

République Algérienne Démocratique et Populaire.
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique.
Université Mouloud Mammeri de Tizi-Ouzou.
Faculté de Génie Electrique et D'informatique.
Département D'électronique.



MEMOIRE



DE FIN D'ETUDES

*En vue de l'obtention du diplôme d'ingénieur d'état en
Electronique.*

OPTION : COMMUNICATION

THEME

*ETUDE DES SYSTEMES DE TELECOMMUNICATION
ET DE LA COUCHE PHYSIQUE DE LA MODULATION
OFDM NORME IEEE 802.16*

Dirigé par :

Mr.M.Tahanout

Réalisé par :

*BOUBEKEUR Karima
BELMOKHTAR Meriem*

Promotion 2010/2011

Remerciement

Avant tout Merci à mon Dieu le tout Puissant.

Nous tenons à exprimer nos vifs remerciements pour notre promoteur Mr.Tahanout pour sa disponibilité qui nous a été précieuse et pour ses conseils tout au long de notre travail.

A tous les enseignants qui ont contribué à notre formation.

A tout les amis et tous ceux qui nous ont soutenus durant tout le cursus universitaire.

Dédicaces

Je dédie ce modeste travail à :

Ma très chère mère, et mon très cher père, à leur grand sacrifice et le dévouement pour mon bonheur.

Mes frères et sœurs, et toute ma famille.

A mes amis et toute personne qui ma aidée de loin ou de près.

Belmokhtar Meriem

Dédicaces

Je dédie ce modeste travail à :

Ma très chère mère, et mon très cher père, à leur grand sacrifice et le dévouement pour mon bonheur.

Mon frère et mes sœurs, et toute ma famille.

A mes amis et toute personne qui ma aidée de loin ou de près.

Boubekeur Karima

Résumé

Le terme télécommunication signifie communiquer à distance, le but de télécommunication est donc de transmettre un signal porteur d'une information (voix, musique, image...) d'un lieu à un autre situés l'un par rapport à l'autre à une certaine distance.

Un réseau de télécommunication peut être vu comme un ensemble de ressources matérielles mises en place pour offrir un ensemble de services. Dans les dernières années, l'évolution technologique a permis l'augmentation de la capacité des fonctionnalités des ressources et le nombre d'informations transmises. Ainsi, par exemple le succès des services de l'internet a engendré une explosion de trafic qui a mené les opérateurs à utiliser de nouvelles technologies dans le cœur des réseaux telles que IP, NGN qui semblent bien adaptés pour la mise en place voix/données.

Les nouvelles générations de réseaux mobiles tels que le GPRS, l'UMTS permettent de faire des progrès dans le nombre de services et le débit à mettre à la disposition des utilisateurs. La mise en place de réseaux multiservices devrait permettre également de réduire le coût de déploiement des différents services et le coût d'exploitation et de maintenance.

Les communications sans fil font l'objet d'étude d'un grand nombre de chercheurs, ceci dans le but d'obtenir une bonne qualité de communication sans fil avec des débits de transmission élevés et de libérer les usagers des câbles afin de leur permettre de se déplacer tout en restant connectés. Dans cette optique, le Wifi est développé et plus tard le WiMAX. Dans ce type de système de communication il faut combattre adéquatement l'effet des trajets multiples plus précisément l'évanouissement sélectif en fréquence. Ceci a mené au développement et à l'utilisation de la technique du multiplexage par division de fréquence orthogonale (OFDM) qui semble être une solution très prometteuse.

Dans la présente étude nous nous intéressons au système de télécommunication d'une façon générale et à la modulation OFDM d'une façon particulière. Comme ce type de modulation sera utilisé dans les prochaines générations des systèmes de télécommunication sans fil nous nous proposons de faire une étude sur le système OFDM pour comprendre son fonctionnement et ses performances. Pour ce faire, nous nous appuyons sur le toolbox du Simulink de MATLAB pour l'étude et simulation de la norme IEEE 802.16 représentant la couche physique d'une transmission OFDM.

Ce travail est présenté en trois chapitres. Le premier traite des systèmes de télécommunications d'une façon générale. Le deuxième illustre les différents types de modulations du signal utilisées dans les systèmes de transmission. Le dernier chapitre est consacré à l'étude et simulation de la couche physique du système OFDM.

L'OFDM est un système multi porteuses qui garantit une bonne immunité contre les interférences inter porteuses grâce à leur caractéristique principale qui est l'orthogonalité. Dans le schéma bloc d'une liaison de communication par un système OFDM les données à émettre sont organisées en blocs présentées de manière parallèle au module IFFT. Dans une technique OFDM, toutes les porteuses sont modulées par les données d'un même utilisateur. Pour supporter la notion de multi utilisateur, OFDM est combinée soit avec TDMA, soit avec FDMA ou le CDMA.

A l'aide du MATLAB nous avons effectué l'étude de la couche physique d'un système OFDM de la norme IEEE 802.16. Cette étude nous a permis de comprendre toutes les étapes nécessaires pour pouvoir se communiquer en utilisant le système émetteur/récepteur OFDM. A l'émission les données modulent les sous-porteuses en utilisant les modulateurs BPSK, QPSK ou QAM. Les données sont codées et entrelacées pour réduire les erreurs de transmission. Les codes convolutionnel et Reed-Solomon sont utilisés pour permettre de récupérer certains bits erronés. Des fréquences pilotes sont ajoutées pour identifier la bande OFDM. Des séparations sont réalisées dans le spectre de l'OFDM pour éviter un recouvrement spectral à l'aide d'intervalles de garde. Le type de modulation est adapté en fonction du SNR dans le canal de transmission.

Les performances du système OFDM sont testées par simulation. Pour ce faire, nous avons varié le SNR du canal entre 0 et 30 dB pour les deux cas : sans et avec trajets multiples. Plusieurs trames OFDM sont transmises, puis on calcule le taux d'erreur « BER » pour chaque SNR. Les résultats ont montré que le trajet multiple influe considérablement sur la qualité de la transmission. L'erreur de transmission peut atteindre les 50 % dans le cas d'un SNR = 0 dB.

Toutefois, les paramètres de simulation de cette étude sont considérés d'une façon arbitraire. C'est pourquoi ce travail peut être amélioré et étendu en utilisant des paramètres réalistes ou réels. En utilisant le modèle numérique du terrain on peut d'une façon exhaustive simuler l'effet des trajets multiples d'un système OFDM donné et évaluer ainsi, d'une façon proche de la réalité, les performances d'un tel système.

Liste des abréviations

- RTC** : Réseau Télécommunication Comité.
- ZAA** : Zone à Autonomie d'Acheminement.
- CAA** : Commutateur à Autonomie d'Acheminement.
- ZTS** : Zone de Transit Secondaire.
- CTS** : Commutateur de Transit Secondaire.
- ZTP** : Zone de Transit Principale.
- CTP** : Commutateur de transit principal.
- CTI** : Commutateur de transit International.
- CL** : Commutateur Local.
- URA** : Unité de Raccordement d'Abonnés.
- MIC** : Modulation par Impulsion Codée.
- PDH**: Présiochrone Digital Hierarchy.
- SONet** : Synchron Optical Network.
- WLL**: Wireless Local Loop.
- GSM**: Global syteme for Mobile Communication.
- UMTS**: Universal Mobile Telecommunication Systeme.
- BSS**: Base Station Subsysteme.
- NSS**: Network Security Service.
- MS**: Station Mobile.
- BTS**: Base Station Transceiver.
- BSC**: Base Station controller.
- MSC**: Mobile Switching Center.
- VLR**: Visitor Location Register.
- HLR**: Home Location Register.
- G MSC**: Gateway Mobile Switching Center.

PLMN: Public Land Mobile Network.

NGN: Next Generation Network.

IP: Internet Protocol.

WIFI: Wireless Fidelity.

WIMAX: Wireless Interoperability for Micro Wave Access.

QoS: Quality Of Service.

IETF: Internet Engineering Task Force.

WLAN: Wireless Local Area Network.

WMAN: Wireless Metropolitan Area Network.

IEEE: Institute of Electrical and Electronics Engineers.

EDGE: enhanced Data Rate for GSM Evolution.

PSK: Phase Shift Keying.

QPSK: Quadrature Phase Shift Keying.

QAM: Quadrature Amplitude Modulation.

OFDM: Orthogonal frequency Division Multiplexing.

DAB: Digital Audio Broadcasting.

DVB: Digital Video Broadcasting.

ADSL: Asymmetric Digital Subscriber Line.

CDMA: Frequency Division Multiple Access.

FDMA: Frequency Division Multiple Access.

TDMA: Time Division Multiple Access.

STBC: Space Temps Block Coding.

FEC: Forward Error correction.

CC: Code Convolution.

RS: Read Solomon.

BPSK: Binary Phase Shift Keying.

AWGN: Additive White Gaussian Noise.

IFFT: Inverse Fast Fourier Transformation.

PDF: Probability Density Function.

SNR: Signal to Noise Ratio.

BER: Bit Error Rate

PHY: Physical Layer.

PC: Préfixe Cyclique.

TFR: Transformé de Fourier Rapide.

TFD : Transformé de Fourier Discrète.

SOMMAIRE

INTRODUCTION	(1)
Chapitre I : Réseaux de télécommunication en général	
I. 1 Architecture générale du réseau téléphonique	(3)
I.1.1. La commutation.....	(4)
I.1.1.1. Organisation du réseau.....	(4)
I.1.1.2. Règles d’acheminement.....	(6)
I.1.1.3. Fonction principale d’un commutateur.....	(6)
I.1.2. Le réseau de transmission	(7)
I.1.2.1. Hiérarchie numérique plésiochrone	(8)
I.1.3. Le réseau de distribution.....	(8)
I.2. Le GSM	(9)
I.2.2. Architecture de GSM	(10)
I.2.2.1. Le sous système radio	(11)
I.2.2.1.1. station de base (BTS)	(11)
I.2.2.1.2. Le contrôleur de station de base BSC.....	(12)
I.2.2.2. Le sous système réseau.....	(12)
I.2.2.2.1. Le centre de commutation mobile(MSC).....	(13)
I.2.2.2.2. L’enregistreur de localisation nominale(HLR).....	(14)
I.2.2.2.3. L’enregistreur de localisation des visiteurs (VLR)	(14)
I.3. Les réseaux de nouvelle génération de téléphonie sans fils réseau NGN.....	(15)
I.3.1. Architectures des réseaux NGN.....	(15)
I.3.1.1. La qualité de service (QoS : Quality of Service).....	(17)
I.4. Le WIMAX.....	(19)
I.4.1. Les différentes normes	(20)

I.4.2. Caractéristiques techniques.....	(22)
I.4.2.1. Couche physique.....	(22)

Chapitre II : Etude des modulations

II.1. Modulation d'amplitude	(23)
II.1.1. Description du signal AM.....	(23)
II.2. Modulation de fréquence (FM).....	(24)
II.2.1. Représentation temporelle du signal FM.....	(25)
II.3. Modulation de phase	(25)
II.4. Les modulations numérique.....	(26)
II.4.1. La modulation PSK	(26)
II.4.1.1. La modulation QPSK.....	(26)
II.4.1.2. La modulation BPSK.....	(28)
II.4.2. Modulation en quadrature QAM	(29)
II.4.2.1. Exemple de codage d'une information binaire avec QAM	(29)
II.5. Présentation de l'OFDM	(30)
II.5.1. Notion d'orthogonalité	(30)
II.5.2. Schéma bloc d'un système OFDM.....	(32)
II.6. Les canaux des trajets multiples	(33)
II.6.1. Problèmes des trajets-multiples	(33)
II.7. Méthode de multiplexes.....	(34)
II.7.1. FDMA (Frequency Division Multiple Access)	(35)
II.7.2. TDMA (Time Division Multiple Access)	(36)
II.7.3. CDMA (Code Division Multiple Access)	(37)

Chapitre III : Simulation OFDM sous MATLAB

III.1. Schéma de simulation d'un système de modulation OFDM.....	(38)
III.2. Paramètre de simulation du système OFDM.....	(39)
III.3. Générateur binaire de Bernoulli (simulation de données).....	(39)

III.4. Emetteur	(40)
III.4.1. Modulateur adaptatif (Modulation Bank)	(40)
III.4.2. Insertion de fréquences pilotes	(43)
III.4.3. Emetteur OFDM.....	(44)
III.5. Canal de transmission.....	(45)
III.6. Récepteur.....	(46)
III.6.1. Récepteur OFDM	(46)
III.6.2. Démodulateur (Frequency Error Correction Bank)	(47)
III.7. Calcul du taux d'erreur (BER)	(49)
III.8. Système d'évaluation des performances	(50)
III.8.1. Données Simulées	(50)
III.8.2. Résultats de simulation sans trajets multiples	(52)
III.8.3. Résultats de simulation avec trajets multiples.....	(54)
III.8.4. Discussion.....	(56)
CONCLUSION	(57)
ANNEXES	
BIBLIOGRAPHIE	

Introduction

Le terme télécommunication signifie communiquer à distance, le but de télécommunication est donc de transmettre un signal porteur d'une information (voix, musique, image...) d'un lieu à un autre situés l'un par rapport à l'autre à une certaine distance.

Un réseau de télécommunication peut être vu comme un ensemble de ressources matérielles mises en place pour offrir un ensemble de services. Dans les dernières années, l'évolution technologique a permis l'augmentation de la capacité des fonctionnalités des ressources et le nombre d'informations transmises. Ainsi, par exemple le succès des services de l'internet a engendré une explosion de trafic qui a mené les opérateurs à utiliser de nouvelles technologies dans le cœur des réseaux telles que IP, NGN qui semblent bien adaptés pour la mise en place voix/données.

Les nouvelles générations de réseaux mobiles tels que le GPRS, l'UMTS permettent de faire des progrès dans le nombre de services et le débit à mettre à la disposition des utilisateurs. La mise en place de réseaux multiservices devrait permettre également de réduire le coût de déploiement des différents services et le coût d'exploitation et de maintenance.

Les communications sans fil font l'objet d'étude d'un grand nombre de chercheurs, ceci dans le but d'obtenir une bonne qualité de communication sans fil avec des débits de transmission élevés et de libérer les usagers des câbles afin de leur permettre de se déplacer tout en restant connectés. Dans cette optique, le Wifi est développé et plus tard le WiMAX. Dans ce type de système de communication il faut combattre adéquatement l'effet des trajets multiples plus précisément l'évanouissement sélectif en fréquence. Ceci a mené au développement et à l'utilisation de la technique du multiplexage par division de fréquence orthogonale (OFDM) qui semble être une solution très prometteuse.

Dans la présente étude nous nous intéressons au système de télécommunication d'une façon générale et à la modulation OFDM d'une façon particulière. Comme ce type de modulation sera utilisé dans les prochaines générations des systèmes de télécommunication sans fil nous nous proposons de faire une étude sur le système OFDM pour comprendre son fonctionnement et ses performances. Pour ce faire, nous nous appuyons sur le toolbox du

Simulink de MATLAB pour l'étude et simulation de la norme IEEE 802.16 représentant la couche physique d'une transmission OFDM.

Ce travail est présenté en trois chapitres. Le premier traite des systèmes de télécommunications d'une façon générale. Le deuxième illustre les différents types de modulations du signal utilisées dans les systèmes de transmission. Le dernier chapitre est consacré à l'étude et simulation de la couche physique du système OFDM.

I. Les systèmes de Télécommunication

I.1. Architecture générale du réseau téléphonique :

Le Réseau Téléphonique Commuté met en relation deux postes d'abonné. L'échange d'informations nécessaires à l'établissement, au maintien et à la rupture de la relation s'appelle la signalisation.

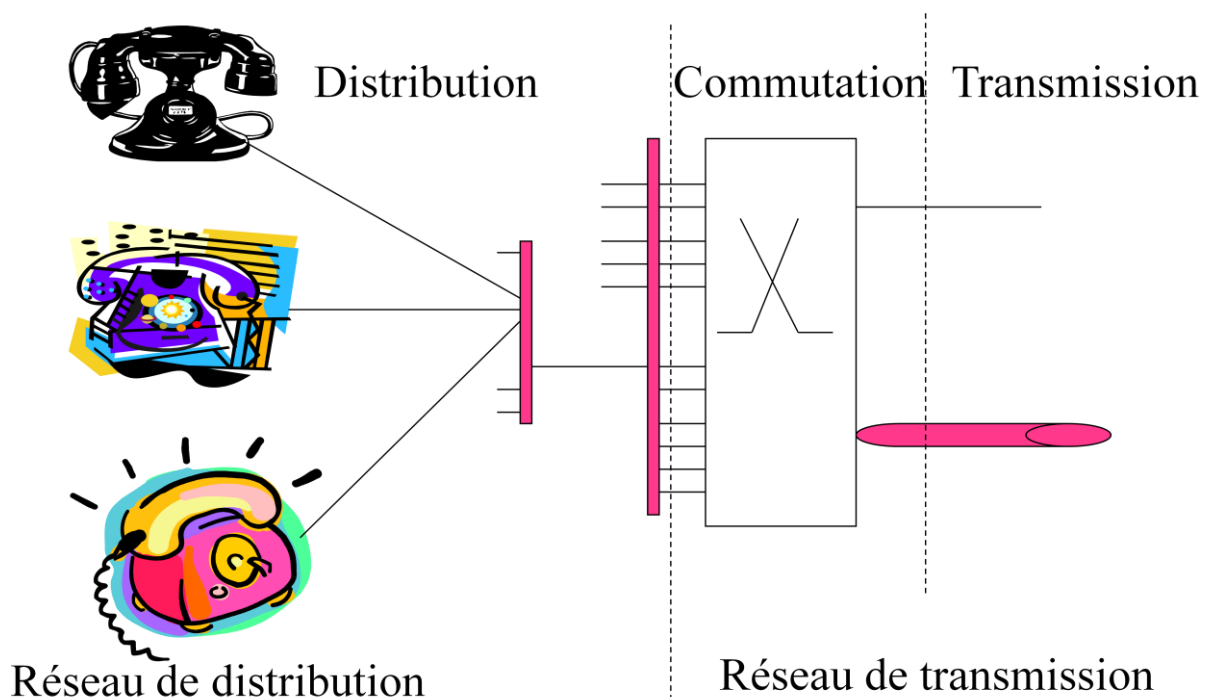


Fig. I.1. Schéma général du réseau téléphonique.

Le RTC est organisé en trois sous-parties :

- **La commutation** : partie centrale du réseau. Elle permet de réaliser la mise en relation entre les abonnés.
- **La transmission** : ensemble des techniques mises en œuvre pour relier les commutateurs entre eux. L'ensemble des commutateurs et des supports de transmissions entre commutateurs est appelé réseau de transmission ou réseau de transport.
- **La distribution** : organisation technique mise en œuvre pour relier les abonnés au commutateur le plus proche (commutateur de rattachement). L'ensemble des dispositifs permettant cette liaison est le réseau de distribution

I.1.1. La commutation

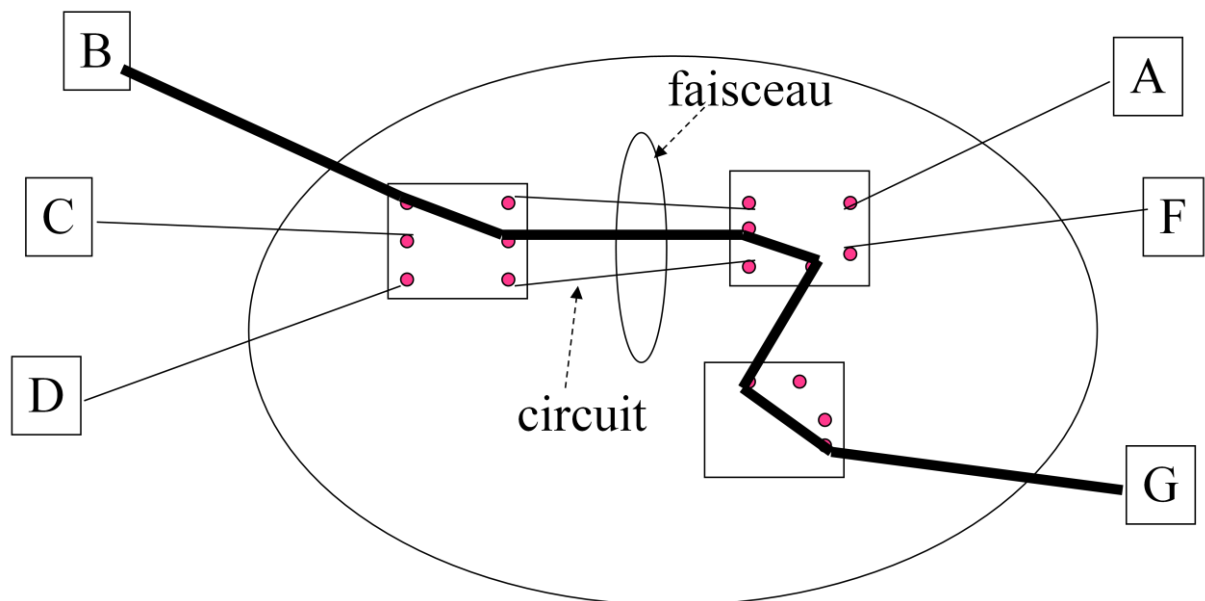


Fig. I.2. Principe de la commutation des circuits.

Faisceau : ensemble des circuits passant par un même chemin.

Pour assurer un bon fonctionnement du réseau en cas de rupture d'un support conduit à dupliquer les faisceaux entre les commutateurs même lorsque la charge à écouler ne le justifie pas.

I.1.1.1. Organisation du réseau

Le réseau téléphonique est organisé en 3 niveaux (zones):

- Zone à Autonomie d'Acheminement (ZAA): au bas de la hiérarchie, les commutateurs (CAA) accueillent les abonnés et peuvent établir différents types de communications.
- Zone de Transit Secondaire (ZTS): comporte les commutateurs (CTS). Les abonnés ne sont pas reliés aux CTS. Ils assurent les brassages des circuits lorsqu'un CAA ne peut atteindre le destinataire directement.
 - Zone de Transit Principale (ZTP): cette zone assure la commutation des liaisons longues distances. L'un des commutateurs (CTP) est relié au commutateur international de transit (CTI).
- Dans les zones à faible densité, les abonnés sont rattachés à des commutateurs locaux (CL): concentrateur de trafic.

