courbe originale sans réduction, est de 11.1 dB, donc une différence de 2.2 dB. Pour $u=32$ porteuses réservées, le PAPR diminue d'environ 5.8 dB par rapport à la courbe originale sans réduction.

Chapitre 4 : Paramètres et résultats des simulations

4.6 Méthode TR-NET

Dans cette partie, on présente les résultats des simulations obtenus en utilisant un réseau Deep learning. Le détail de la méthode a été donné au chapitre 3. Le réseau consiste en 03 blocs, chaque bloc comporte 3 couches (FN, BN, Dropout) et une fonction d'activation tanh.

```
layers = [  
sequenceInputLayer(InputSize, , 'none', 'Name', 'xn')  
batchNormalizationLayer('Name', 'bnorm1')  
dropoutLayer(dropoutProb, 'Name', 'dropout1')  
fullyConnectedLayer(InputSize, 'Name', 'Fc1')  
tanhLayer('Name', 'tanh1')  
  
fullyConnectedLayer(InputSize, 'Name', 'Fc2')  
batchNormalizationLayer('Name', 'bnorm2')  
dropoutLayer(dropoutProb, 'Name', 'dropout2')  
tanhLayer('Name', 'tanh2')  
  
fullyConnectedLayer(4*u, 'Name', 'Fc3')  
batchNormalizationLayer('Name', 'bnorm3')  
dropoutLayer(dropoutProb, 'Name', 'dropout3')  
tanhLayer('Name', 'tanh3')];  
  
lgraph = layerGraph(layers);  
dlnet = dlnetwork(lgraph);
```



Lors de la phase d'apprentissage, un grand nombre de signaux a été généré. Cette base de données doit être la plus diverse possible. La base des données générées est stockée ensuite utilisée pour l'apprentissage sur le réseau proposé.

Le réseau utilisé n'est pas un réseau supervisé, ce qui veut dire que veut dire qu'il n'y a pas de sortie désirée. La sortie du réseau est directement utilisée pour réduire la fonction LOSS, autrement dit le PAPR. Ceci a rendu la construction de la boucle d'apprentissage plus difficile

Chapitre 4 : Paramètres et résultats des simulations

à réaliser surtout que cette sortie doit être décodée pour construire un signal complexe et utilisée pour la minimiser la fonction LOSS.

Un problème a été rencontré lors de cette évaluation, car ceci exige de sortie du format **dlarray** utilisé par la toolbox deep learning pour introduire les données d'apprentissage **X**.

```
idx = (i-1)*miniBatchSize+1:i*miniBatchSize;  
    X = XTrain(:,idx);  
    xf=xif(:,idx);  
dlX = dlarray(X, 'CBT');
```

Lors de l'apprentissage, après la configuration de l'architecture du réseau, l'algorithme utilisé va d'abord générer des poids d'une manière aléatoire pour la couche FC. Ensuite une mise à jour est réalisée en calculant la fonction LOSS et le gradient d'une manière numérique.

```
function [gradients,state,loss] = modelGradients(dlnet,dlX,xf,M,Indices_C,Q,u)  
  
[dlYPred,state] = forward(dlnet,dlX);  
loss = myLoss(xf,dlYPred,M,Indices_C,Q,u);  
gradients = dlgradient(loss,dlnet.Learnables);  
  
end
```

L'apprentissage a été réalisé sur 200 batches, et un taux d'apprentissage (learning rate) de 0.001. Pour terminer l'apprentissage, il fallu faire plusieurs testes et plusieurs modifications. Cette a été de loin la plus difficile à achever.

Une fois l'apprentissage terminé, on genere 10000 symboles OFDM pour tracer la fonction CCDF. Les resultats obtenus sont illustrés par la figure suivante.