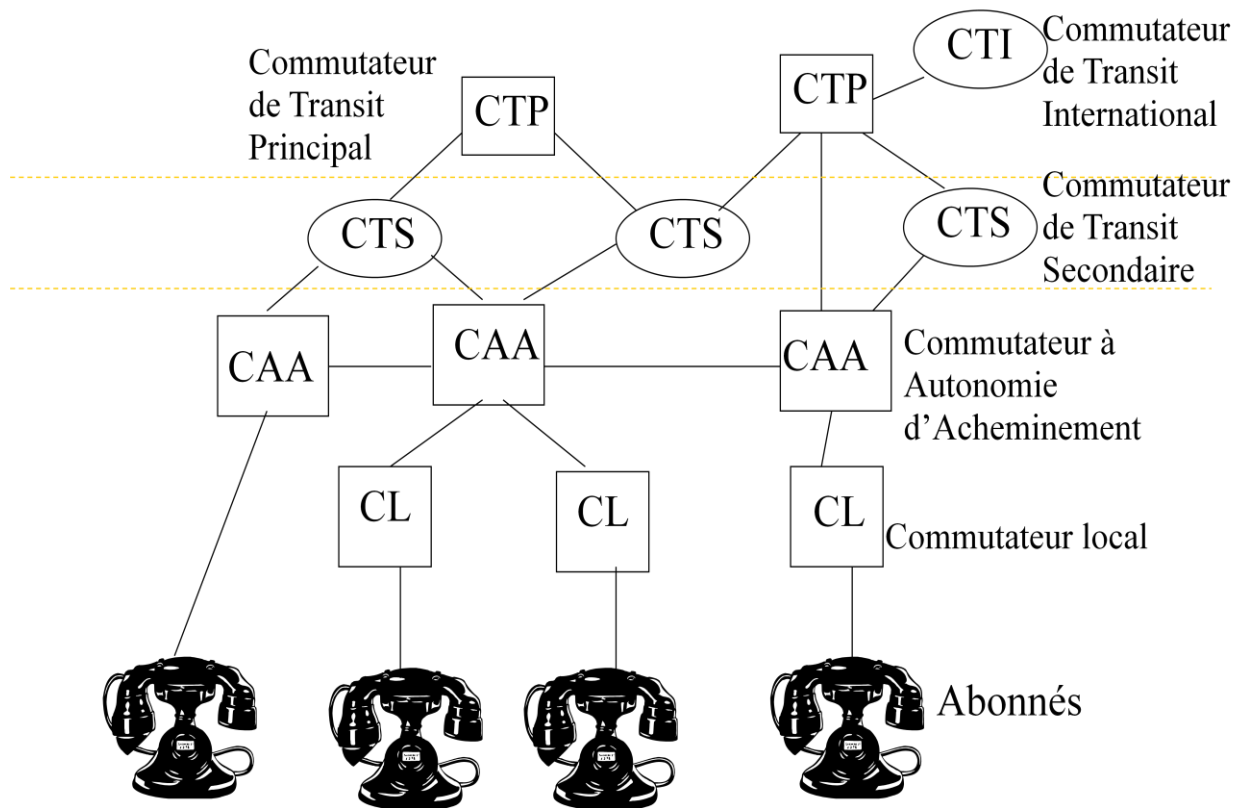


Fig. I.3. Hiérarchie de réseau téléphonique.

I.1.1.2. Règles d'acheminement

Le réseau étant partiellement maillé, plusieurs itinéraires sont généralement possibles pour atteindre un abonné. Pour un numéro donné, le faisceau de premier choix est choisi de telle manière qu'il conduise l'appel vers le commutateur le plus proche de l'abonné appelé en empruntant les faisceaux de plus faible hiérarchie.

I.1.1.3. Fonction principale d'un commutateur

Sa principale fonction est la connexion c'est-à-dire la liaison temporaire entre 2 jonctions.
(Une jonction est un circuit ou une ligne d'abonné).

- L'établissement des connexions nécessite l'échange de signalisation entre les commutateurs.
- C'est assuré par l'unité de commande (ordinateur).
- Les Unités de Raccordement d'Abonnés (URA).
- Fournissent l'énergie à l'alimentation des postes téléphoniques, adaptent les caractéristiques électriques.
- Détectent le décrocher et le raccroché d'un poste
- Génèrent une sonnerie vers un poste et exécutent des tests des lignes d'abonnés.

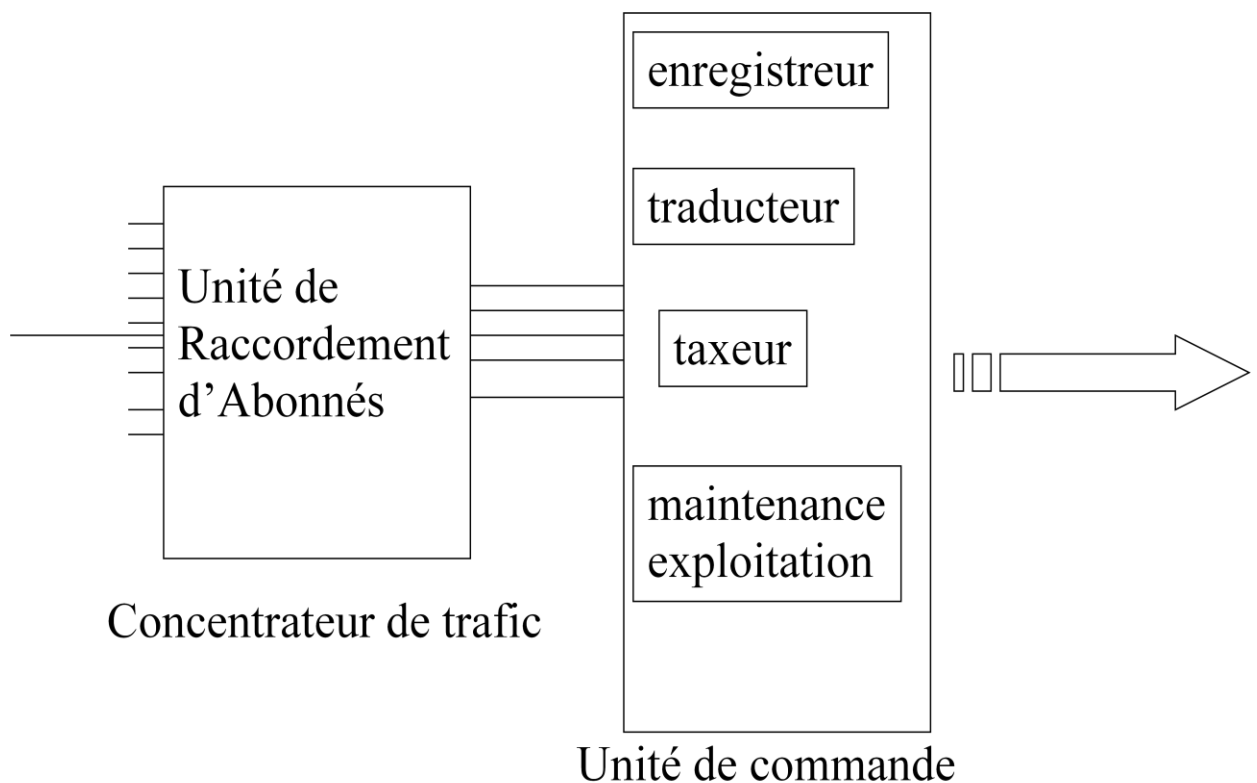


Fig. I.4. Architecture d'un commutateur.

I.1.2. Le réseau de transmission

Les commutateurs du réseau téléphonique (CAA, CTS, CTP) sont reliés entre eux par des supports de transmission de très grande capacité qui sont constitués de voies numériques multiplexées.

Le multiplexage est basé sur la technique MIC (Modulation par impulsion et Codage) qui regroupe 32 voies à 64 Kbit/s (en Europe) soit un débit global de 2048 Mbit/s.

Une hiérarchie de multiplexage est définie dans le réseau de transmission afin de regrouper un très grand nombre de voies sur des supports large bande.

I.1.2.1. Hiérarchie numérique plésiochrone

PDH Cette technique suppose plusieurs niveaux de multiplexage en cascade.

Avec cette technique PDH (Plesiochrone Digital Hierarchy), chaque multiplexeur est indépendant des autres (plusieurs horloges).

Cette technique utilise des systèmes de transmissions allant jusqu'à 140 Mbit/s sur des faisceaux hertziens, des câbles coaxiaux, dans des liaisons sous-marines ou par satellite.

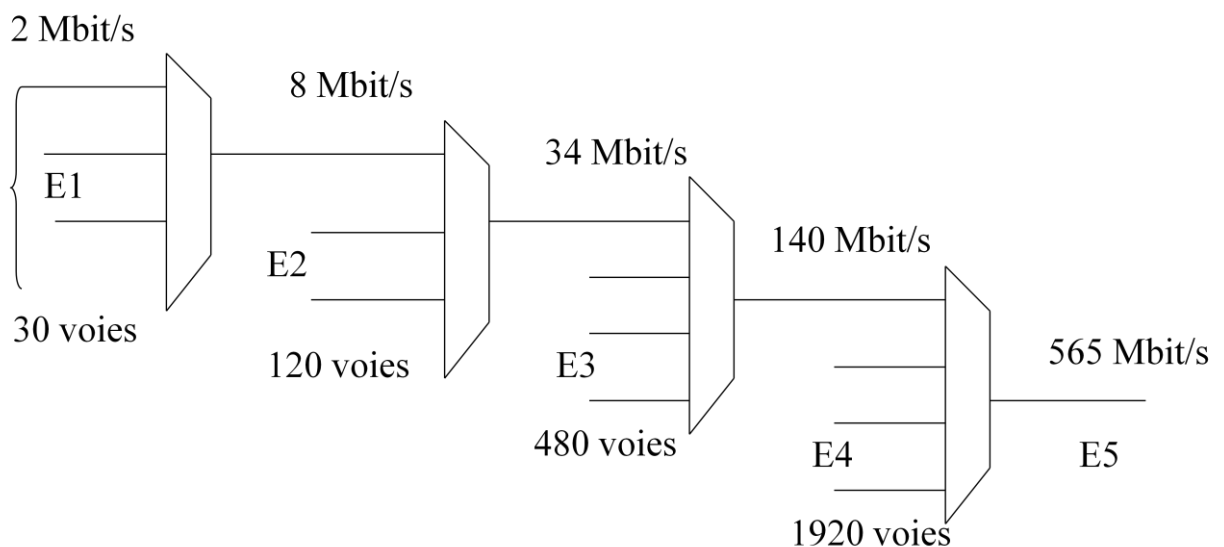


Fig. I.5. Débit de sortie = au moins 4 X débits d'entrée.

I.1.3. Le réseau de distribution

Il est souvent appelé boucle locale.

Le poste d'abonné est relié par une paire de fils de cuivre à un commutateur : la ligne d'abonné.

Le réseau de distribution a pour but de rassembler dans un même câble le maximum de paires afin de minimiser les travaux de génie civil.

La nécessité d'amener une paire de bout en bout chez l'abonné rend le coût du réseau de distribution important. C'est pourquoi d'autres solutions, telles qu'une liaison radio entre l'abonné et le central téléphonique sont envisagées. C'est la boucle locale radio ou Wireless Local Loop (WLL).

I.2. Le GSM

Le GSM (Global System for Mobile communications), est un système cellulaire et numérique de télécommunication mobile. Il a été rapidement accepté et a vite gagné des parts de marché telles qu'aujourd'hui plus de 180 pays ont adopté cette norme et plus d'un milliard d'utilisateurs sont équipés d'une solution GSM. L'utilisation du numérique pour transmettre les données permettent, des services élaborés, par rapport à tout ce qui a existé. On peut citer, par exemple, la possibilité de téléphoner depuis n'importe quel réseau GSM dans le monde. Les services avancés et l'architecture du GSM ont fait de lui un modèle pour la troisième génération de systèmes cellulaires, le réseau UMTS. Ce qui suit donne une vue globale de l'architecture du réseau, des liaisons radio, et du fonctionnement du réseau.

I.2.1. Les objectifs de GSM

- Offrir un vaste éventail de services de télécommunications compatibles avec ceux des réseaux fixes.
- Offrir des services spécifiques dus à la mobilité des usagers.
- Assurer la compatibilité d'accès à n'importe quel utilisateur dans n'importe quel pays exploitant le système GSM.
- Assurer la localisation automatique des mobiles sous la couverture globale de l'ensemble des réseaux.
- Permettre une grande variété de terminaux mobiles.
- Obtenir une bonne efficacité spectrale.
- Obtenir des coûts permettant d'assurer le succès du service.

I.2.2. Architecture de GSM

Rôle d'un réseau de radiotéléphonie : permettre des communications entre abonnés mobiles et abonnés du RTC caractérisées par un accès très spécifique, la liaison radio. Il est constitué de trois sous-ensembles :

- sous-système radio (BSS : Base Station Subsystem) Transmission radioélectrique
Gestion de la ressource radio.
- sous-système d'acheminement - réseau fixe (NSS : Network Subsystem)
établissement des appels mobilité.
- sous-système d'exploitation et de maintenance admission sur réseau.

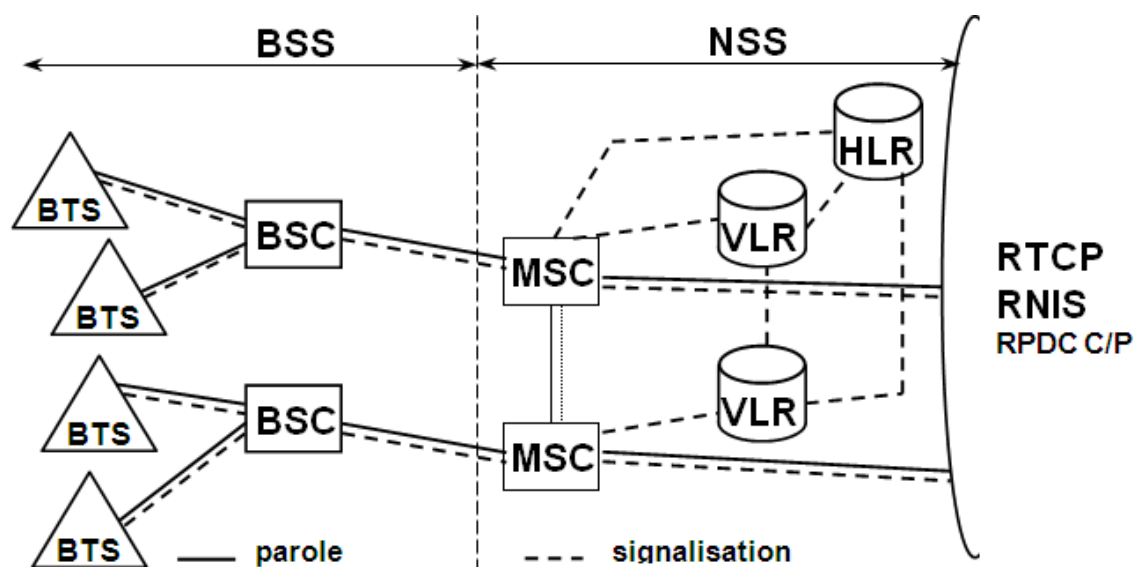


Fig. I.6. Architecture de GSM.

I.2.2.1. Le sous système radio

Le sous-système radio gère la transmission radio. Il est constitué de plusieurs entités dont le mobile (MS), la station de base (BTS, Base Transceiver station) et un contrôleur de station de base (BSC, Base station Controller).

I.2.2.1.1. station de base (BTS)

La station de base est l'élément central, que l'on pourrait définir comme un ensemble émetteur/récepteur pilotant une ou plusieurs. Dans le réseau GSM, chaque cellule principale au centre de laquelle se situe une station base peut-être divisée, grâce à des antennes directionnelles, en plus petites cellules qui sont des portions de celle de départ et qui utilisent des fréquences porteuses différentes cellules.

C'est la station de base qui fait le relais entre le mobile et le sous-système réseau. Comme le multiplexage temporel est limité à 8 intervalles de temps, une station de base peut gérer tout

au plus huit connections simultanées par cellule. Elle réalise les fonctions de couche physique et de la couche liaison de données.

En cas de besoin, on peut exploiter une station de base localement ou par télécommande à travers son contrôleur de station de base.



Fig. I.7. Photo d'une BTS.

I.2.2.1.2. Le contrôleur de stations de base BSC

Le contrôleur de station de base gère une ou plusieurs stations de base et communique avec elles par le biais de l'interface A-bis. Ce contrôleur remplit différentes fonctions tant au niveau communication qu'au niveau exploitation.

Pour les fonctions de communications, le BSC agit vis-à-vis du trafic abonné venant des stations de base comme un concentrateur puisqu'il véhicule les communications provenant des différentes stations de base. Dans l'autre sens, le contrôleur commute les données en les dirigeants vers la bonne station de base.

Il remplit à la fois le rôle de relais pour les différents signaux d'alarme destinés au centre d'exploitation et de maintenance et de banque de données des données installées sur les stations de base .

Une autre fonctionnalité importante est la gestion des ressources radio pour la zone couverte par les différentes stations de base qui y sont connectées. En effet, le contrôleur gère les transferts intercellulaires des utilisateurs dans sa zone de couverture, c'est-à-dire quand une station mobile passe d'une cellule dans une autre. Il doit alors communiquer avec les stations de base qui va prendre en charge l'abonné et lui communiquer les informations nécessaires tout en avertissant la base de données locale VLR (Visitor Location Register) de la nouvelle localisation de l'abonné.

I.2.2.2. Le sous système réseau

Le sous-système réseau, appelé Network Switching Center (NSS), joue un rôle essentiel dans un réseau mobile. Alors que le sous-réseau radio, les éléments du NSS permanent en charge toutes les fonctions de contrôle et d'analyse d'informations contenues dans des bases de données nécessaires à l'établissement de connexion. Le NSS est constitué de :

- Mobile Switching Center (MSC).
- Home Location Register (HLR)
- Visitor Location Register (VLR).

I.2.2.2.1. Le centre de commutation mobile(MSC)

Le centre de commutation mobile est relié au sous-système radio via l'interface A.

Son rôle principal est d'assurer la commutation entre les abonnés du réseau mobile et ceux du réseau commuté public(RTC).

De plus, il participe à la fourniture des différents services aux abonnés tels que la téléphonie, les services supplémentaires et services de messagerie. Il permet encore de mettre à jour les différentes bases de données (HLR, VLR) qui donnent toutes les informations concernant les abonnés et leurs localisations dans le réseau.

Les commutateurs MSC d'un opérateur sont reliés entre eux pour la commutation interne des informations. Des MSC servant de passerelle (Gateway Mobile Switching Center, GMSC) sont placées en périphérie du réseau d'un opérateur de manière à assurer une interopérabilité entre réseaux d'opérateurs.



Fig. I.8. Photo d'un MSC.

1.2.2.2.2. L'enregistreur de localisation nominale(HLR)

Il existe au moins un enregistreur de localisation (HLR) par réseau (PLMN).il faut le voir comme une base de données avec des informations essentielles avec un temps d'accès doit être réduit au stricte minimum. Plus la réponse du HLR est rapide et plus le temps d'établissement de la connexion sera petit.

Le HLR contient à la fois :

- Toutes les informations relatives aux abonnés
- Ainsi qu'un certain nombre de données dynamiques telles que la position de l'abonné dans le réseau-en fait, son VLR-et l'état de son terminal (allumé, éteint, en communications, libres,...).

Les données dynamiques sont mises à jour par le MSC. Cette base de données est souvent unique pour un réseau GSM et seules quelques personnes y ont accès directement.

I.2.2.2.3. L'enregistreur de localisation des visiteurs (VLR)

Cette base de données ne contient que des informations dynamiques et liée à un MSC.il y en a donc plusieurs dans un réseau GSM. Elle contient des données dynamiques qui lui sont transmises par le HLR avec lequel elle rentre en communication lorsqu'un abonné entre dans la zone de couverture du centre de communication mobile auquel elle est rattachée. Lorsque l'abonné quitte cette zone de couverture, ses données sont transmises à un autre VLR et suivent l'abonné.

I.3. Les réseaux de nouvelle génération de téléphonie sans fils réseau NGN

Actuellement, l'ensemble des architectures de communication est en train d'évoluer vers une infrastructure globale fondée sur IP : les réseaux de nouvelle génération (NGN : Next Generation Networks). Le modèle IP est caractérisé par une structure modulaire composée de plusieurs couches distinctes qui communiquent à travers des interfaces bien définies. L'évolution vers les réseaux NGN s'explique par la capacité du modèle IP d'offrir un mode d'acheminement des données indépendant, d'une part, du type de technologies réseaux sous-jacentes (Ethernet, Wi-Fi, WiMax, Satellite, ...etc.) et d'autre part, du type de données véhiculées (audio, vidéo, données). L'objectif est ainsi de réaliser à travers les réseaux NGN, le support de multiples services (téléphonie, télévision, services Internet) au sein d'une unique infrastructure tirant parti de l'hétérogénéité des technologies d'accès.

L'architecture des réseaux NGN a donc pour objectif de permettre à des utilisateurs utilisant différents terminaux d'accéder à plusieurs types de services via des réseaux d'accès variés. Dans ce contexte, l'un des défis majeurs est de gérer les besoins des différents utilisateurs en termes de qualité de service (QoS : Quality of Service), de sécurité des services fournis, mais aussi de mobilité des utilisateurs.

I.3.1. Architectures des réseaux NGN

Un cœur de réseau unique et mutualisé pour tous les types de réseaux d'accès et de services une architecture de cœur de réseau en 2 couches : « Transport » / « Services », Une évolution du transfert des données vers le mode paquet.

Des interfaces ouvertes et standardisées entre chaque couche afin de réaliser l'indépendance des services vis-à-vis du réseau.

Un découplage entre la fourniture de service et la fourniture de réseau. Le support de technologies d'accès multiples, Le support de la convergence des réseaux voix/données et fixe/mobile, Le support de terminaux multiples (modulaires, multi-mode, multimédia et adaptatifs).

Ainsi, la principale caractéristique d'un réseau de nouvelle génération est son fondement sur IP qui offre un mode de transfert homogène de bout en bout indépendant, d'une part, des réseaux sous-jacents et, d'autre part, du type de données applicatives véhiculées. Après l'évolution du cœur de réseau vers le « tout-IP », la notion la plus importante reste la décomposition en plans fonctionnels séparés par des interfaces ouvertes qui assure à la fois le passage à l'échelle et la flexibilité d'une telle architecture en offrant une facilité d'interconnexion et d'intégration de nouveaux services. La Figure ci-dessous offre une représentation communément admise de l'architecture des réseaux NGN.

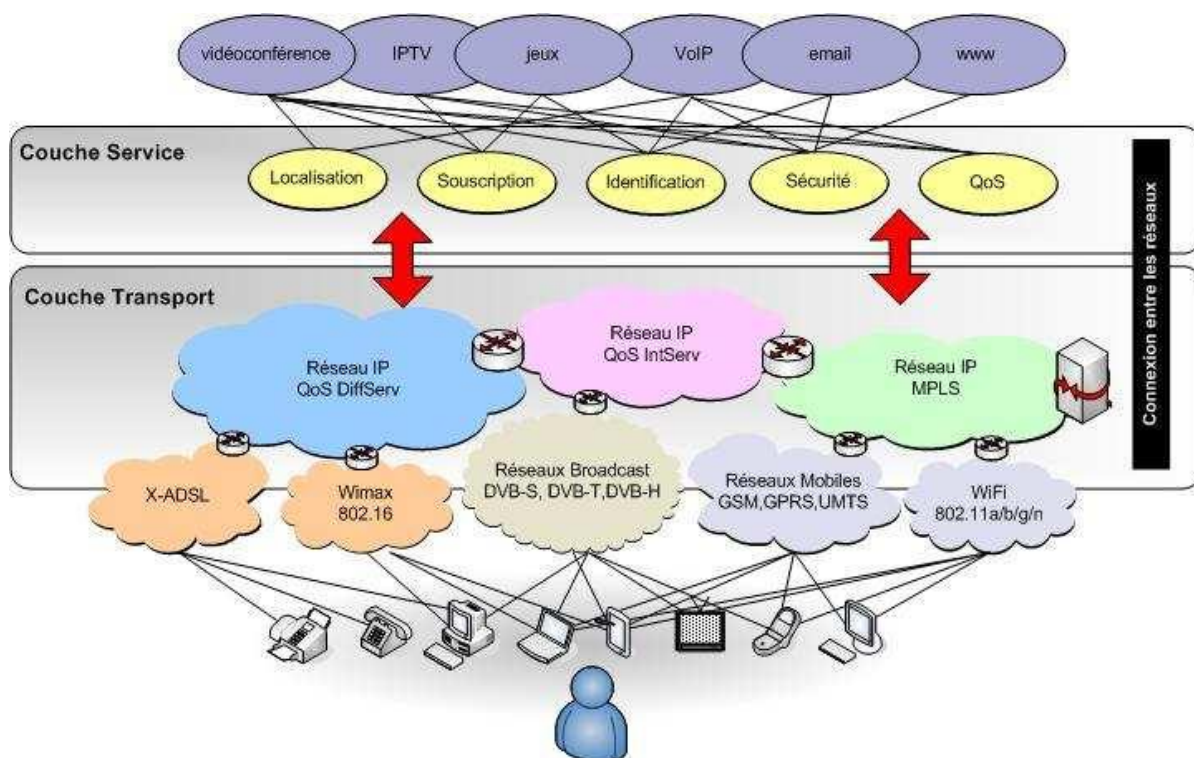


Fig. I.9. Architecture d'un réseau NGN

Le plan de « Transport » regroupe l'ensemble des ressources mises en place pour assurer le transfert de données. Ainsi, il gère l'acheminement du trafic vers sa destination en fournissant une connectivité IP aux différents composants d'un réseau NGN tout en garantissant une QoS de bout en bout. Ce plan dépend directement de la technologie du réseau de transport utilisé pour acheminer les paquets.

Le plan de « Service » fournit les fonctionnalités de base pour l'exécution des services avec ou sans session. En effet, il regroupe les plateformes d'exécution de service et de diffusion de contenu tout en masquant la diversité technologique aux clients et aux fournisseurs de services.

Après ces plans horizontaux, on peut différencier les réseaux d'accès des réseaux de transit. En effet, les premiers ont pour but de concentrer le trafic des différents utilisateurs vers des équipements centraux. Alors que, les deuxièmes ont pour vocation d'acheminer des volumes de trafic importants entre quelques entités limitées. Dans la Figure ci-dessus.

nous pouvons noter la diversité des technologies permettant à l'utilisateur d'accéder aux services ; ce qui peut former ce qu'on appelle parfois le plan d'accès. Il est important de noter que l'interaction entre les réseaux de nouvelle génération est les réseaux traditionnels est prise en charge par différentes passerelles d'interconnexion afin d'assurer une compatibilité avec les technologies déployées actuellement.

I.3.1.1. La qualité de service (QoS : Quality of Service)

Dans une architecture NGN, le transport des services de nouveaux types (multimédia, conversationnels, temps réel, etc.) est basé sur le protocole IP (Internet Protocol). Cependant, IP souffre d'un inconvénient qui réside dans le routage de toutes les données d'une manière équivalente. En effet, le protocole IP fournit un acheminement au mieux (Best-Effort) qui n'offre aucun contrôle, ni garantie, sur les conditions de transport de bout en bout des paquets IP. Grâce à sa simplicité, ce service a rencontré un grand succès dans le passé pour le transport des données asynchrones. Cependant, ce succès est de plus en plus contesté, actuellement, par la multiplication des flux synchrones, ou temps réel, qui exigent une certaine QoS pour leur transport.

Etant donné que ces nouveaux services, essentiellement les services multimédia et les services temps réel, répondent à des exigences très strictes en termes de délai de transport et de bande passante, il devient primordial de gérer l'offre de QoS dans les réseaux fondés sur IP tels que les réseaux NGN.

La qualité de service (QoS) est un concept incontournable dans le monde des télécommunications, qui a pour but de donner la possibilité à un utilisateur de disposer d'une qualité de transmission supérieure à celle d'un utilisateur quelconque. Bien que complexe, le concept de la QoS n'a rien de révolutionnaire puisqu'il se fonde sur des technologies préexistantes qu'il vise à rationaliser et souvent à simplifier afin d'en faciliter la mise en œuvre.

Le nouveau cadre de travail introduit par les réseaux NGN est caractérisé par une segmentation des tâches entre différents opérateurs et fournisseurs (services et réseaux) à travers différentes couches verticales (terminal utilisateur, réseau d'accès et réseau de transit) et horizontales (« Transport » / « Service »). Cette segmentation permet

d'envisager des solutions de QoS de bout en bout résultant de l'agrégation des solutions déjà proposées par la communauté scientifique pour l'Internet.

C'est pourquoi, avant de décrire une vision de l'offre de QoS dans la cadre de l'architecture NGN, nous allons dans un premier temps rappeler les principales architectures de QoS développées par l'IETF.

I.4. Le WIMAX

Le Wifi se développe énormément et devient de plus en plus accessible financièrement. Mais le Wifi est limité par sa portée et ne peut donc être utilisé que pour les réseaux locaux (WLAN : Wireless Local Area Network). Le Wimax est une technologie complémentaire du Wifi, faisant partie des WMAN (Wireless Metropolitan Area Network). Wimax, qui signifie « World Interoperability for Microwave Access », est une norme technique basée sur le standard de transmission radio 802.16, validé en 2001 par l'organisme international de normalisation IEEE. Elle est activement développée par le consortium Wimax Forum, qui rassemble une soixantaine d'industriels, tels que Intel, Fujitsu, Nokia et bien d'autres.

Le WiMAX, première norme pour les équipements de la boucle locale radio relance la BLR qui était un peu à bout de souffle.

Etendu du WiMAX par rapport aux autres normes :

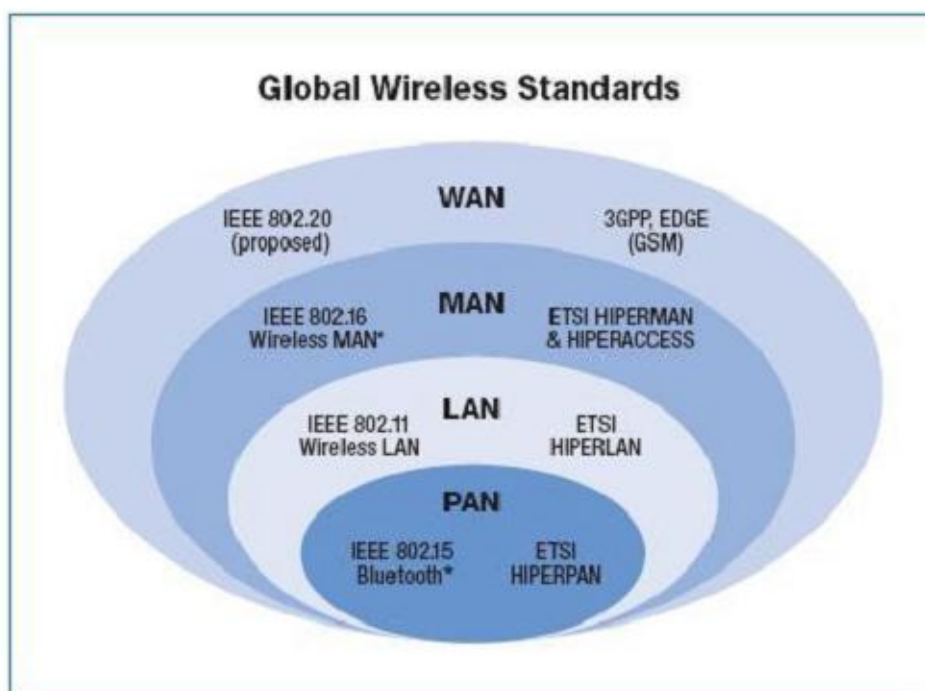


Fig. I.10. Différent norme de réseaux de communication.

I.4.1. Les différentes normes

Il existe plusieurs axes de travail au sein de l'IEEE 802.16

- IEEE std 802.16 définit des réseaux métropolitains sans fil utilisant des fréquences supérieures à 10Ghz. Il est homologué comme standard IEEE 802 depuis octobre 2002.
- IEEE std 802.16a est un amendement standard 802.16 pour les fréquences entre 2 et 11Ghz. Il est homologué comme standard IEEE 802 depuis le 09 octobre 2003
- IEEE P802.16b a travaillé sur des solutions très proches mais dans Les bandes de fréquences sans licence. Il a été fusionné avec 802.16a.
- IEEE std 802.16c définit les profils (les choix d'option) pour les réseaux utilisant les fréquences entre 10 et 66GHZ. Il est homologué comme standard IEEE 802.16 depuis juillet 2003.

- IEEE std 802.16d a été converti en projet des révisions des standards de base (802.16, 802.16a et 802.16c) il porte maintenant le nom de IEEE 802.16-REVd.
- IEEE std 802.16.2s'occupe de l'interopérabilité entre toute les solutions 802.16.

Une nouvelle version vient d'être publiée en mars 2004.

En juin 2004 toutes ces normes ont été rassemblées en une seule : 802.16-2004.

Voici les caractéristiques des différentes normes pour le WiMAX :

Normes	802.16	802.16a	802.16e
Spectre	11-66GHz	2-11GHz	<6 GHz
Débit	32 à 134 Mb/s	<= 70 ou 100 Mb/s	Jusqu'à 15Mb/s
Modulation	QPKS, QAM-16, QAM-64	OFDM avec 256 porteuses utilisant QPKS, QAM-16, QAM-64, QAM-256	Comme 802.16a
Mobilité	Fixe	Fixe	<= 100 km/h
Bande passante du canal	20, 25, 28 MHz	Sélectionnable Entre 1,25 à 20 MHz	5 MHz (prévisionnel)

Table.I.1. Caractéristique des différentes normes du WIMAX.

La norme 802.16^e est très attendu car elle rajoute ce qu'il manque au Wimax par rapport au Wifi : la mobilité ou autrement dit le « roaming », terme qui définit le fait de ne pas perdre le flux du réseau si l'on est mobile. C'est à dire que l'on pourra profiter d'un accès internet sans fil même en voyageant à plus de 100Km/h à condition de rester dans la zone de couverture. Il ya aussi la norme 802.20 qui est en phase de développement, elle permettra un accès rapide à internet par la BLR pour les terminaux mobiles (PDA, téléphone portables...)

I.4.2. Caractéristiques techniques

I.4.2.1. Couche physique

La norme 802.16 utilisait des fréquences comprises dans la bande 11-66 GHz. A l'origine, la conception des spécifications imposaient l'utilisation d'une propagation par vue directe (Line of Sight). C'est pourquoi la modulation d'une simple porteuse a été choisie et en raison de l'architecture point-multipoint la station de base transmet un signal TDM (Time Division Multiplex), avec une attribution d'un intervalle de temps (slot time) pour chacun.

La transmission des données qui s'effectue par burst a été choisie afin de pouvoir supporter en même temps le TDD (Time Division Duplex) et le FDD (Frequency Division duplex). Le profil de burst peut subir une adaptation dynamique de codage et de modulation en fonction des conditions d'émission-réception radio et quelque soit le type de duplexage choisi, ces paramètres peuvent être adaptés individuellement pour chaque station d'abonné.

II. Les modulations en Télécommunication

II.1. Modulation d'amplitude

Des techniques de modulations analogiques, la modulation d'amplitude fut la première employée. On la trouve fréquemment pour les transmissions hertziennes (stations radiophoniques grandes ondes par exemple).

II.1.1. Description du signal AM. Cas d'un signal sinusoïdal

Comme son nom l'indique un signal $s(t)$ (courant ou tension) modulé en amplitude est un signal constitué par une porteuse sinusoïdale de fréquence $F(p)$ dont l'amplitude $A(p)$ est modifié suivant une loi linéaire par le signal informatif $u(t)$. Si nous prenons le cas d'un signal modulant sinusoïdal, l'expression de $s(t)$ est donc :

$$s(t) = A_p \cos w_p t \quad (\text{II.1})$$

$$s(t) = A_p \cdot \cos(w_p t) + k \cdot A_m \cos(w_m t) \cdot \cos(w_p t) \quad (\text{II.2})$$

Où k est le facteur de proportionnalité du modulateur k est parfois appelé sensibilité du modulateur. La grandeur de k dépend des grandeurs de A_m et A_p .

$$\text{Soit : } s(t) = A_p \cdot \left(1 + \frac{k \cdot A_m}{A_p} \cdot \cos(w_m \cdot t)\right) \cos(w_p t) = A_p (1 + m \cdot u(t)) \cdot \cos(w_p t) \quad (\text{II.3})$$

Dans le cas général l'expression d'un signal modulé en amplitude est :

$$s(t) = A_p (1 + m \cdot u(t)) \cos(w_p t) \quad (\text{II.4})$$

Où m est un paramètre essentiel appelé taux de modulation. Il est d'usage d'exprimer m en %.

Amplitude

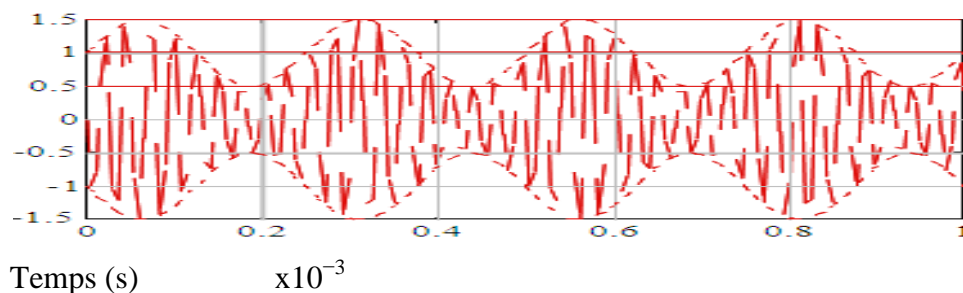


Fig.II.1. Représentation temporelle de signal modulé $s(t)$.

II.2. Modulation de fréquence (FM)

La modulation de fréquence ou FM très souvent employée en VHF (Very High Frequency) ou en UHF (Ultra High Frequency) pour la transmission des signaux audio, consiste à utiliser une porteuse dont la fréquence varie en fonction du signal modulant. Il est alors possible d'écrire :

$$F_s = F_p + F(t) \quad (\text{II.5})$$

Où $F(t)$ est une fréquence variant linéairement avec le signal modulant $u(t)$.

C'est à dire :

$$F(t) = v \cdot u(t) \quad (\text{II.6})$$

Avec v facteur de proportionnalité de modulateur (en Hz/Am) qui prend parfois la dénomination de sensibilité du modulateur.

La fréquence de la porteuse étant constante, la phase instantanée de $s(t)$ est donnée par :

$$\varphi_s(t) = \int (\omega_p + 2\pi \cdot f(t)) \cdot dt \quad (\text{II.7})$$

$$\varphi_s(t) = \omega_p t + 2\pi \cdot \int f(t) \cdot dt \quad (\text{II.8})$$

$$\text{Ou encore : } \varphi_s(t) = \omega_p t + 2 \cdot \pi \cdot v \cdot \int u(t) \cdot dt \quad (\text{II.9})$$

L'expression du signal $s(t)$ modulé en fréquence est alors :

$$s(t) = A_p \cos(\omega_p t + 2 \cdot \pi \cdot v \cdot \int u(t) \cdot dt) \quad (\text{II.10})$$

II.2.1. Représentation temporelle du signal FM

Prenons les cas d'un signal modulant sinusoïdal :

$$u(t) = A_m \cos(\omega_m t) \quad (\text{II.12})$$

avec $\omega_m \ll \omega_p$

L'expression du signal $s(t)$ est alors :

$$s(t) = A_p \cos(w_p t + 2\pi \cdot v \cdot \int A_m \cos(w_m t) \cdot dt) \quad (\text{II.13})$$

$$s(t) = A_p \cos(w_p t + \frac{2\pi \cdot v}{w_m} \cdot A_m \sin(w_m t)) \quad (\text{II.14})$$

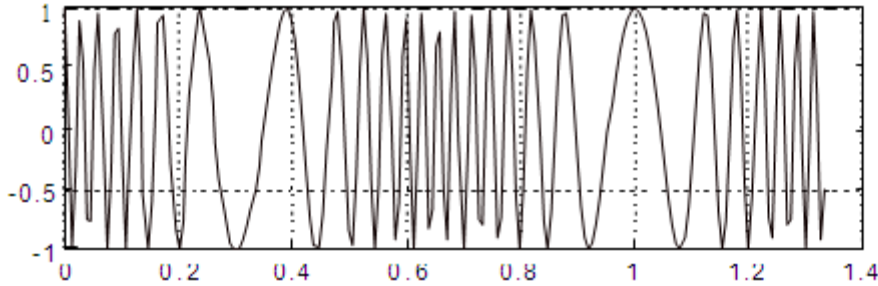


Fig. II.2. Représentation temporelle du signal modulé.

II.3. Modulation de phase

La modulation de phase ou MP est un mode de modulation consiste à transmettre un signal par la modulation de la phase d'un signal porteur (porteuse), cette modulation est non linéaire.

Signal porteur :

$$s(t) = A \cos(w_p t + \varphi(t)) \quad (\text{II.15})$$

La phase instantanée de porteuse est donnée par :

$$\Omega(t) = w_p t + \varphi(t) \quad (\text{II.16})$$

Soit un signal modulant : $m(t)$

Moduler en phase le signal revient à effectuer l'opération suivante :

$$\varphi(t) = K_p m(t) \quad (\text{II.17})$$

Le signal modulé s'écrit :

$$s(t) = A \cos(w_p t + K_p m(t)) \quad (\text{II.18})$$

II.4. Les modulations numériques

II.4.1. La modulation PSK

La modulation par déplacement de phase PSK (Phase Shift Keying) peut s'exprimer par la relation :

$$u(t) = U_0 \cos 2\pi \left(f_0 t + a_k \frac{1}{m} \right) \quad (\text{II.19})$$

Dans le cas général d'un signal modulant discret pouvant comporter m niveaux distincts. a_k peut alors prendre les valeurs de 0 à $m-1$. Pour le cas particulier d'un signal modulant binaire, on obtient :

$$u(t) = U_0 \cos(w_0 t + a_k \pi) = \pm U_0 \cdot \cos(w_0 t) \quad (\text{II.20})$$

On peut donc considérer cette modulation comme une modulation d'amplitude avec porteuse supprimée par un signal binaire d'amplitudes (+1, -1). Pratiquement, la modulation PSK n'est jamais utilisée en raison de la référence de phase absolue qu'elle implique. On préférera utiliser une modulation différentielle, qui code l'information binaire dans un écart de phase avec le moment précédent.

II.4.1.1. La modulation QPSK

QPSK (Quadrature Phase Shift Keying): techniques de modulation numérique procédant par déplacement de phase en quadrature. La modulation par déplacement d'amplitude en quadrature est économe en puissance, lorsqu'elle est à petits nombres d'états, et utilisée sur de longues distances (ex: transmissions par satellite). QPSK est une technique de modulation de phase, utilisée sur les réseaux câblés, très efficace sur le plan de la limitation du bruit

(perturbation du signal). QPSK est également utilisé pour le système mobile de la troisième génération UMTS.

Phase de signal $\varphi(t)$	Cos $\varphi(t)$ AXE I	Sin $\varphi(t)$ AXE Q
$-\frac{3\pi}{4}$	$\frac{-1}{\sqrt{2}}$	$\frac{-1}{\sqrt{2}}$
$-\frac{\pi}{4}$	$\frac{1}{\sqrt{2}}$	$\frac{-1}{\sqrt{2}}$
$\frac{\pi}{4}$	$\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$
$\frac{3\pi}{4}$	$\frac{-1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$

Table. II.1. Phase du signal QPSK.

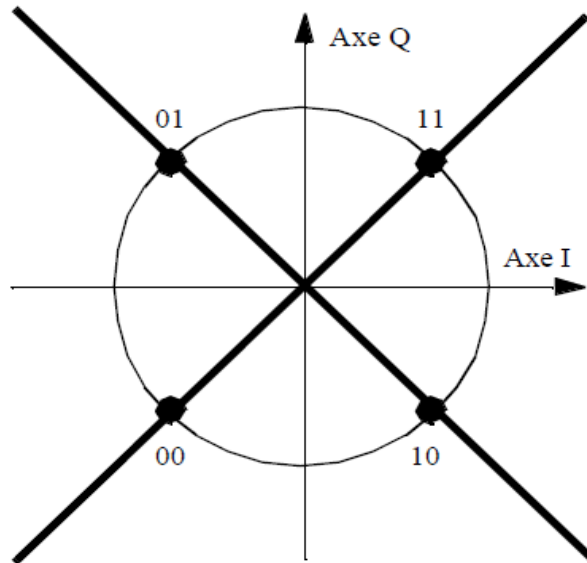


Fig. II.3. Diagramme de constellations pour QPSK.

II.4.1.2. La modulation BPSK

BPSK est la forme la plus simple du PSK. Elle utilise deux phases qui sont séparées de 180° , on l'appelle également 2-PSK. Cette modulation est la plus robuste de toutes les PSK car il faut une grande déformation du signal pour que le démodulateur se trompe sur le symbole reçu. Cependant on ne peut moduler qu'un seul bit par symbole (voir le schéma), ce qui est un inconvénient pour les applications qui nécessitent un débit binaire élevé.

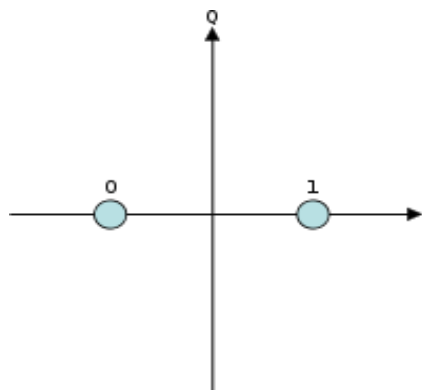


Fig. II.4. Diagramme de constellations pour BPSK.

II.4.2. Modulation en quadrature QAM

La modulation QAM poursuit le même but, mais dispose les points sur un treillis plutôt que sur un ensemble de cercles. Les modulations de type QAM sont actuellement les plus utilisées en technique de modem haute vitesse (UIT-T, Rec. V22bis). La modulation QAM utilise deux porteuses en quadrature (Inphase et Quadrature, I et Q) qu'elle combine par addition pondérée en fonction du signal binaire à retransmettre.

II.4.2.1. Exemple de codage d'une information binaire avec QAM

Soit à coder l'information 1100 en modulation QAM avec 4 bit par moment (16QAM). On utilise les deux premiers bits pour la porteuse I et les deux autres pour la porteuse Q selon une table de vérité arbitraire à priori. Le résultat de l'addition pondérée de I et de Q donne un vecteur résultant d'amplitude et de phases données.