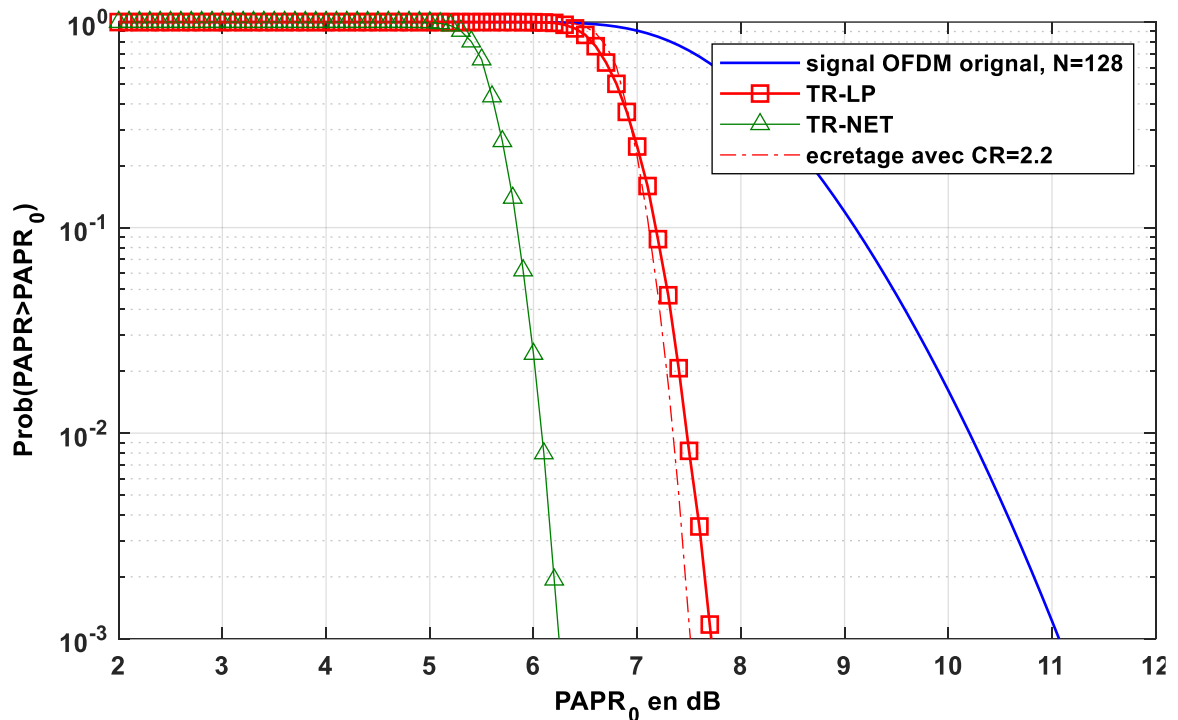


Figure 4. 4: CCDF de la méthode TR-NET avec $u=8$, $N=128$.

La figure 4.4, montre la supériorité en termes de réduction du PAPR de la méthode TR-NET en comparant avec la méthode classique TR-LP ou aussi avec la méthode d'écèlement avec $\text{CR}=2.2$. Une amélioration de 1.2 dB en réduction du PAPR a été observée en utilisant le deep learning. On peut aussi rappeler que la méthode TR-NET est une méthode sans distorsion.

4.7 Conclusion :

Les résultats de la simulation ont montré que le schéma proposé peut réduire efficacement le PAPR des signaux OFDM par rapport à la technique TR conventionnelle. D'autre part, en tant que schéma sans distorsion, TR-NET n'a pas besoin de compenser la perte de BER, ce qui améliore considérablement la vitesse de formation.

Conclusion Générale :