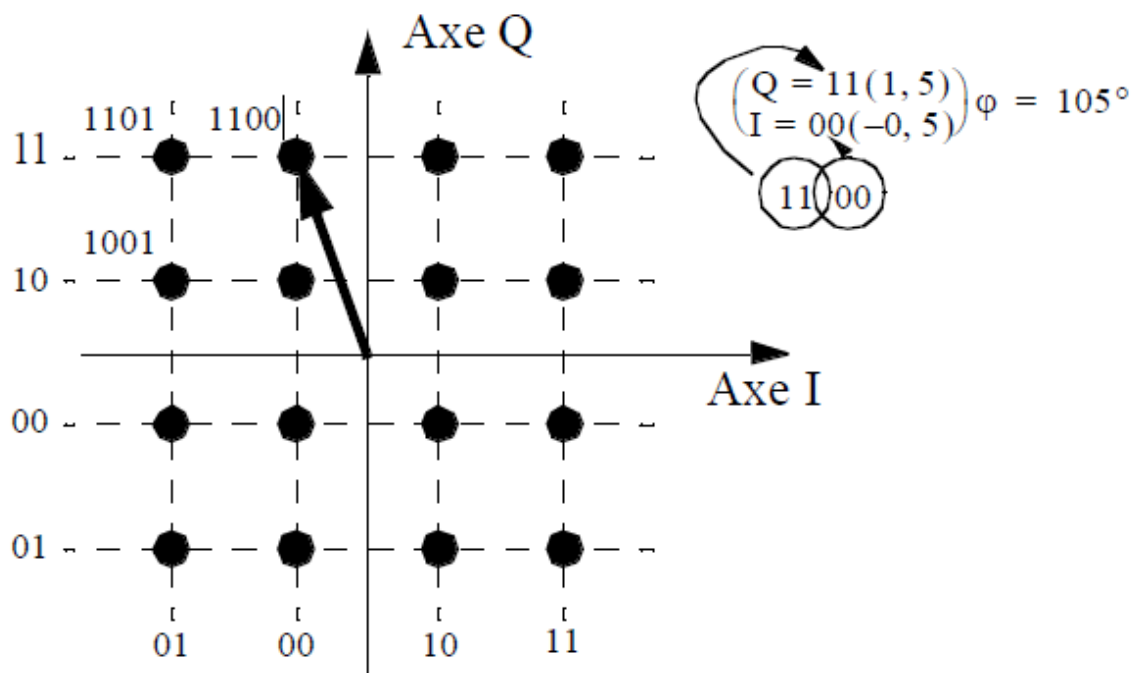


Fig. II.5. Diagramme de constellation pour QAM

II.5. Présentation de l'OFDM

Les origines de l'OFDM remontent à 1930 lorsque les célèbres laboratoires de recherche *Bell* ont développé une nouvelle technique d'accès pour le canal radio, qui succéda au FDM bien établi, et fut appelée OFDM (Orthogonal Frequency Division Multiplexing). L'intérêt pour cette technologie a connu un renouveau du milieu des années soixante.

Délaissé ensuite lors du développement de la théorie de l'égalisation pour les systèmes mono-porteuses (de moindre complexité), l'OFDM dû son retour en grâce, vers le milieu des années 1980, au projet de radiodiffusion numérique DAB (Digital Audio Broad casting). En effet, les systèmes mono-porteurs, contrairement à l'OFDM, ne remplissaient pas les conditions de résistance aux trajets multiples et de débit élevé pour un taux d'erreur binaire faible requis par cette nouvelle application. Depuis lors, l'OFDM est restée une technique prépondérante, puisqu'elle est utilisée pour de nombreuses applications comme la télévision numérique DVB (Digital Vidéo Broad casting) ou la norme ADSL (Assymetric Digital Suscriber Line) permettant des liaisons Internet à haut débit. Enfin l'OFDM s'adapte parfaitement aux communications mobiles, et semble incontournable pour les futurs standards de troisième et quatrième générations, où il est notamment question de l'associer au CDMA (Code Division Multiple Access) ou aux systèmes multi-antennes.

II.5.1. Notion d'orthogonalité

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

L'utilisation d'un très grand nombre de porteuses est une perspective très contraignante : il faut sûrement beaucoup de modulateurs / démodulateurs et de filtres. Il faut aussi davantage de largeur de bande. Il est heureusement simple de résoudre ces deux

problèmes en spécifiant un espacement rigoureusement régulier de $f_u = 1/T_u$ entre les sous-porteuses, où T_u est la période (utile ou active) du symbole pendant laquelle le récepteur intègre le signal démodulé. Les porteuses forment alors ce que les mathématiciens appellent un ensemble orthogonal.

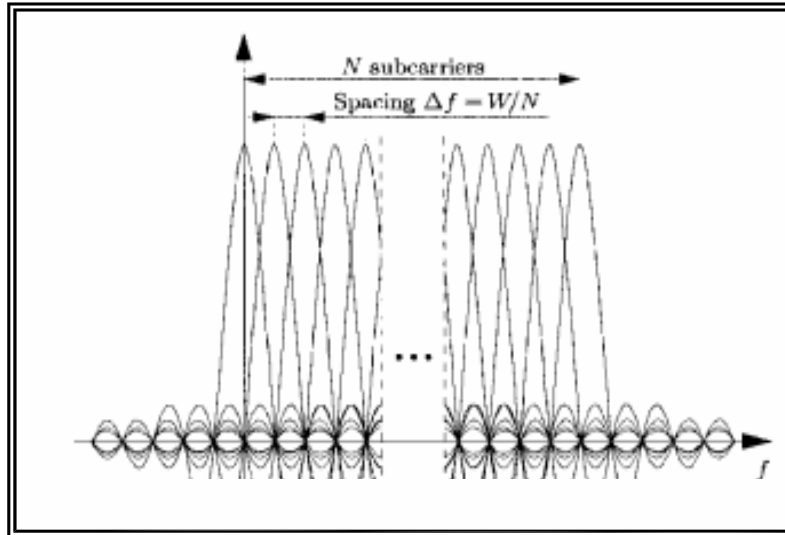


Fig. II.6. N sous-porteuses orthogonales pour un système OFDM.

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

$$\psi_k(t) = e^{jkw_u t} \quad (\text{II.21})$$

Avec $w_u = \frac{2\pi}{T_u}$

Les porteuses doivent satisfaire la condition d'orthogonalité

$$\int_{\tau}^{\tau+T} \psi_k(t) \psi_1^*(t) dt = 0, k \neq 1 \quad (\text{II.22})$$

$$= T_u, k = 1$$

Donc, cette contrainte est une condition d'orthogonalité pour les fonctions $\Psi_k(t)$, et les fonctions $\Psi_k(t)$ forment une base orthogonale de l'espace temps-fréquence, ce qui permet de retrouver facilement les symboles et autorise donc un recouvrement spectral sans perte de l'information.

II.5.2. Schéma bloc d'un système OFDM

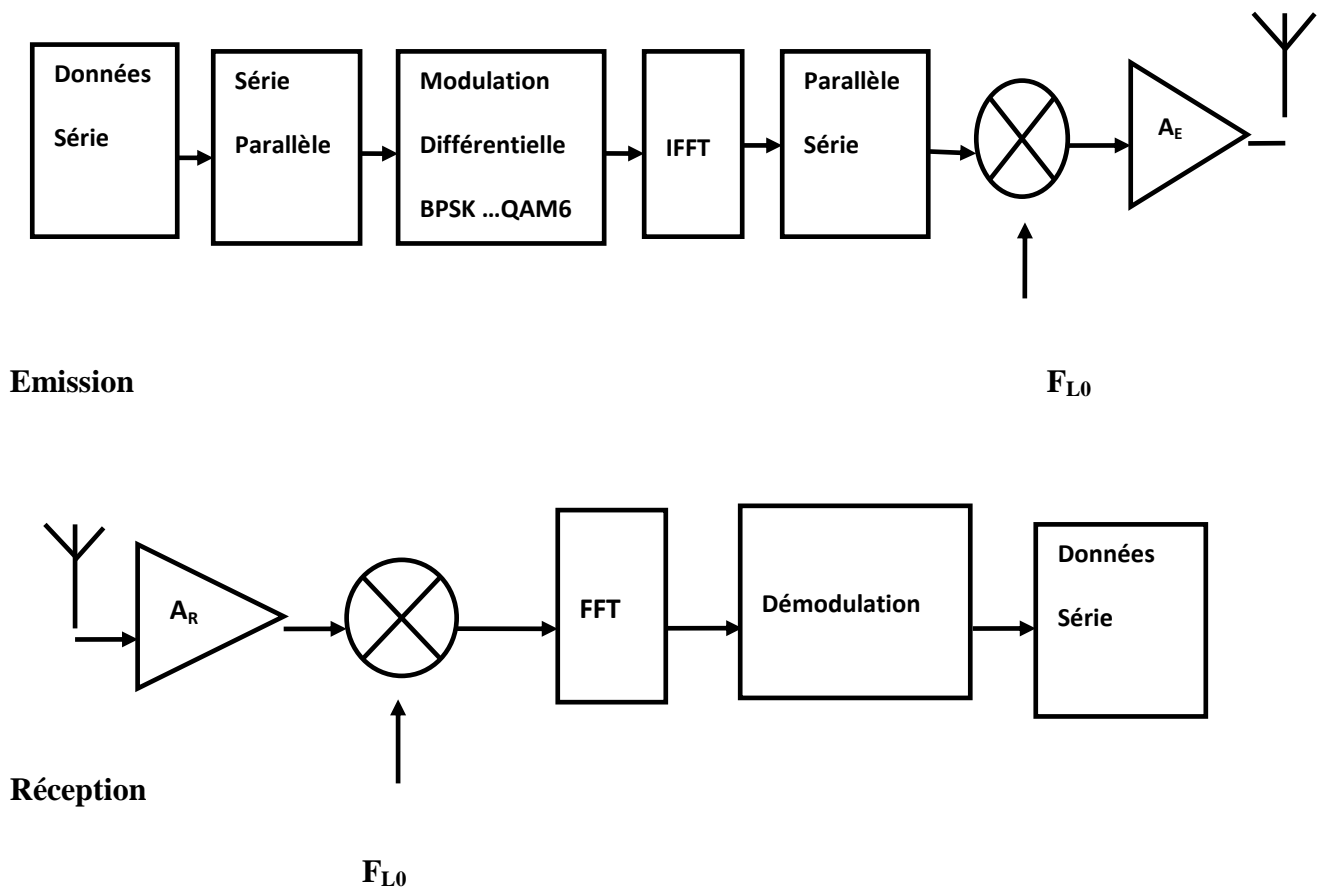


Fig. II.7. Schéma bloc d'un système OFDM.

Le schéma représente un système OFDM complet. Il comporte un émetteur, un récepteur et un canal radio à travers lequel se fait la transmission. On a encore le bloc de modulation numérique (QAM, 16-QAM ...), les blocs FFT et IFFT pour la modulation et démodulation des sous-porteuses et enfin les blocs caractérisant le canal de transmission. A la sortie, on

rejoint les flux de données parallèles pour reconstituer les données initiales.

II.6. Les canaux des trajets multiples

Nous avons une relation entre le nombre de coefficients de la réponse temporelle discrète du canal et la corrélation entre les différents coefficients de la transformée de Fourier discrète de la réponse temporelle discrète du canal. Plus la réponse temporelle est courte, plus les coefficients de la transformée de Fourier discrète de la réponse temporelle du canal sont corrélés. De façon duale une relation similaire existe entre l'étalement temporel de la réponse du canal et la sélectivité en fréquence du canal.

II.6.1. Problèmes des trajets-multiples

Transmettre un train numérique par voie hertzienne n'est pas une mince affaire et c'est la raison pour laquelle c'est la technologie qui a demandé le plus de temps pour émerger.

Le canal est caractérisé par plusieurs phénomènes physiques :

- La réflexion du signal sur un obstacle.
- La réfraction du signal lorsque celui-ci traverse un milieu d'indice différent de celui d'où il provient.
- La défraction due à un obstacle (masques).
- Atténuation due à l'air et aux phénomènes météorologiques s'y produisant.
- Interférences avec les différentes sources radiofréquences et bruits de canal et celui du système d'émission et réception.
- Evanouissement du signal par effet des trajets multiples.

Tous ces phénomènes physiques entraînent des échos (propagation par trajets multiples due à la présence d'obstacles) pouvant de l'annulation du signal à un instant et une fréquence donnée. Par conséquent, lorsqu'on est en réception fixe, portable ou mobile, la probabilité de recevoir uniquement une onde directe provenant d'un émetteur est très faible. On va donc recevoir le signal émis par l'émetteur ainsi qu'une multitude de signaux atténués et retardés provenant des différents échos (voir la figure)

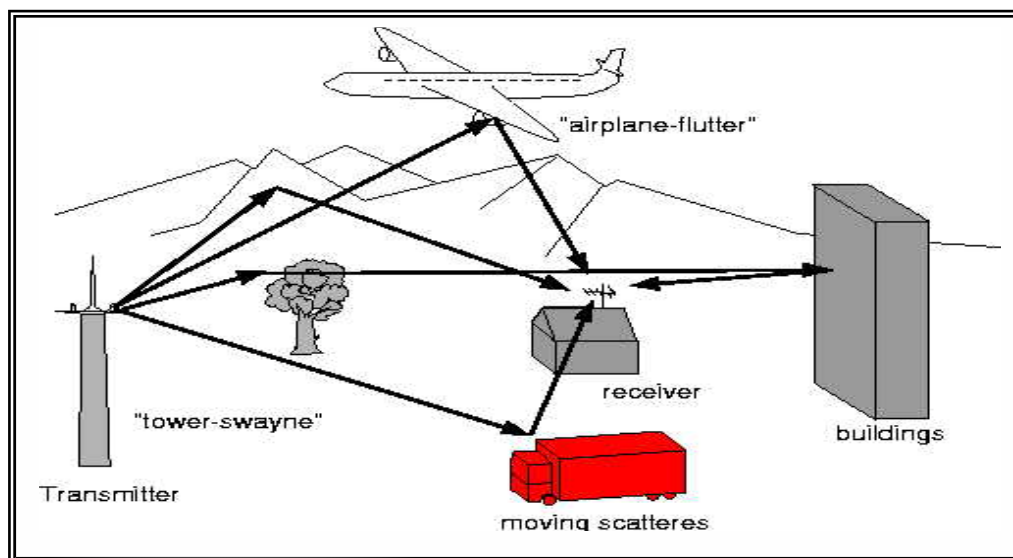


Fig. II.8. Trajet multiple dans un canal radio.

II.7. Méthode de multiplexes

Dans notre environnement, nous échangeons continuellement des informations en utilisant l'interface air. Dans le cas de la téléphonie mobile, on nome l'interface air « Um ». Les trois principales méthodes utilisées pour se partager un même interface sont : la fréquence, le temps ou le code.

La première méthode utilisée par les systèmes analogique est la répartition en fréquence

FDMA (Frequency Division Multiple Access) par la suite, avec les systèmes numériques apparait la répartition dans le temps TDMA (Time Division Multiple Access).

Le dernier système est la répartition par étalement spectral CDMA (Code Division Multiple Access) ou un code permet de repartir l'accès à notre interface unique.

II.7.1. FDMA (Frequency Division Multiple Access)

La technique FDMA a été la première méthode développée et utilisée dans les systèmes de téléphonie analogique. Pour ce type d'accès multiples une bande de fréquence est allouée à chaque utilisateur. L'ensemble des juxtaposés est transmis sur un même canal de transmission. En réception, un filtre sélectif accordé sur la bande de fréquence de l'utilisateur désiré permet de récupérer les données.

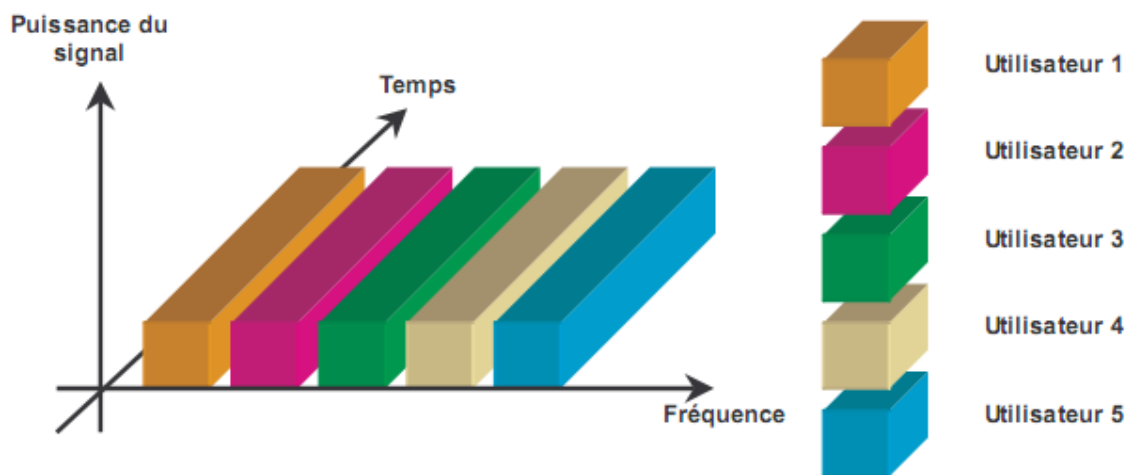


Fig. II.9. La technique d'accès multiple FDMA.

II.7.2. TDMA (Time Division Multiple Access)

La méthode TDMA est basée sur la répartition des ressources en temps. Les utilisateurs partagent la même bande passante et émettent les données à transmettre dans les différents intervalles de temps ou « slot » qui leur sont alloués. Le récepteur effectue l'opération de démultiplexage pour récupérer les données.

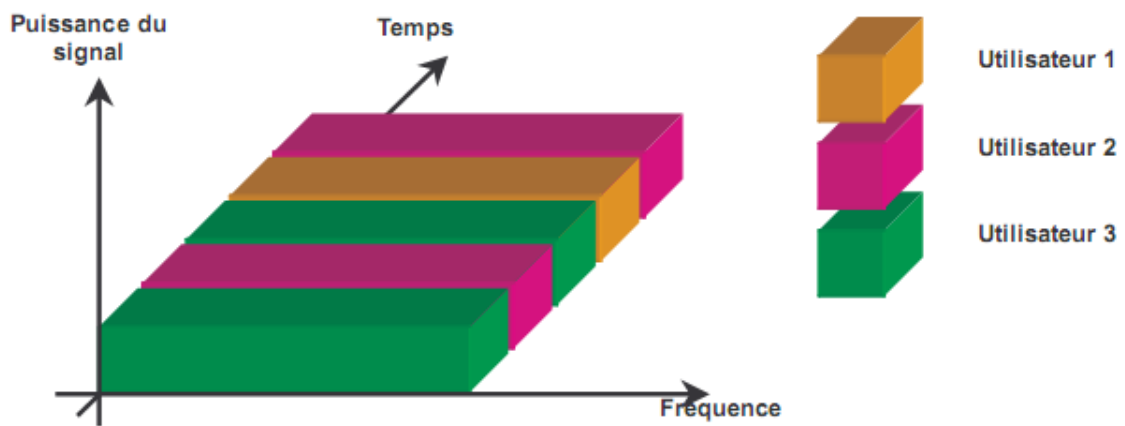


Fig. II.10. La technique d'accès multiple TDMA.

II.7.3. CDMA (Code Division Multiple Access)

Avec la méthode CDMA, tous les utilisateurs ont accès simultanément à la totalité de la bande, ils sont distingués à la réception grâce à des codes distincts pour chacun d'entre eux. Cette technique permet de multiplexer des informations sur un seul support de manière synchrone ou asynchrone.

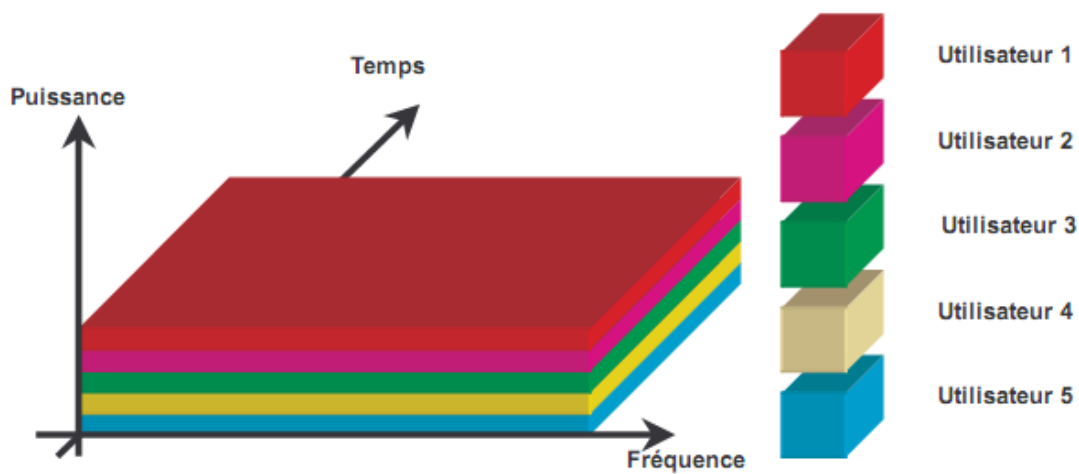


Fig. II.11. La technique d'accès multiple CDMA.

III. Simulation OFDM sous MATLAB

III.1. Schéma de simulation d'un système de modulation OFDM

Pour l'étude de la couche physique d'un système de télécommunication basé sur la modulation OFDM nous avons choisi de travailler avec les blocs OFDM de Simulink de MATLAB. Le schéma choisi est celui de la norme IEEE 802.16 OFDM sans le codage spatio-temporel. Le schéma est donné en fig. III.1. ce schéma est composé d'un émetteur récepteur OFDM avec le canal de transmission AWGN représentant la propagation dans l'air avec un bruit gaussien et muni d'un modèle de trajets multiples. Les données sont générées par un générateur de signal aléatoire de Bernoulli. Après conversion série parallèles, les données sont codées et modulent les différentes composantes du spectre OFDM en utilisant la modulation BPSK, QPSK et QAM. Le nombre de bits par symbole est choisi en fonction du SNR (Rapport Signal sur Bruit). Le SNR est affecté pendant la propagation dans le canal de transmission. Pour évaluer les erreurs de transmission nous avons calculé le taux d'erreur des bits des signaux transmis.

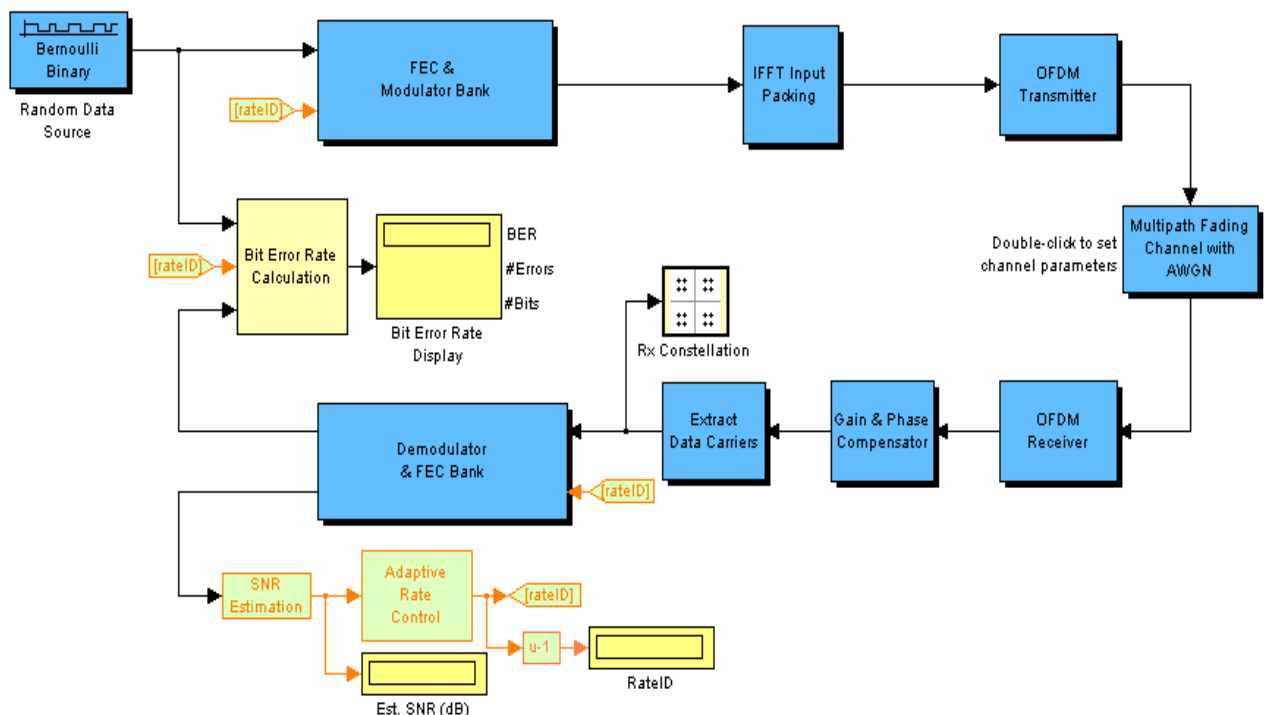


Fig.III.1. Schéma de simulation de la couche physique IEEE 802.16-2004 OFDM.

III.2. Paramètre de simulation du système OFDM

Pour notre étude nous avons choisi une bande passante du canal de 3,5 MHz avec 256 composantes. 192 sous-porteuses pour la transmission de données, 8 sous-porteuses sont utilisées comme pilotes du canal et le reste (2 x 28) comme sous-porteuses de garde pour séparer spatialement dans le domaine spectral les sous-porteuses de données, ainsi diminuer le taux de recouvrement spectral entre les sous-porteuses. Pour chaque composante spectrale (sous-porteuse) de l'OFDM on fait correspondre un symbole correspondant à un certain nombre de bits générés par le générateur aléatoire de Bernoulli. Les seuils du SNR sont choisis à priori, ils permettent de basculer entre les différents types de modulations. La table III.1. donne la répartition de type de modulation en fonction du SNR.

Plages du SNR en dB	type de modulation
$SNR \leq 4$	BPSK
$4 < SNR \leq 10$	QPSK 1/2
$10 < SNR \leq 12$	QPSK 3/4
$12 < SNR \leq 19$	16-QAM1/2
$19 < SNR \leq 22$	16-QA M3/4
$22 < SNR \leq 28$	64-QAM1/2
$28 < SNR$	64-QAM3/4

Table III.1. Table d'adaptation de la modulation en fonction du SNR.

III.3. Générateur binaire de Bernoulli (simulation de données)

Les données binaires sont générées à l'aide d'un générateur de nombres binaires aléatoires utilisant une distribution de Bernoulli. Le nombre 0 est généré avec une probabilité de p et le 1 est généré avec une probabilité égale à $q = 1 - p$. La distribution de Bernoulli a une valeur moyenne de $m(x) = 1 - p$ et la variance $var(x) = p(1 - p)$. Dans notre cas nous avons choisi

$p = q = 0,5$. La figure III.2. montre le schéma utilisé pour générer ce type de données en utilisant un générateur de nombres aléatoires de distribution uniforme. Dans notre exemple nous avons généré 864 bits correspondant à 108 octets.

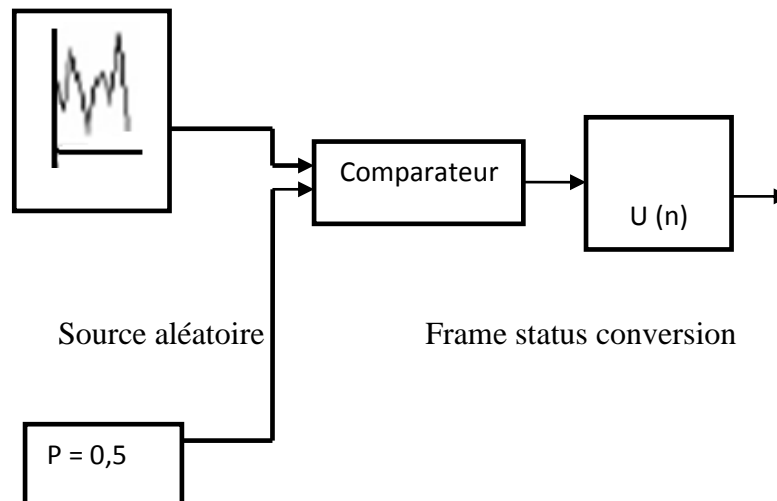


Fig.III.2. Générateur de Bernoulli.

III.4. Emetteur

Comme le montre la fig.III.1. les données série issues du générateur de Bernoulli sont transformées en parallèle et passent par le modulateur qui génère les sous-porteuses selon la constellation correspondante à la modulation choisie (Modulation Bank). Les données de sortie sont un complexe dont la phase et l'amplitude correspondent à la position dans la constellation. Ensuite, on ajoute les 8 composantes pilotes dans le bloc (IFFT Input padding) ainsi que la composante de garde à mi-bande. 2 x 28 sous-porteuses de garde sont intégrées dans le module suivant et la FFT inverse est appliquée pour obtenir le signal temporel à transmettre.

III.4.1 Modulateur adaptatif (Modulation Bank)

Ce module est montré en fig.III.3. ce module est consacré à la modulation. Nous disposons de 6 seuils de SNR en fonction de quoi le système choisi le type de modulation. Les seuils sont traduits par un paramètre d'indentification u . u prend une valeur entière entre 0 et 6 selon la plage du SNR estimé à partir du signal reçu à la réception. La Table III.2. montre l'équivalent u pour une plage donnée du SNR.

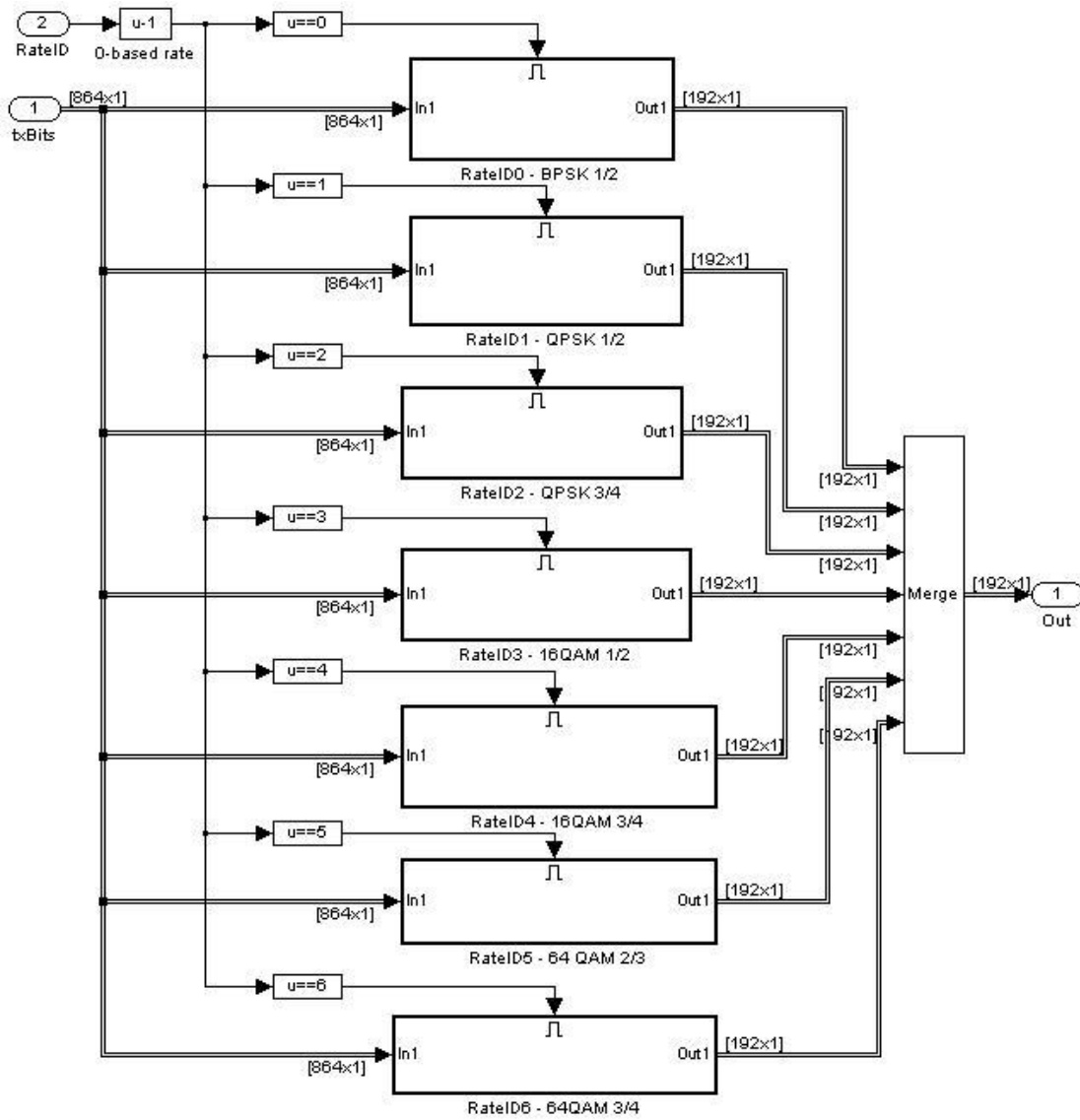


Fig.III.3. Modulateur adaptatif.

Plages du SNR en dB	u
$SNR \leq 4$	0
$4 < SNR \leq 10$	1
$10 < SNR \leq 12$	2
$12 < SNR \leq 19$	3
$19 < SNR \leq 22$	4
$22 < SNR \leq 28$	5
$28 < SNR$	6

Table III.2. Paramètres u pour le choix de la modulation.

Pour $u = 0$ la modulation est réalisée par un module de modulation BPSK. Dans ce cas le nombre de bits codé par sous-porteuse est 1 bit. Un codage par convolution est ajouté pour la correction des erreurs et un autre est utilisé pour le réarrangement des bits dans la trame (Interleaver) pour augmenter les performances de restitution de l'information. Le taux de codage (code rate) est fixé à $\frac{1}{2}$ (voir Fig.III.4 (a)).

Pour $u=1$ et $u=2$, la modulation utilisée est la QPSK. Dans ce cas en plus du module de codage par convolution et le module Interleaver on utilise le codeur Reed-Solomon. Deux taux de codages sont utilisés, pour $u=1$ le taux est de $\frac{1}{2}$ et pour $u=2$ ce taux est de $\frac{3}{4}$ (voir Fig.III.4 (b)). De la même façon on utilise la modulation QAM pour $u \geq 3$ (voir Fig.III.4 (c)).

La trame de sortie se compose de nombres complexes correspondant aux constellations déterminées par les différentes modulations et le tout est multiplexé pour former les 192 symboles de la trame OFDM.

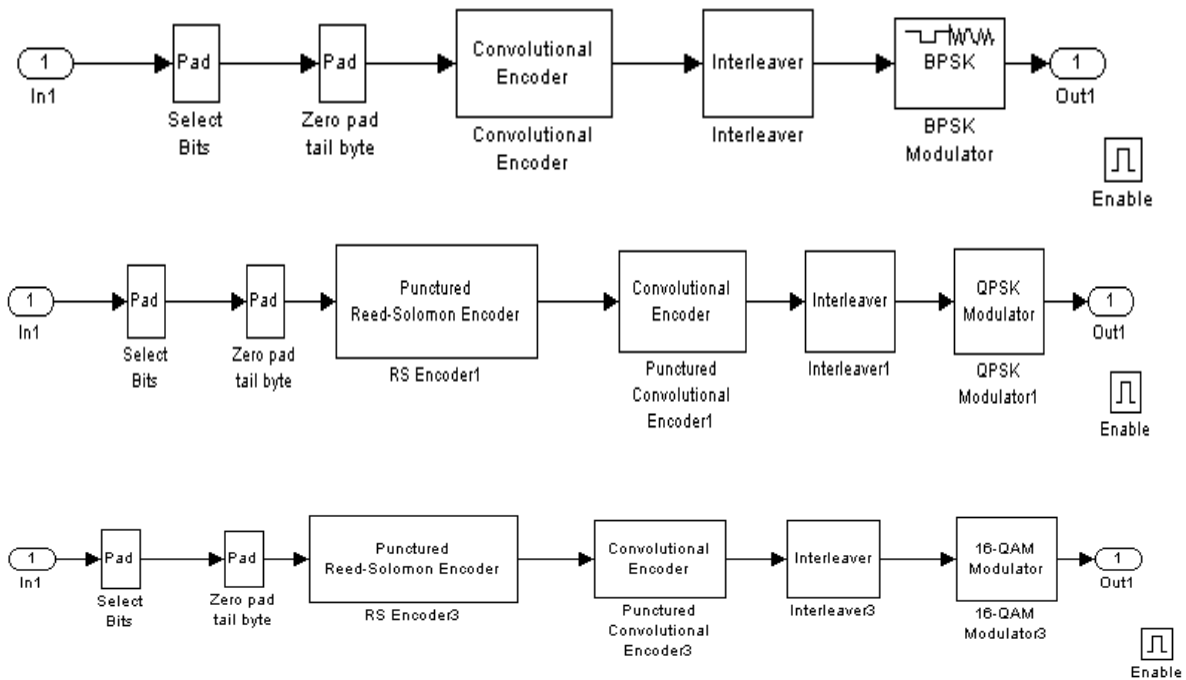


Fig. III.4. Schémas de modulation. (a) BPSK, (b) QPSK, (c) QAM.

III. 4.2. Insertion de fréquences pilotes

Afin de détecter le signal OFDM pendant la phase écoute et se synchroniser, on insert un certain nombre de sous-porteuses, sans données, d'amplitude et phase constantes. Ces pilotes permettent par ailleurs de mesurer les dérives en amplitude et phase pour déterminer les paramètres du canal de transmission tel que le SNR (rapport signal sur bruit) et de déterminer les conditions de propagation dans le canal. Dans notre application nous avons étudié un système à 8 composantes pilotes inséré comme c'est indiqué en fig.III.5. ces composantes enlace les différentes sous-bandes ou sous-porteuses du signal Numériques de sorte à mesurer les dégradations du signal tout le long de la bande OFDM. L'amplitude de chaque fréquence pilote est choisie à 1 avec une phase 0° pour 4 et une phase de π pour le reste des pilotes. Dans cet exemple de simulation, une bande de garde (bande nulle) est insérée pour séparer la bande supérieure de la bande inférieure de l'OFDM. Cette bande est représenté Sur fig.III.5. par le petit rectangle au milieu portant la lettre « -c- ».

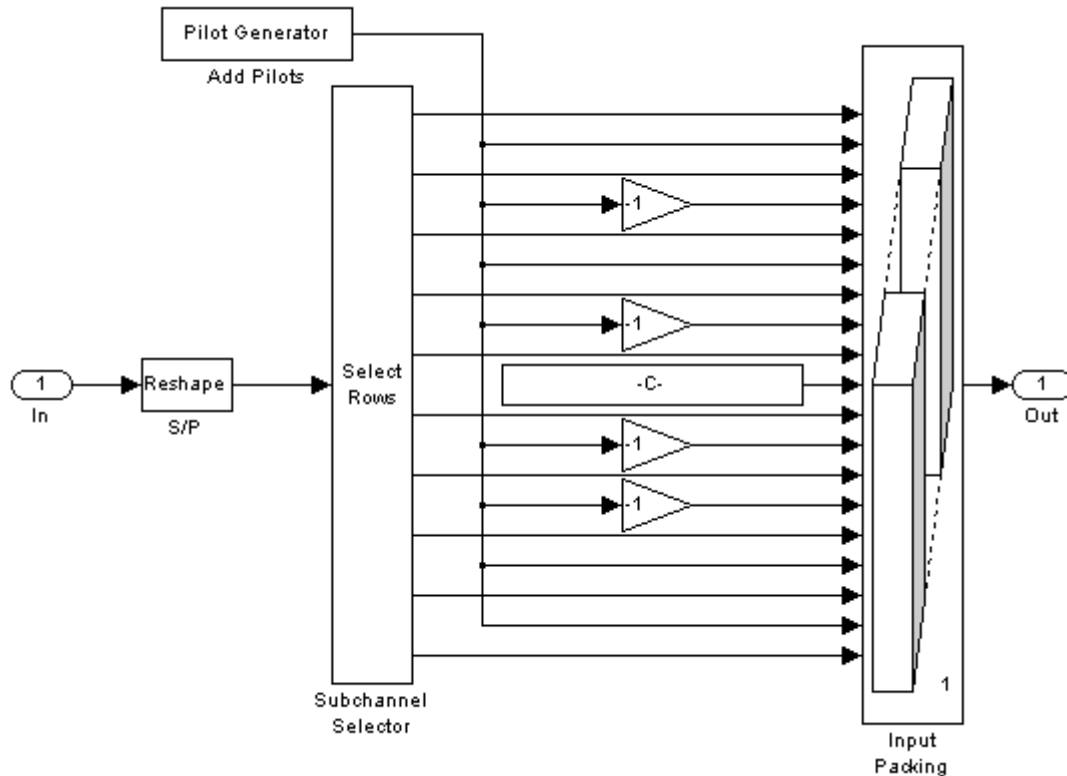


Fig.III.5. Insertion de composantes pilotes dans le paquet OFDM.

III. 4.3. Emetteur OFDM

Le paquet OFDM est ensuite enrichi par les bandes de garde afin de séparer au mieux les sous-porteuses de données OFDM. Cette séparation permet de réduire davantage les recouvrements spectraux entre les sous-porteuses adjacentes. Dans cette application on insert en tout 2×28 sous bandes de garde. Une fois les sous-bandes de garde sont insérées on applique la FFT inverse sur les 256 points. On obtient ainsi un vecteur ou un paquet de données dans le domaine temporel de 256 valeurs.ces valeurs sont amplifiées puis transmises en série dans le canal après avoir rajouter un préfixe qui délimite la trame de transmission dans le domaine temporel (voir fig.III.6). un gain est appliqué pour ramener le niveau du signal à un niveau suffisant pour la transmission.

Pour faire les corrections nécessaires sur l'amplitude et la phase, on transmet un paquet OFDM dont les valeurs des sous-porteuses sont connues par l'émetteurs et le récepteur ce paquet appelé préambule est utilisé à la réception pour déterminer les dérives amplitude et phase pour chaque sous-canal de la transmission.

Les variations d'amplitude et de phase du préambule sont utilisées pour faire les corrections pour les sous-porteuses des données à la réception.

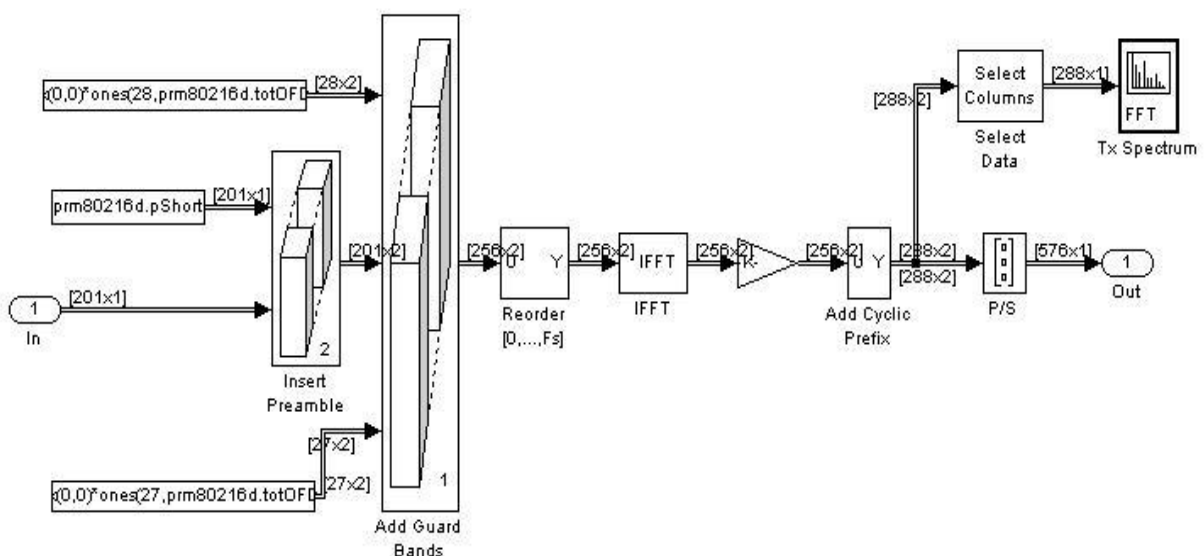


Fig.III.6. Emetteur OFDM.

III. 5. Canal de transmission

Dans cette étude, le canal de transmission est modélisé par le module AWGN de Toolbox du Simulink qui simule un bruit de canal de type gaussien en fonction du SNR introduit. En considérant les trajets multiples nous avons simulé leur effet sur le signal reçu. Pour ce faire nous avons pris en compte le trajet principal ainsi que deux trajets secondaires. Dans cette étude nous avons varié le SNR du canal de 0 à 30 dB pour les deux cas : sans et avec trajets multiples.

III.6. Récepteur

Comme nous l'avons déjà montré en fig.III.1. lorsque le signal se présente au niveau du récepteur il est rééchantillonné et reformater pour pouvoir appliquer la FFT sur les 256 points. Le préfixe cyclique de la trame du signal temporel est enlevé puis les sous-porteuses sont estimées en appliquant la FFT. Avant la démodulation on procède à la correction du gain et de phase puis éliminer les composantes OFDM de pilotage (module gain phase compensator). La démodulation consiste à décoder les constellations BPSK, QPSK ou QAM estimées à partir des sous-porteuses OFDM.

III.6.1. Récepteur OFDM

Comme le montre la fig.III.7. le signal est mis en forme par le convertisseur série parallèle. Ensuite, pour chaque trame le préfixe cyclique est enlevé. En appliquant la FFT on obtient le spectre OFDM avec l'ensemble des porteuses pilotes et les bandes de garde. Un gain est appliqué pour la normalisation. Enfin, les bandes de gardes sont enlevées dans ce module.

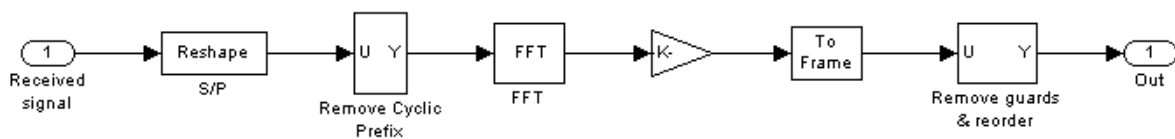


Fig.III.7. Récepteur OFDM.

Les fréquences pilotes sont éliminées en utilisant le bloc de la fig.III.8. dans ce cas on récupère les sous-porteuses OFDM de données. On rappelle que l'amplitude et la phase de ces sous-porteuses occupent les des positions dans les constellations des différentes modulations.

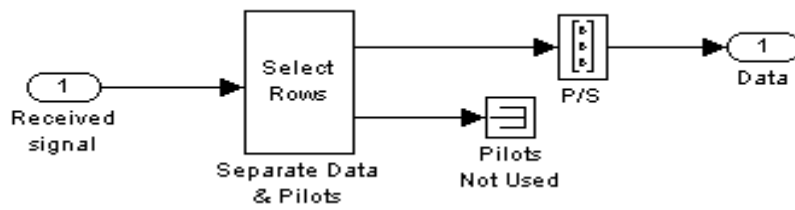


Fig.III.8. Extrait de porteuses de données.