Les travaux réalisés dans notre thème de mémoire ont porté sur la réduction du PAPR dans les systèmes OFDM par le deep learning. Nos recherches à travers les documents consultés, nous ont permis de déceler et de comprendre les diverses facettes concernant la réduction du PAPR et le deep learning. Ceci a permis de valider nos recherches et réponses aux questions posées selon la problématique formulée dans le cadre de notre mémoire de fin d'études. Dans ce travail, nous avons commencé tout d'abord par définir le deep learning ou l'apprentissage profond, qui est un sous-domaine de L'intelligence artificielle (IA), et qui aussi désigne l'ensemble des techniques d'apprentissage automatique (machine learning), autrement dit une forme d'apprentissage fondée sur des approches mathématiques. La modulation OFDM qui reste une technique largement utilisée en communication numérique, souffre du problème du PAPR élevé. L'état de l'art sur les techniques de réduction du PAPR en se basant sur les différents critères de ce dernier (compatibilité descendante, diminution du débit utile et complexité) a été aussi dressé, ce qui nous a permis d'évaluer les performances de toutes techniques de réduction du PAPR. Nous nous sommes focalisés sur la réduction du PAPR en associant un réseau deep learning à la méthode connue sous le nom de TR (Tone Reservation). La méthode appelée TR-NET par rapport à la méthode TR classique nécessite moins de sous porteuse. Le TRnet utilise un réseau neuronal nommée, FFNN au niveau de l'émetteur contrairement aux autres schémas de réduction du PAPR qui utilisent des réseaux neuronaux dans l'émetteur et le récepteur, pour générer le signal de réduction en fonction des caractéristiques du signal d'entrée. Les simulations que nous avons réalisées ont été obtenues en utilisant la toolbox deep learning de Matlab 2020. La méthode appelée TR-NET génère de manière adaptative le signal de réduction du PAPR en fonction des caractéristiques du signal d'entrée. Les résultats de la simulation ont montré ce schéma peut réduire efficacement le PAPR des signaux OFDM par rapport à la technique TR conventionnelle.

Bibliographie :

[1] : S.Russell et P.Norvig, "Introduction à l'Intelligence Artificielle",

Ces notes ont été préparées pour un cours du niveau L3 offert en télé-

Enseignement par l'Université Aix-Marseille 1 en 2007-2008 et 2008-2009.

[2] : J.p.Muller et L.Massaron, "Deep Learning for Dummies"

[3] : Y.Lecun, "Intelligence Artificielle".

[4] : K. Dahmane, "Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées ", Vision par ordinateur et reconnaissance de formes,25 Novembre 2020.

[5] : J. Delon,"Introduction aux réseaux de neurones et à l'apprentissage profond ", 21 aout 2018.

[6] : H. T, " Réseaux de neurones récurrents ", *Neural Networks and Analog Computation*, 153-164, 1997

[7] : S. Arab, "Une étude sur la réduction du PAPR dans un système OFDM ", 2010

[8] : S.Qintuya et J.Minglu, "A Novel Tone Reservation Scheme Based on Deep Learning for PAPR Reduction in OFDM Systems Benwei Wang ",

[9] : S. B. Weinstein et P. M. Elbert, "Data Transmission by Frequency-Division Multiplexing Using the Discrete Fourier Transform", *IEEE Transactions on Communications*, pp. 628-634, octobre 1971.

[10] : J. G. Proakis, "Digital communications", McGraw-Hill, 4e édition, 2001.

[11] : R. Van Nee et R. Prasad, "OFDM for Wireless Multimedia Communications", Artech House Publishers, 4e édition, 2000.

[12] : C. T. Lam, D. D. Falconer et F. Danilo-Lemoine, "PAPR Reduction using Frequency Domain Multiplexed Pilot Sequences", *Wireless Communications and Networking Conference, IEEE WCNC 2007*, 11-15 mars 2007.

[13] : A. Mobasher et A. K. Khandani, "Integer-Based Constellation-Shaping Method for PAPR Reduction in OFDM Systems", IEEE Transactions on Communications, pp. 119-127, janvier 2006.

[14] : C. Tellambura, "Computation of the continuous-time PAR of an OFDM signal with BPSK subcarriers," IEEE Commun. Lett., May, 2001

[15] : M. Kim, W. Lee, and D. Cho, "A novel PAPR reduction scheme for OFDM system based on deep learning," IEEE Commun. Lett., vol. 22, no. 3, pp. 510–513, Mar, 2018

[16] : P. Foomooljareon and W.A.C. Fernando, "PAPR Reduction in OFDM Systems", Thammasat Int. J. Sc.T ech., Vol. 7, No. 3, September-December 2002.