III.6.2. Démodulateur (Forward Error Correction Bank)

Ce module effectue la démodulation. Comme dans le cas de l'émission la démodulation est contrôlée par le paramètre d'identification de modulation u . Les sous porteuses sont démodulées selon la valeur de u (voir table III.2). Après démodulation BPSK, QPSK ou QAM les données sont multiplexées et transmises en série. La fig.III.9. montre le module de démodulation de MATLAB.

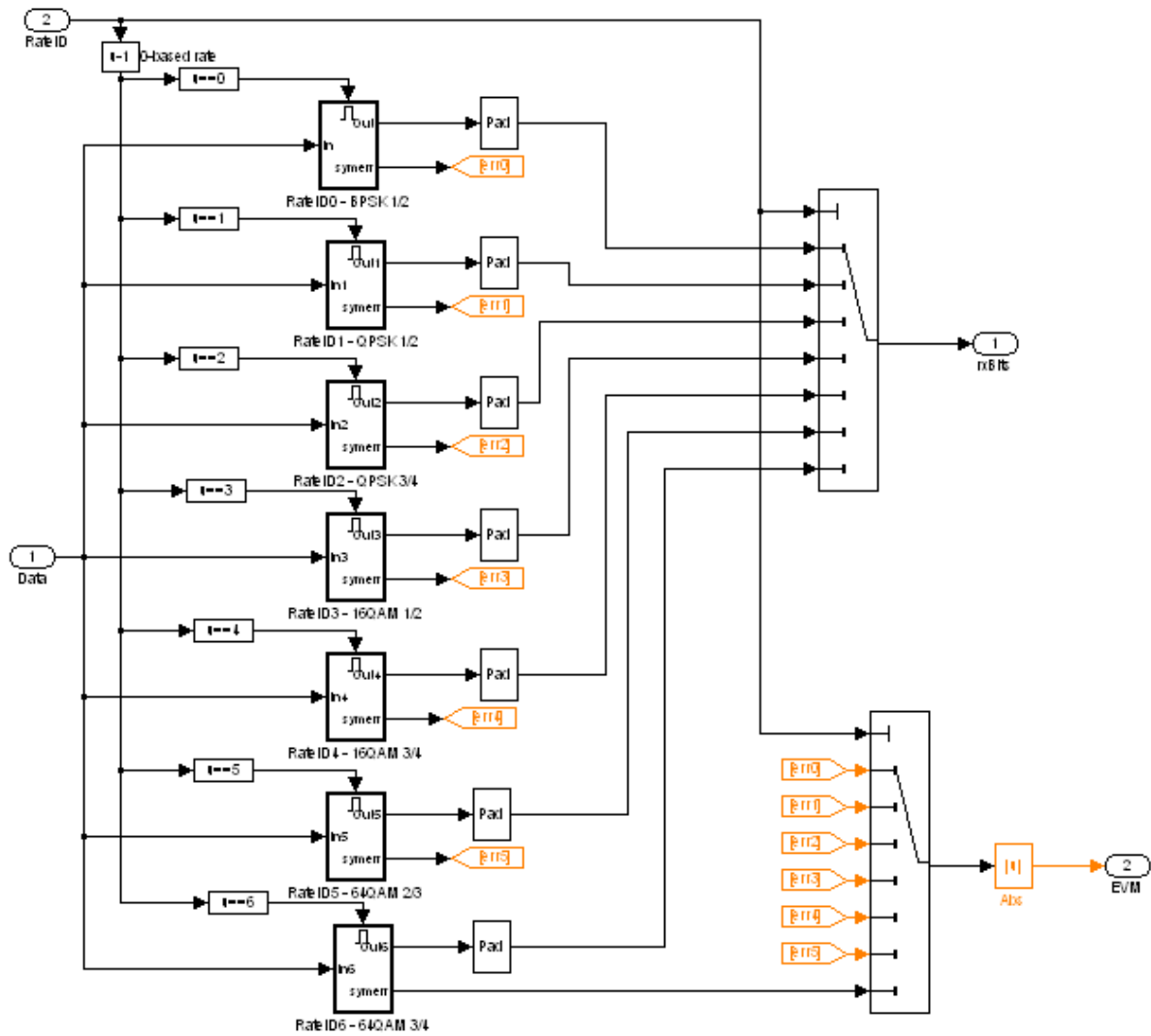


Fig.III.9. Démodulateur OFDM (FEC Bank).

Le schéma de démodulation est donné en fig.III.10. après démodulation BPSK on applique l'algorithme de désentrelacement. Ensuite on procède au décodage de Viterbi pour récupérer les données d'origine et éventuellement corriger certaine erreurs. Dans le cas de la modulation QPSK et QAM le décodage Reed-Solomon est appliqué.

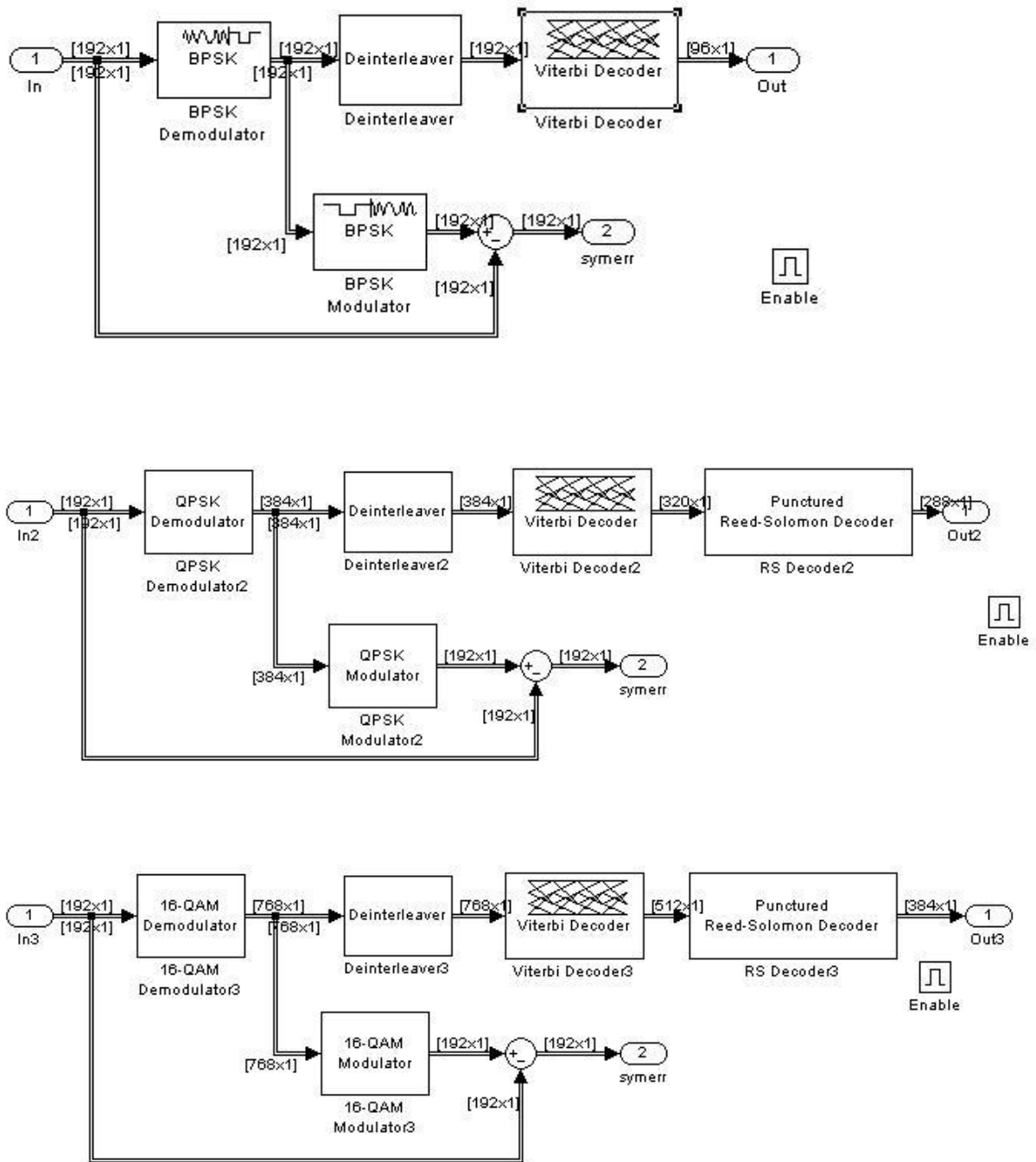


Fig.III.10. Schéma de démodulation en OFDM.

III.7. Calcul du taux d'erreur (BER)

Le taux d'erreur est le rapport entre le nombre de bits erronés et le nombre total de bits transmis. Le taux d'erreur est calculé par le bloc donné en fig.III.11. comme le montre cette figure les bits le signal émis et reçu sont utilisés comme entrées pour le calcul du BER.

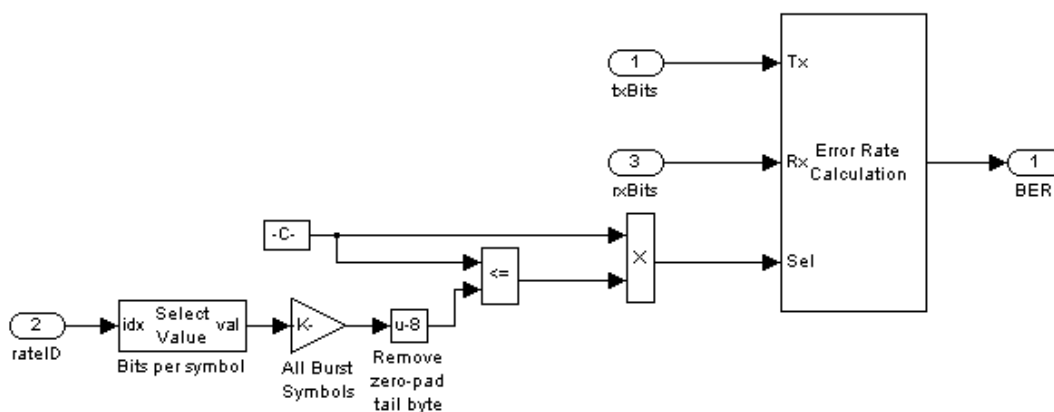


Fig.III.11. Bit Error Rate Calculation.

III.8. Système d'évaluation des performances

Pour évaluer les performances du système OFDM nous avons effectué plusieurs simulations. Pour ce faire, Nous avons considéré deux cas. Le premier c'est la propagation dans un canal sans trajets multiples. Dans ce cas, le SNR est varié de 0 à 30 dB. Pour chaque valeur du SNR on enregistre le BER le taux d'erreur. Le deuxième cas est celui avec des trajets multiples. Dans ce cas, nous avons considéré une propagation dans le même canal que précédent mais avec deux trajets secondaires par rapport au trajet principal.

III.8.1. Données Simulées

Comme nous l'avons déjà expliqué les données sont générées par le générateur de Bernoulli avec une probabilité de 0,5 pour les nombres binaires 1 et 0. La fig.III.12. cette figure montre quelques données d'une trame générées sur une période de 8 μ s.

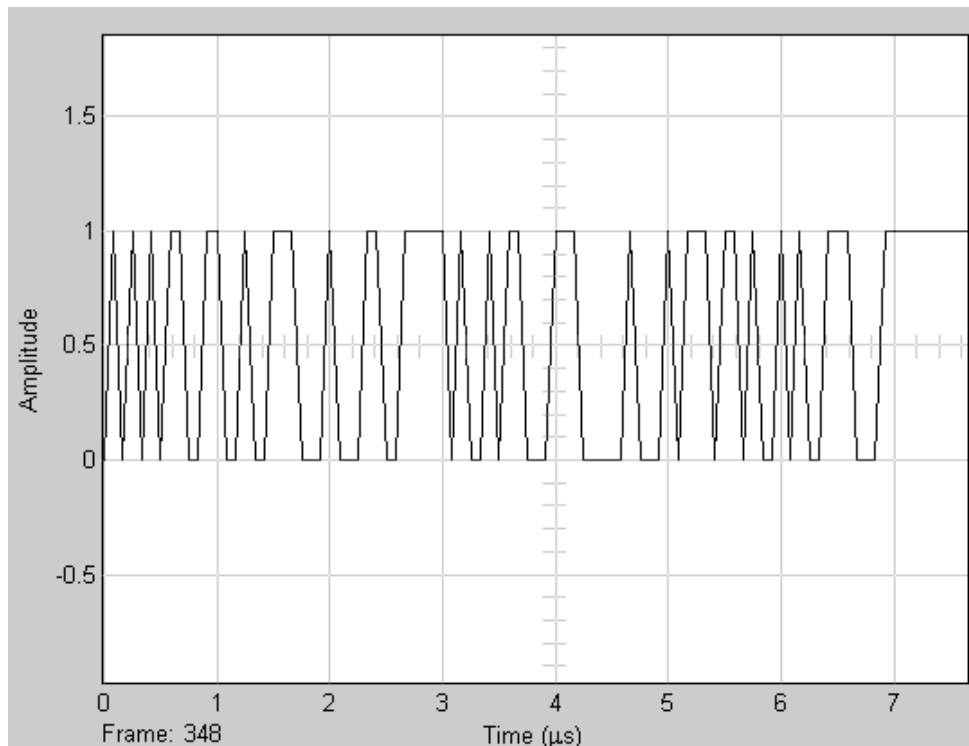


Fig.III.12. Données du générateur de Bernoulli.

Le spectre correspondant à une trame de donnée est illustré sur la fig.III.13. comme le montre cette figure le spectre est de large bande et correspond à la largeur fixée dans les paramètres de simulation qui est de 3,5MHz.

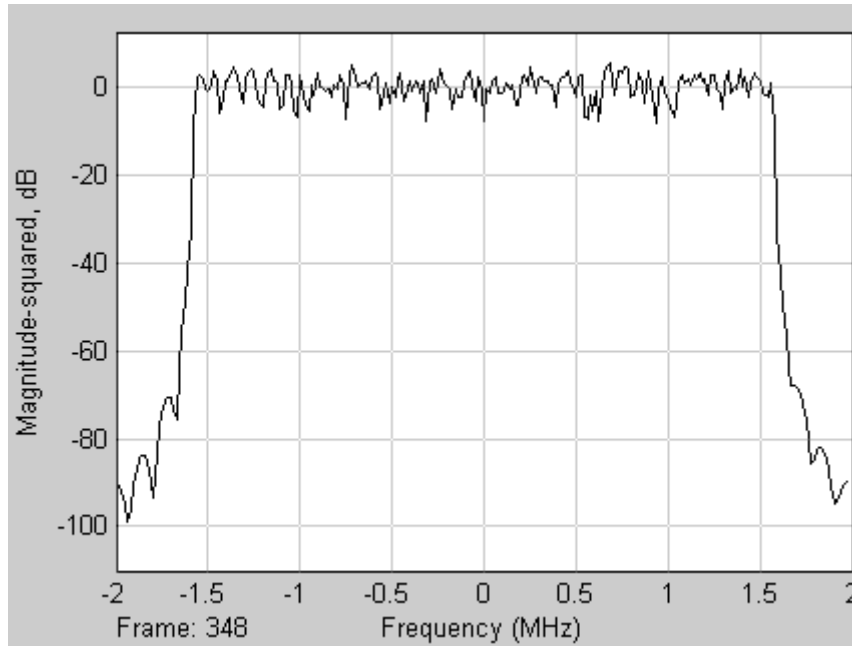


Fig.III.13. Exemple d'un spectre OFDM des données transmises.

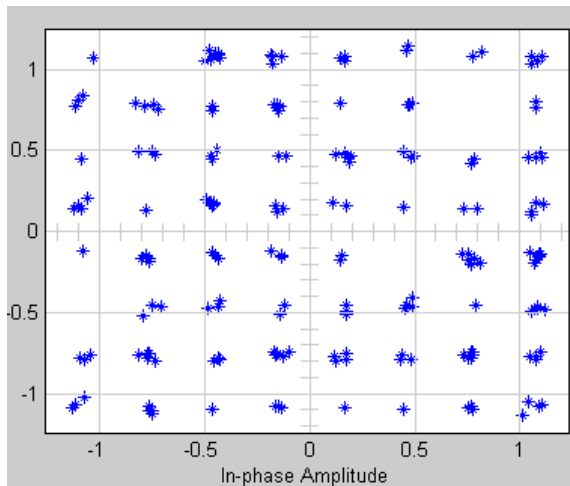
III.8.2. Résultats de simulation sans trajets multiples (no fading)

Les résultats obtenus pour un canal sans trajets multiple sont illustrés à la table III.3. sur cette table nous avons montré les différent SNR imposé au canal de transmission, le BER calculé, le nombre de bits erronés, l'ensemble de bits générés ou transmis, le SNR estimé par le système et le paramètre d'identification u des modulation utilisé

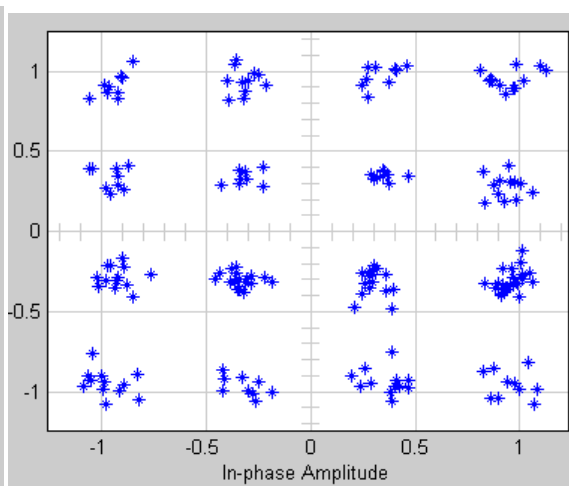
SNR	BER	Errors	Bits	SNR	u
30	0	0	2.95 e +005	29.67	6
20	0	0	2.95 e +005	20.09	4
10	0	0	8.064 e +004	9.643	1
6	0	0	6.394 e +004	5.556	1
3	0.002336	72	3.082 e +004	1.639	0
0	0.02021	619	3.062 e+004	-2.42	0

Table III. 3. Résultats de simulation pour un canal sans trajets multiples.

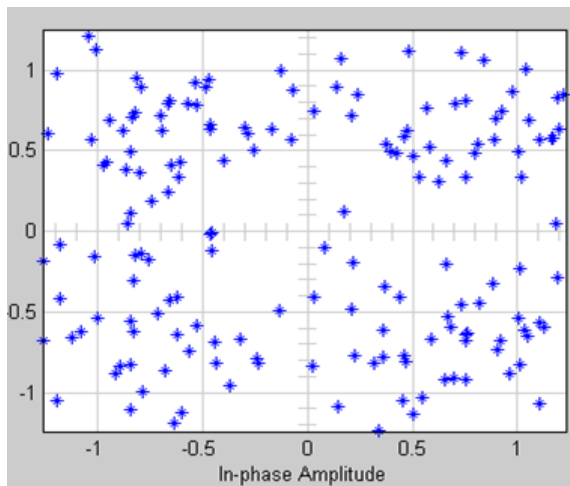
En fonction du SNR nous montrons sur la fig.III.14. les différents types de modulations utilisées dans le spectre OFDM. Ceci illustre l'adaptation du type de modulation en fonction du SNR pour réduire les erreurs de transmission. En ayant un identificateur de modulation u plus petit, le nombre de bits codés par symbole est diminué. Ceci a pour effet, malheureusement, de réduire le débit. Ainsi on montre sur la fig.III.14. la modulation QAM-64 est utilisée pour un SNR de 30dB qui utilise 6 bits par symbole mais on réduit le nombre de bits par symbole jusqu'à l'utilisation de la BPSK où on utilise qu'un bit par symbole.



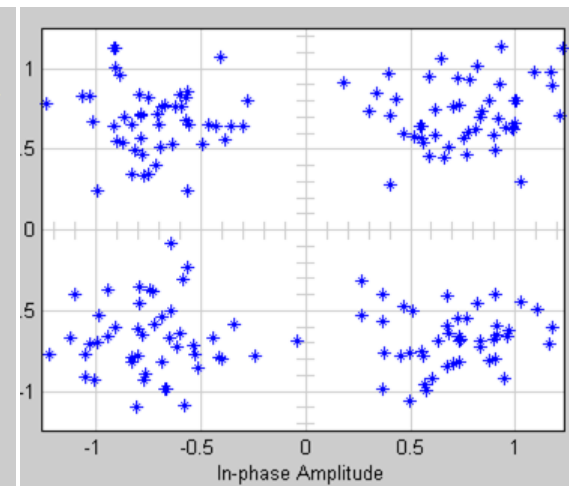
SNR=30



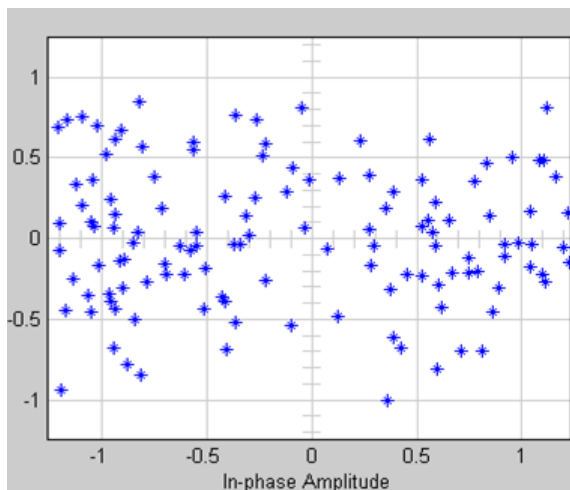
SNR=20



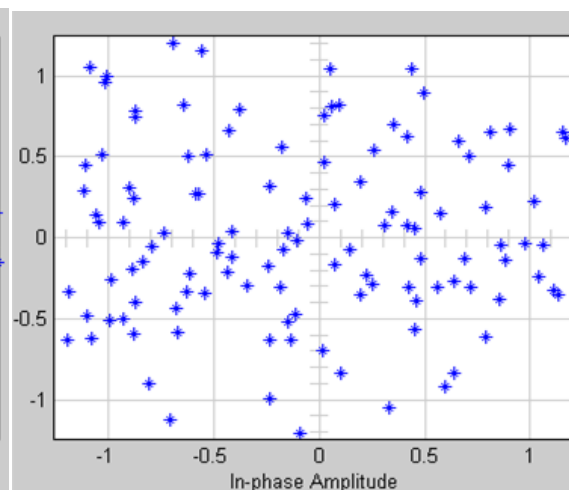
SNR=10



SNR=6



SNR=3



SNR=0

Fig.III.14. Constellation des différentes modulations en fonction du SNR pour le canal sans trajets multiples.

III.8.3. Résultats de simulation avec trajets multiples

Nous avons considéré deux trajets secondaires par rapport au trajet principal. Les paramètres des trajets sont donnés à la table III.4.

Délais (μ s)	Gain (dB)
0,4	-5
0,9	-10

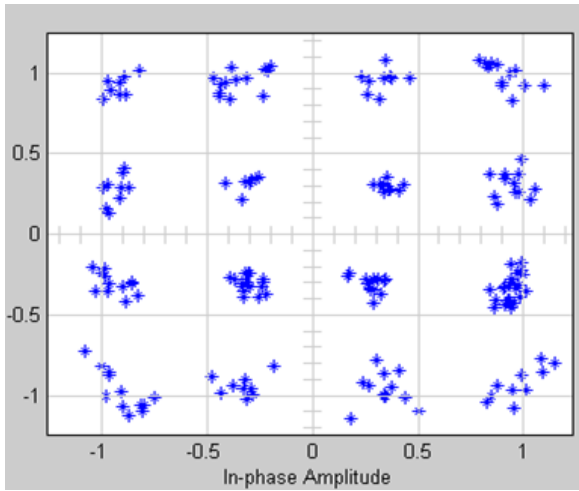
Table III.4. Paramètres du trajet multiple.

Les résultats obtenus pour le trajet multiple sont résumés sur la table III.5. comme dans le cas précédent nous avons calculé les différents paramètres pour une variation du SNR entre 0 et 30 dB.

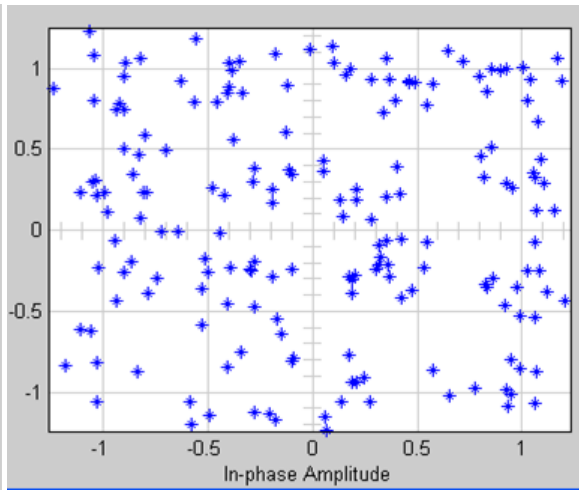
SNR	BER	Errors	Bits	SNR	u
30	0	0	1.886 e+005	19.5	4
20	0.0003	49	1.3 e +005	13.01	3
10	0.0002	9	3.062 e +004	1.964	0
6	0.1072	3284	3.062 e +004	-2.753	0
3	0.4349	1.332 e +004	3.062 e +004	-11.85	0
0	0.4875	1.493 e +004	3.062 e +004	-11.23	0

Table. III.5. Résultats de simulation avec trajets multiples.

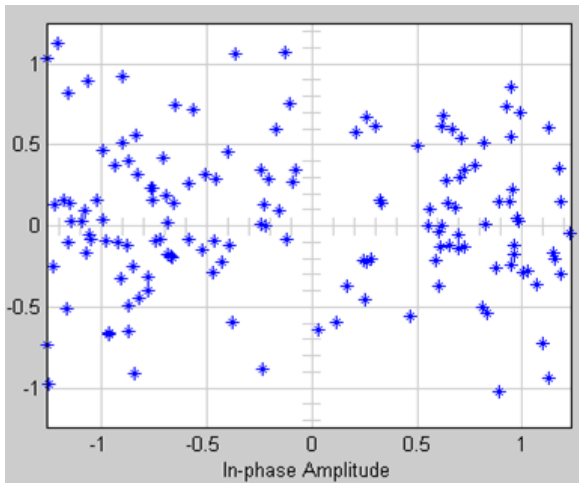
La fig.III.15. montre les types de modulations utilisées. Dans le cas du trajet multiple la modulation QAM-64 n'a pas été utilisée et la BPSK est la plus utilisée. Ceci est certainement dû à l'atténuation du signal par l'effet des trajets secondaire.



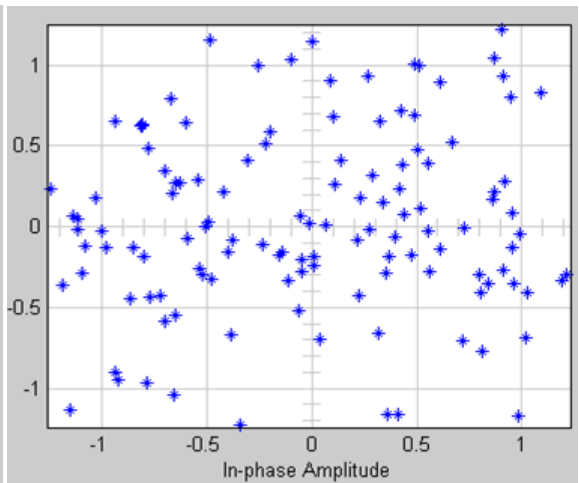
SNR=30



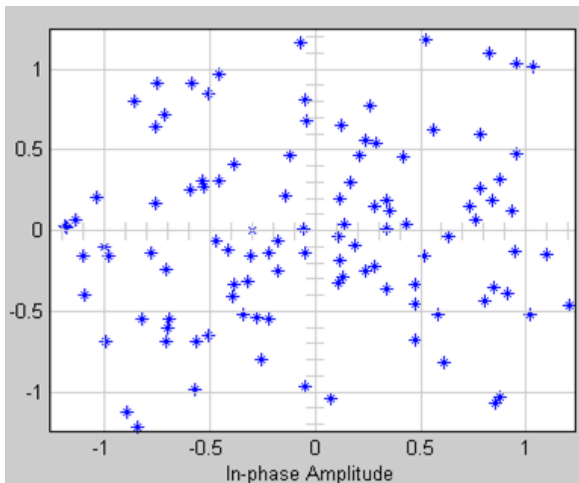
SNR=20



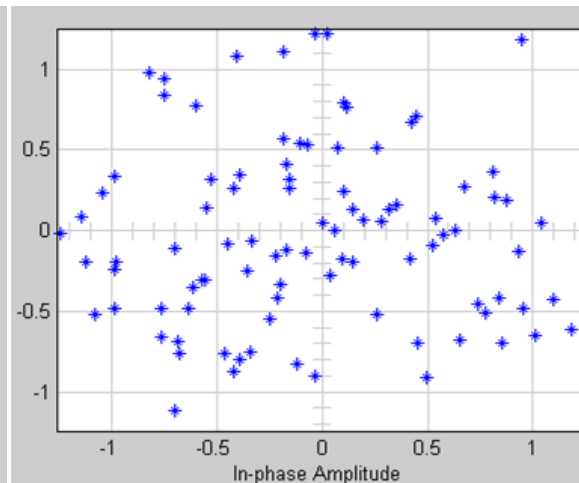
SNR=10



SNR=6



SNR=3



SNR=0

Fig.III.15. Constellation des différentes modulations en fonction du SNR pour le canal avec trajets multiples.

III.8.4. Discussion

Le résultat du taux d'erreur BER est illustré en fig.III.16. cette figure montre l'évolution de l'erreur en fonction du SNR dans les deux cas d'une liaison sans trajet et avec trajet multiples. D'après ce résultat lorsque le champ radio-fréquence des porteuses OFDM ne rencontre aucun obstacle les performances sont plus importantes que dans le cas où les ondes se propagent dans des chemins multiples. Dans ce cas, le système OFDM est obligé de réduire le nombre de bits par symbole dans les sous-porteuses. Ceci réduit d'une façon significative le débit.

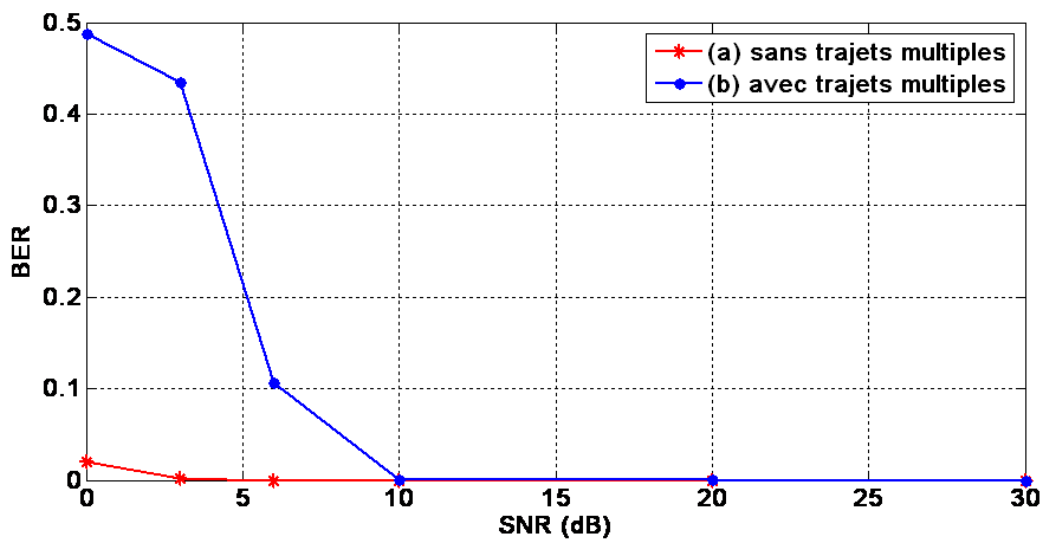


Fig.III. 16. Variation du BER en fonction du SNR

- (a) Sans trajets multiples.
- (b) Avec trajets multiples.

Conclusion

L'OFDM est un système multi porteuses qui garantit une bonne immunité contre les interférences inter porteuses grâce à leur caractéristique principale qui est l'orthogonalité. Dans le schéma bloc d'une liaison de communication par un système OFDM les données à émettre sont organisées en blocs présentées de manière parallèle au module IFFT. Dans une technique OFDM, toutes les porteuses sont modulées par les données d'un même utilisateur. Pour supporter la notion de multi utilisateur, OFDM est combinée soit avec TDMA, soit avec FDMA ou le CDMA.

A l'aide du MATLAB nous avons effectué l'étude de la couche physique d'un système OFDM de la norme IEEE 802.16. Cette étude nous a permis de comprendre toutes les étapes nécessaires pour pouvoir se communiquer en utilisant le système émetteur/récepteur OFDM. A l'émission les données modulent les sous-porteuses en utilisant les modulateurs BPSK, QPSK ou QAM. Les données sont codées et entrelacées pour réduire les erreurs de transmission. Les codes convolutionnel et Reed-Solomon sont utilisés pour permettre de récupérer certains bits erronés. Des fréquences pilotes sont ajoutées pour identifier la bande OFDM. Des séparations sont réalisées dans le spectre de l'OFDM pour éviter un recouvrement spectral à l'aide d'intervalles de garde. Le type de modulation est adapté en fonction du SNR dans le canal de transmission.

Les performances du système OFDM sont testées par simulation. Pour ce faire, nous avons varié le SNR du canal entre 0 et 30 dB pour les deux cas : sans et avec trajets multiples. Plusieurs trames OFDM sont transmises, puis on calcule le taux d'erreur « BER » pour chaque SNR. Les résultats ont montré que le trajet multiple influe considérablement sur la qualité de la transmission. L'erreur de transmission peut atteindre les 50 % dans le cas d'un SNR = 0 dB.

Toutefois, les paramètres de simulation de cette étude sont considérés d'une façon arbitraire. C'est pourquoi ce travail peut être amélioré et étendu en utilisant des paramètres réalistes ou réels. En utilisant le modèle numérique du terrain on peut d'une façon exhaustive simuler l'effet des trajets multiples d'un système OFDM donné et évaluer ainsi, d'une façon proche de la réalité, les performances d'un tel système.

Bibliographie

- [1] BOUCIF Amar Bensaber, Math Info. Introduction aux Réseaux Téléphonique, UQTR
- [2] A. DUPRET et A. FISCHER. Cours de télécommunication : Modulation Numérique et Multiplexage des Signaux. IUT Villetaneuse, Département de Génie Télécom et Réseaux,
- [3] X.Lagrange,P.Godlew,S.Tabane, 1997 Transmission des Télécommunications,
« Réseaux GSM-DCS » Ed. Le nouveau GSM : Cours de Hermes .
- [4] Naufal M.Saad ,3 mai 2005.Thèse présentée à l'université de Limoges pour l'obtention du grade Docteur « Télécommunications des Hautes Fréquences et Optiques ».
- [5] Daniel Kofman, 21 Septembre 2001,Synthèse sur l'évolution des Réseaux de Télécommunication Ecole Nationale Supérieure des Télécommunication,
- [6] Virginie Dégardin , 19 Décembre 2002. Thèse présentée à l'université des sciences et technologie de Lille pour obtenir le titre Docteur « Analyse de la faisabilité d'une transmission de données haut débit sur le réseau électrique basse tension »,
- [7] Pascal Nicolas, Cours de réseaux Maitrise Informatique Université d'Angers.
- [8] Mohamed Aymen Chalouf, 3 Décembre 2009.Thèse présentée à l'Université de Bordeaux I, Ecole Doctorale de Mathématique et d'Informatique : « Offre de service dans les réseaux de nouvelle génération »,
- [9] , Hussein Hijazi ,Genevieve Jourdain et Laurent Ros, 25 Novembre 2008. « Estimation du canal radio-mobile à évolution rapide dans les systèmes à modulation OFDM ».
- [10] Nathalie DEFEZ Dess Tni ,Mathieux GBILLOUX GARCIA , Frédéric 6 Décembre 2004.Thèse présentée à l'université de Montpellier II,: « Le WIMAX ».

Sites internet :

<http://www.huawei.com>

[http : //www .abcelectronique.com](http://www.abcelectronique.com)

[http://www.technique de l'ingenieur.com.](http://www.technique de l'ingenieur.com)

ANNEXE

Convolutional code

In telecommunication, a **convolutional code** is a type of error-correcting code in which

- each m -bit information symbol (each m -bit string) to be encoded is transformed into an n -bit symbol, where m/n is the code rate ($n \geq m$) and
- the transformation is a function of the last k information symbols, where k is the constraint length of the code.

Where convolutional codes are used

Convolutional codes are used extensively in numerous applications in order to achieve reliable data transfer, including digital video, radio, mobile communication, and satellite communication. These codes are often implemented in concatenation with a hard-decision code, particularly Reed Solomon. Prior to turbo codes, such constructions were the most efficient, coming closest to the Shannon limit.

Convolutional encoding

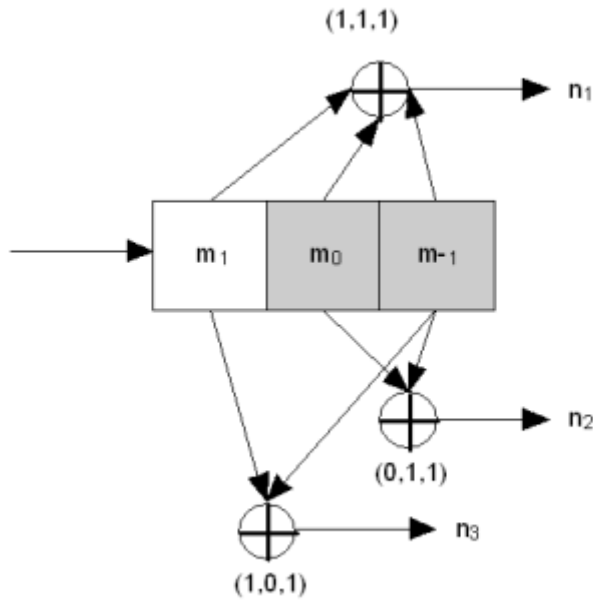
To convolutionally encode data, start with k memory registers, each holding 1 input bit. Unless otherwise specified, all memory registers start with a value of 0. The encoder has n modulo-2 adders (a modulo 2 adder can be implemented with a single Boolean XOR gate, where the logic is: $0+0 = 0$, $0+1 = 1$, $1+0 = 1$, $1+1 = 0$), and n generator polynomials — one for each adder (see figure below). An input bit m_1 is fed into the leftmost register. Using the generator polynomials and the existing values in the remaining registers, the encoder outputs n bits. Now bit shift all register values to the right (m_1 moves to m_0 , m_0 moves to m_{-1}) and wait for the next input bit. If there are no remaining input bits, the encoder continues output until all registers have returned to the zero state.

The figure below is a rate $1/3$ (m/n) encoder with constraint length (k) of 3. Generator polynomials are $G_1 = (1,1,1)$, $G_2 = (0,1,1)$, and $G_3 = (1,0,1)$. Therefore, output bits are calculated (modulo 2) as follows:

$$n_1 = m_1 + m_0 + m_{-1}$$

$$n_2 = m_0 + m_{-1}$$

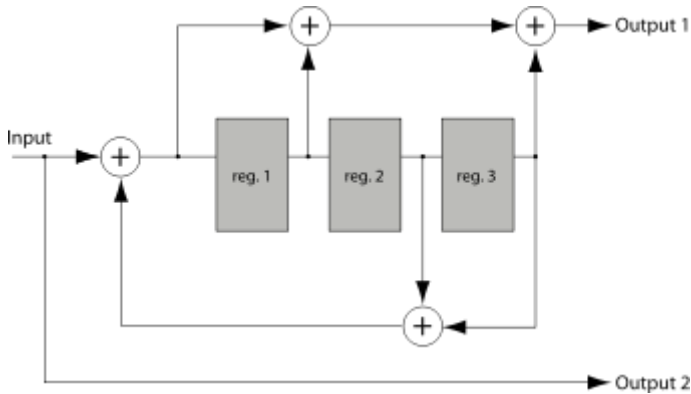
$$n_3 = m_1 + m_{-1}$$



Img.1. Rate 1/3 non-recursive, non-systematic convolutional encoder with constraint length 3

Recursive and non-recursive codes

The encoder on the picture above is a *non-recursive* encoder. Here's an example of a recursive one:



Img.2. Rate 1/2 recursive, systematic convolutional encoder with constraint length 4

One can see that the input being encoded is included in the output sequence too (look at the output 2). Such codes are referred to as *systematic*; otherwise the code is called *non-systematic*.

Recursive codes are almost always systematic and, conversely, non-recursive codes are non-systematic. It isn't a strict requirement, but a common practice.

Impulse response, transfer function, and constraint length

A convolutional encoder is called so because it performs a *convolution* of the input stream with the encoder's *impulse responses*:

$$y_i^j = \sum_{k=0}^{\infty} h_k^j x_{i-k},$$

where x is an input sequence, y^j is a sequence from output j and h^j is an impulse response for output j .

A convolutional encoder is a discrete linear time-invariant system. Every output of an encoder can be described by its own transfer function, which is closely related to a generator polynomial. An impulse response is connected with a transfer function through Z-transform.

Transfer functions for the first (non-recursive) encoder are:

- $H_1(z) = 1 + z^{-1} + z^{-2},$
- $H_2(z) = z^{-1} + z^{-2},$
- $H_3(z) = 1 + z^{-2}.$

Transfer functions for the second (recursive) encoder are:

- $H_1(z) = \frac{1 + z^{-1} + z^{-3}}{1 - z^{-2} - z^{-3}},$
- $H_2(z) = 1.$

Define m by

$$m = \max_i \text{polydeg}(H_i(1/z))$$

where, for any rational function $f(z) = P(z)/Q(z),$

$$\text{polydeg}(f) = \max(\text{deg}(P), \text{deg}(Q)).$$

Then m is the maximum of the polynomial degrees of the $H_i(1/z),$ and the *constraint length* is defined as $K = m + 1.$ For instance, in the first example the constraint length is 3, and in the second the constraint length is 4.

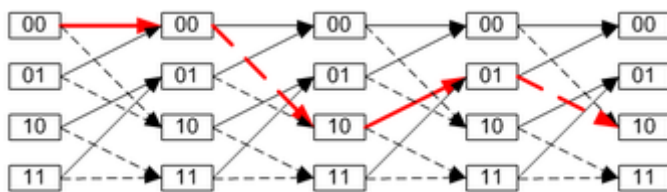
Trellis diagram

A convolutional encoder is a finite state machine. An encoder with n binary cells will have 2^n states.

Imagine that the encoder (shown on Img.1, above) has '1' in the left memory cell (m_0), and '0' in the right one (m_1). (m_1 is not really a memory cell because it represents a current value).

We will designate such a state as "10". According to an input bit the encoder at the next turn can convert either to the "01" state or the "11" state. One can see that not all transitions are possible (e.g., a decoder can't convert from "10" state to "00" or even stay in "10" state).

All possible transitions can be shown as below:



Img.3. A trellis diagram for the encoder on Img.1. A path through the trellis is shown as a red line. The solid lines indicate transitions where a "0" is input and the dashed lines where a "1" is input.

An actual encoded sequence can be represented as a path on this graph. One valid path is shown in red as an example.

This diagram gives us an idea about *decoding*: if a received sequence doesn't fit this graph, then it was received with errors, and we must choose the nearest *correct* (fitting the graph) sequence. The real decoding algorithms exploit this idea.

Free distance and error distribution

The **free distance** (d) is the minimal Hamming distance between different encoded sequences. The *correcting capability* (t) of a convolutional code is the number of errors that can be corrected by the code. It can be calculated as

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor.$$

Since a convolutional code doesn't use blocks, processing instead a continuous bitstream, the value of t applies to a quantity of errors located relatively near to each other. That is, multiple groups of t errors can usually be fixed when they are relatively far apart.

Free distance can be interpreted as the minimal length of an erroneous "burst" at the output of a convolutional decoder. The fact that errors appear as "bursts" should be accounted for when

designing a concatenated code with an inner convolutional code. The popular solution for this problem is to interleave data before convolutional encoding, so that the outer block (usually Reed-Solomon) code can correct most of the errors.

Decoding convolutional codes

Several algorithms exist for decoding convolutional codes. For relatively small values of k , the Viterbi algorithm is universally used as it provides maximum likelihood performance and is highly parallelizable. Viterbi decoders are thus easy to implement in VLSI hardware and in software on CPUs with SIMD instruction sets. Longer constraint length codes are more practically decoded with any of several sequential decoding algorithms, of which the Fano algorithm is the best known. Unlike Viterbi decoding, sequential decoding is not maximum likelihood but its complexity increases only slightly with constraint length, allowing the use of strong, long-constraint-length codes. Such codes were used in the Pioneer program of the early 1970s to Jupiter and Saturn, but gave way to shorter, Viterbi-decoded codes, usually concatenated with large Reed-Solomon error correction codes that steepen the overall bit-error-rate curve and produce extremely low residual undetected error rates.

Both Viterbi and sequential decoding algorithms return hard-decisions: the bits that form the most likely codeword. An approximate confidence measure can be added to each bit by use of the Soft output Viterbi algorithm. Maximum a posteriori (MAP) soft-decisions for each bit can be obtained by use of the BCJR algorithm.

Popular convolutional codes

An especially popular Viterbi-decoded convolutional code, used at least since the Voyager program has a constraint length k of 7 and a rate r of $1/2$.

- Longer constraint lengths produce more powerful codes, but the complexity of the Viterbi algorithm increases exponentially with constraint lengths, limiting these more powerful codes to deep space missions where the extra performance is easily worth the increased decoder complexity.
- Mars Pathfinder, Mars Exploration Rover and the Cassini probe to Saturn use a k of 15 and a rate of $1/6$; this code performs about 2 dB better than the simpler $k=7$ code at a cost of $256\times$ in decoding complexity (compared to Voyager mission codes).

Viterbi decoder

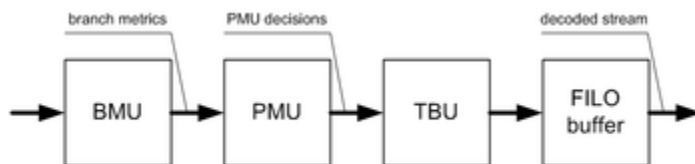
A **Viterbi decoder** uses the Viterbi algorithm for decoding a bitstream that has been encoded using forward error correction based on a convolutional code.

There are other algorithms for decoding a convolutionally encoded stream (for example, the Fano algorithm). The Viterbi algorithm is the most resource-consuming, but it does the maximum likelihood decoding. It is most often used for decoding convolutional codes with constraint lengths $k \leq 10$, but values up to $k=15$ are used in practice.

Viterbi decoding was developed by Andrew J. Viterbi and published in the paper "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm", IEEE Transactions on Information Theory, Volume IT-13, pages 260-269, in April, 1967.

There are both hardware (in modems) and software implementations of a Viterbi decoder.

Hardware implementation

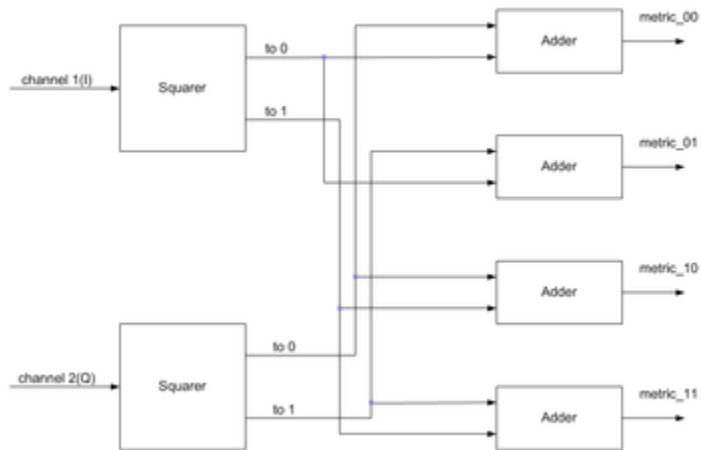


A common way to implement a hardware viterbi decoder

A hardware Viterbi decoder for basic (not perforated) code usually consists of the following major blocks:

- Branch metric unit (BMU)
- Path metric unit (PMU)
- Traceback unit (TBU)

Branch metric unit (BMU)



A sample implementation of a branch metric unit

A branch metric unit's function is to calculate *branch metrics*, which are normed distances between every possible symbol in the code alphabet, and the received symbol.

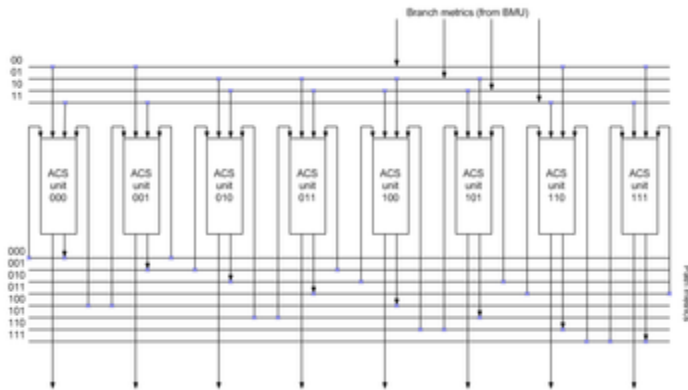
There are hard decision and soft decision Viterbi decoders. A hard decision Viterbi decoder receives a simple bitstream on its input, and a Hamming distance is used as a metric. A soft decision Viterbi decoder receives a bitstream containing information about the *reliability* of each received symbol. For instance, in a 3-bit encoding, this *reliability* information is encoded as follows:

value	meaning
000	strongest 0
001	relatively strong 0
010	relatively weak 0
011	weakest 0
100	weakest 1
101	relatively weak 1
110	relatively strong 1
111	strongest 1

Of course, it is not the only way to encode reliability data.

The *squared* Euclidean distance is used as a metric for soft decision decoders.

Path metric unit (PMU)

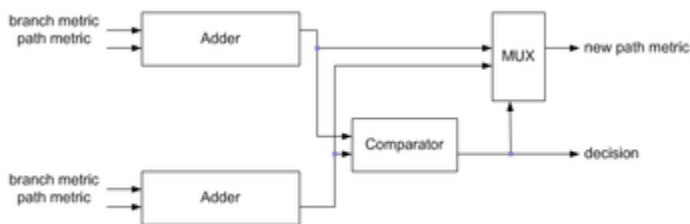


A sample implementation of a path metric unit for a specific K=4 decoder

A path metric unit summarizes branch metrics to get metrics for 2^{K-1} paths, one of which can eventually be chosen as *optimal*. Every clock it makes 2^{K-1} decisions, throwing off wittingly nonoptimal paths. The results of these decisions are written to the memory of a traceback unit.

The core elements of a PMU are *ACS (Add-Compare-Select)* units. The way in which they are connected between themselves is defined by a specific code's trellis diagram.

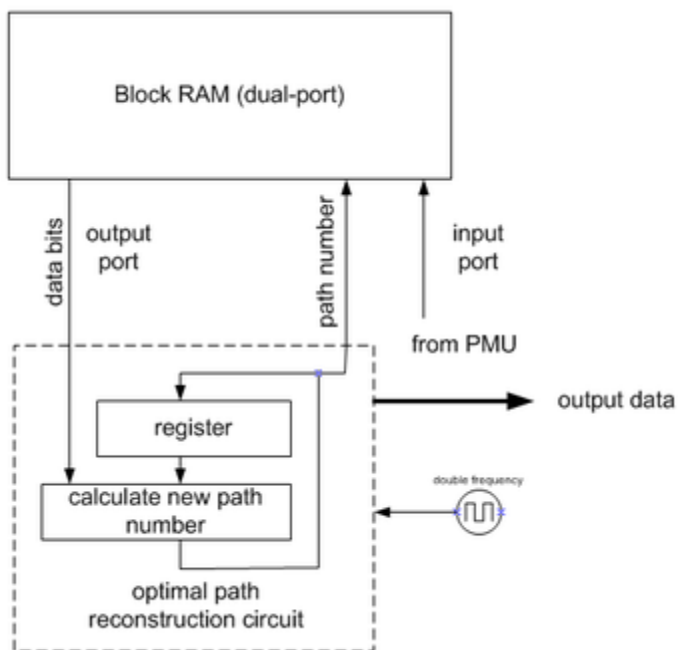
Since branch metrics are always ≥ 0 , there must be an additional circuit preventing metric counters from overflow (it isn't shown on the image). An alternate method that eliminates the need to monitor the path metric growth is to allow the path metrics to "roll over", to use this method it is necessary to make sure the path metric accumulators contain enough bits to prevent the "best" and "worst" values from coming within $2^{(n-1)}$ of each other. The compare circuit is essentially unchanged.



A sample implementation of an ACS unit

It is possible to monitor the noise level on the incoming bit stream by monitoring the rate of growth of the "best" path metric. A simpler way to do this is to monitor a single location or "state" and watch it pass "upward" through say four discrete levels within the range of the accumulator. As it passes upward through each of these thresholds, a counter is incremented that reflects the "noise" present on the incoming signal.

Traceback unit (TBU)



A sample implementation of a traceback unit

Back-trace unit restores an (almost) maximum-likelihood path from the decisions made by PMU. Since it does it in inverse direction, a viterbi decoder comprises a FILO (first-in-last-out) buffer to reconstruct a correct order.

Note that the implementation shown on the image requires double frequency. There are some tricks that eliminate this requirement.

Implementation issues

Quantization for soft decision decoding

In order to fully exploit benefits of soft decision decoding, one needs to quantize input signal properly. The optimal quantization zone width is defined by the following formula:

$$T = \sqrt{N_0/2^k},$$

where N_0 is a noise power spectral density, and k is a number of bits for soft decision.

Euclidean metric computation

The squared norm distance (l_2) distance between the received and the actual symbols in the code alphabet may be further simplified into a linear sum/difference form, which makes it less computationally intensive.

Consider a 1/2 convolutional coder, which generates 2 bits (00, 01, 10 or 11) for every input bit (1 or 0). These *Return-to-Zero* signals are translated into a *Non-Return-to-Zero* form shown alongside.

code alphabet	vector mapping
00	1, 1
01	1, -1
10	-1, 1
11	-1, -1

Each received symbol may be represented in vector form as $\mathbf{v}_r = \{r_0, r_1\}$, where r_0 and r_1 are soft decision values, whose magnitudes signify the *joint reliability* of the received vector, \mathbf{v}_r .

Every symbol in the code alphabet may, likewise, be represented by the vector $\mathbf{v}_i = \{\pm 1, \pm 1\}$.

The actual computation of the Euclidean distance metric is:

$$D = (\vec{v}_r - \vec{v}_i)^2 = \vec{v}_r^2 - 2\vec{v}_r \vec{v}_i + \vec{v}_i^2$$

Each square term is a normed distance, depicting the *energy* of the symbol. For ex., the *energy* of the symbol $\mathbf{v}_i = \{\pm 1, \pm 1\}$ may be computed as

$$\overrightarrow{v}_i^2 = (\pm 1)^2 + (\pm 1)^2 = 2$$

Thus, the energy term of all symbols in the code alphabet is constant (at (*normalized*) value 2).

The *Add-Compare-Select* (ACS) operation compares the metric distance between the received symbol $||\mathbf{v}_r||$ and any 2 symbols in the code alphabet whose paths merge at a node in the corresponding trellis, $||\mathbf{v}_i^{(0)}||$ and $||\mathbf{v}_i^{(1)}||$. This is equivalent to comparing

$$D_0 = \overrightarrow{v}_r^2 - 2\overrightarrow{v}_r\overrightarrow{v}_i^0 + \overrightarrow{v}_i^0^2$$

and

$$D_1 = \overrightarrow{v}_r^2 - 2\overrightarrow{v}_r\overrightarrow{v}_i^1 + \overrightarrow{v}_i^1^2$$

But, from above we know that the *energy* of \mathbf{v}_i is constant (equal to (normalized) value of 2), and the *energy* of \mathbf{v}_r is the same in both cases. This reduces the comparison to a minima function between the 2 (middle) *dot product* terms,

$$\min(-2\overrightarrow{v}_r\overrightarrow{v}_i^0, -2\overrightarrow{v}_r\overrightarrow{v}_i^1) = \max(\overrightarrow{v}_r\overrightarrow{v}_i^0, \overrightarrow{v}_r\overrightarrow{v}_i^1)$$

since a *min* operation on negative numbers may be interpreted as an equivalent *max* operation on positive quantities.

Each *dot product* term may be expanded as

$$\max(\pm r_0 \pm r_1, \pm r_0 \pm r_1)$$

where, the signs of each term depend on symbols, $\mathbf{v}_i^{(0)}$ and $\mathbf{v}_i^{(1)}$, being compared. Thus, the *squared* Euclidean metric distance calculation to compute the *branch metric* may be performed with a simple add/subtract operation.

Traceback

The general approach to traceback is to accumulate path metrics for up to five times the constraint length ($5 * (K - 1)$), find the node with the largest accumulated cost, and begin traceback from this node.

However, computing the node which has accumulated the largest cost (either the largest or smallest integral path metric) involves finding the *maxima* or *minima* of several (usually 2^{K-1}) numbers, which may be time consuming when implemented on embedded hardware systems.

Most communication systems employ Viterbi decoding involving data packets of fixed sizes, with a fixed bit/byte pattern either at the beginning or/and at the end of the data packet. By using the known bit/byte pattern as reference, the start node may be set to a fixed value, thereby obtaining a perfect Maximum Likelihood Path during traceback.

Limitations

A physical implementation of a viterbi decoder will not yield an *exact* maximum-likelihood stream due to quantization of the input signal, branch and path metrics, and finite *traceback length*. Practical implementations do approach within 1dB of the ideal.

Perforated codes

A hardware viterbi decoder of *perforated codes* is commonly implemented in such a way:

- A deperforator, which transforms the input stream into the stream which looks like an original (imperforated) stream with ERASE marks at the places where bits were erased.
- A basic viterbi decoder understanding these ERASE marks (that is, not using them for branch metric calculation).

Punctured convolutional codes

Puncturing is a technique used to make a m/n rate code from a "basic" rate 1/2 code. It is reached by deletion of some bits in the encoder output. Bits are deleted according to puncturing matrix. The following puncturing matrices are the most frequently used:

Code rate	Puncturing matrix	Free distance (for NASA standard K=7 convolutional code)
1/2 (No perf.)	1 1	10
2/3	1 0	6

	1 1	
3/4	1 0 1 1 1 0	5
5/6	1 0 1 0 1 1 1 0 1 0	4
7/8	1 0 0 0 1 0 1 1 1 1 1 0 1 0	3

For example, if we want to make a code with rate $2/3$ using the appropriate matrix from the above table, we should take a basic encoder output and transmit every second bit from the first branch and every bit from the second one. The specific order of transmission is defined by the respective communication standard.

Punctured convolutional codes are widely used in the satellite communications, for example, in INTELSAT systems and Digital Video Broadcasting.

Punctured convolutional codes are also called "perforated".

Turbo codes: replacing convolutional codes

Simple Viterbi-decoded convolutional codes are now giving way to turbo codes, a new class of iterated short convolutional codes that closely approach the theoretical limits imposed by Shannon's theorem with much less decoding complexity than the Viterbi algorithm on the long convolutional codes that would be required for the same performance. Concatenation with an outer algebraic code (e.g., Reed-Solomon) addresses the issue of error floors inherent to turbo code designs.

Reed-Solomon error correction

In coding theory, **Reed-Solomon (RS) codes** are non-binary^[1] cyclic error-correcting codes invented by Irving S. Reed and Gustave Solomon. They described a systematic way of building codes that could detect and correct multiple random symbol errors. By adding t

check symbols to the data, an RS code can detect any combination of up to t erroneous symbols, and correct up to $\lfloor t/2 \rfloor$ symbols. As an erasure code, it can correct up to t known erasures, or it can detect and correct combinations of errors and erasures. Furthermore, RS codes are suitable as multiple-burst bit-error correcting codes, since a sequence of $b+1$ consecutive bit errors can affect at most two symbols of size b .^[2] The choice of t is up to the designer of the code, and may be selected within wide limits.

In Reed–Solomon coding, source symbols are viewed as coefficients of a polynomial $p(x)$ over a finite field. The original idea was to create n code symbols from k source symbols by oversampling $p(x)$ at $n > k$ distinct points, transmit the sampled points, and use interpolation techniques at the receiver to recover the original message. That is not how RS codes are used today. Instead, RS codes are viewed as cyclic BCH codes, where encoding symbols are derived from the coefficients of a polynomial constructed by multiplying $p(x)$ with a cyclic generator polynomial. This gives rise to an efficient decoding algorithm, which was discovered by Elwyn Berlekamp and James Massey, and is known as the Berlekamp–Massey decoding algorithm.

Reed–Solomon codes have since found important applications from deep-space communication to consumer electronics. They are prominently used in consumer electronics such as CDs, DVDs, Blu-ray Discs, in data transmission technologies such as DSL & WiMAX, in broadcast systems such as DVB and ATSC, and in computer applications such as RAID 6 systems.

Description

Original view (transmitting points)

The original concept of Reed–Solomon coding (Reed & Solomon 1960) describes encoding of k message symbols by viewing them as coefficients of a polynomial $p(x)$ of maximum degree $k - 1$ over a finite field of order N , and evaluating the polynomial at $n > k$ distinct input points. Sampling a polynomial of degree $k - 1$ at more than k points creates an overdetermined system, and allows recovery of the polynomial at the receiver given any k out of n sample points using (Lagrange) interpolation. The sequence of distinct points is created by a generator of the finite field's multiplicative group, and includes 0, thus permitting any value of n up to N .

Using a mathematical formulation, let (x_1, x_2, \dots, x_n) be the input sequence of n distinct values over the finite field F ; then the codebook \mathbf{C} created from the tuples of values obtained by evaluating every polynomial (over F) of degree less than k at each x_i is

$$\mathbf{C} = \{(f(x_1), f(x_2), \dots, f(x_n)) \mid f \in F[x], \deg(f) < k\},$$

where $F[x]$ is the polynomial ring over F , and k and n are chosen such that $1 \leq k \leq n \leq N$.

As described above, an input sequence (x_1, x_2, \dots, x_n) of $n = N$ values is created as

$$(0, \alpha^0, \alpha^1, \dots, \alpha^{N-2}),$$

where α is a primitive root of F . When omitting 0 from the sequence, and since $\alpha^{N-1} = 1$, it follows that for every polynomial $p(x)$ the function $p(\alpha x)$ is also a polynomial of the same degree, and its codeword is a cyclic left-shift of the codeword derived from $p(x)$; thus, a Reed–Solomon code can be viewed as a cyclic code. This is pursued in the classic view of RS codes, described subsequently.

As outlined in the section on a theoretical decoder, the original view does not give rise to an efficient decoding algorithm, even though it shows that such a code can work.

Classic view (Reed–Solomon codes as BCH codes)

In practice, instead of sending sample values of a polynomial, the encoding symbols are viewed as the coefficients of an output polynomial $s(x)$ constructed by multiplying the message polynomial $p(x)$ of maximum degree $k - 1$ by a generator polynomial $g(x)$ of degree $t = N - k - 1$. The generator polynomial $g(x)$ is defined by having $\alpha, \alpha^2, \dots, \alpha^t$ as its roots, i.e.,

$$g(x) = (x - \alpha)(x - \alpha^2) \cdots (x - \alpha^t) = g_0 + g_1x + \cdots + g_{t-1}x^{t-1} + x^t$$

The transmitter sends the $N - 1$ coefficients of $s(x) = p(x)g(x)$, and the receiver can use polynomial division by $g(x)$ of the received polynomial to determine whether the message is in error; a non-zero remainder means that an error was detected.^[3] Let $r(x)$ be the non-zero remainder polynomial, then the receiver can evaluate $r(x)$ at the roots of $g(x)$, and build a system of equations that eliminates $s(x)$ and identifies which coefficients of $r(x)$ are in error, and the magnitude of each coefficient's error. (Berlekamp 1984) (Massey 1969) If the system of equations can be solved, then the receiver knows how to modify his $r(x)$ to get the most likely $s(x)$.

Reed–Solomon codes are a special case of a larger class of codes called BCH codes. The Berlekamp–Massey algorithm has been designed for the decoding of such codes, and is thus applicable to Reed–Solomon codes.

To see that Reed–Solomon codes are special BCH codes, it is useful to give the following alternative definition of Reed–Solomon codes.^[4]

Given a finite field F of size q , let $n = q - 1$ and let α be a primitive n th root of unity in F . Also let $1 \leq k \leq n$ be given. The *Reed–Solomon code* for these parameters has code word $(f_0, f_1, \dots, f_{n-1})$ if and only if $\alpha, \alpha^2, \dots, \alpha^{n-k}$ are roots of the polynomial

$$p(x) = f_0 + f_1x + \cdots + f_{n-1}x^{n-1},$$

With this definition, it is immediately seen that a Reed–Solomon code is a polynomial code, and in particular a BCH code. The generator polynomial $g(x)$ is the minimal polynomial with roots $\alpha, \alpha^2, \dots, \alpha^{n-k}$ as defined above, and the code words are exactly the polynomials that are divisible by $g(x)$.

Equivalence of the two formulations

At first sight, the above two definitions of Reed–Solomon codes seem very different. In the first definition, code words are *values* of polynomials, whereas in the second, they are *coefficients*. Moreover, the polynomials in the first definition are required to be of small degree, whereas those in the second definition are required to have specific roots.

The equivalence of the two definitions is proved using the discrete Fourier transform. This transform, which exists in all finite fields as well as the complex numbers, establishes a duality between the coefficients of polynomials and their values. This duality can be approximately summarized as follows: Let $p(x)$ and $q(x)$ be two polynomials of degree less than n . If the *values* of $p(x)$ are the *coefficients* of $q(x)$, then (up to a scalar factor and reordering), the *values* of $q(x)$ are the *coefficients* of $p(x)$. For this to make sense, the values must be taken at locations $x = \alpha^i$, for $i = 0, \dots, n - 1$, where α is a primitive n th root of unity.

To be more precise, let

$$p(x) = v_0 + v_1x + v_2x^2 + \dots + v_{n-1}x^{n-1},$$
$$q(x) = f_0 + f_1x + f_2x^2 + \dots + f_{n-1}x^{n-1}$$

and assume $p(x)$ and $q(x)$ are related by the discrete Fourier transform. Then the coefficients and values of $p(x)$ and $q(x)$ are related as follows: for all $i = 0, \dots, n - 1$, $f_i = p(\alpha^i)$ and $v_i = \frac{1}{n}q(\alpha^{n-i})$.

Using these facts, we have: (f_0, \dots, f_{n-1}) is a code word of the Reed–Solomon code according to the first definition

- if and only if $p(x)$ is of degree less than k (because f_0, \dots, f_{n-1} are the values of $p(x)$),
- if and only if $v_i = 0$ for $i = k, \dots, n - 1$,
- if and only if $q(\alpha^i) = 0$ for $i = 1, \dots, n - k$ (because $q(\alpha^i) = nv_{n-i}$),
- if and only if (f_0, \dots, f_{n-1}) is a code word of the Reed–Solomon code according to the second definition.

This shows that the two definitions are equivalent.

Remarks

Reed–Solomon codes are usually constructed as systematic codes. Instead of sending $s(x) = p(x)g(x)$, the encoder will construct the transmitted polynomial $s(x)$ such that it is evenly divisible by $g(x)$ and $p(x)$ is apparent in the codeword. Ordinarily, the construction is done by multiplying $p(x)$ by x^t to make room for the t check symbols, dividing that product by $g(x)$ to find the remainder, and then compensating for that remainder. In this case, the t check symbols are created by computing the remainder, $s_r(x)$:

$$s_r(x) = (p(x) \times x^t) \bmod g(x)$$

and that remainder is used to make an evenly divisible codeword:

$$s(x) = p(x) \times x^t - s_r(x)$$

with the result

$$s(x) \bmod g(x) = (p(x) \times x^t - s_r(x)) \bmod g(x) = s_r(x) - s_r(x) = 0$$

showing that $s(x)$ is a multiple of the generator polynomial $g(x)$.^[5]

Designers are not required to use the “natural” sizes of Reed–Solomon code blocks. A technique known as “shortening” can produce a smaller code of any desired size from a larger code. For example, the widely used (255,223) code can be converted to a (160,128) code by padding the unused portion of the source block with 95 binary zeroes and not transmitting them. At the decoder, the same portion of the block is loaded locally with binary zeroes. The Delsarte-Goethals-Seidel^[6] theorem illustrates an example of an application of shortened Reed–Solomon codes. In parallel to shortening, a technique known as puncturing allows omitting some of the encoded parity symbols.

Properties

The Reed–Solomon code is a $[n, k, n - k + 1]$ code; in other words, it is a linear block code of length n (over F) with dimension k and minimum Hamming distance $n - k + 1$. The Reed–Solomon code is optimal in the sense that the minimum distance has the maximum value possible for a linear code of size (n, k) ; this is known as the Singleton bound. Such a code is also called a maximum distance separable (MDS) code.

The error-correcting ability of a Reed–Solomon code is determined by its minimum distance, or equivalently, by $n - k$, the measure of redundancy in the block. If the locations of the error symbols are not known in advance, then a Reed–Solomon code can correct up to $(n - k) / 2$ erroneous symbols, i.e., it can correct half as many errors as there are redundant symbols added to the block. Sometimes error locations are known in advance (e.g., “side information” in demodulator signal-to-noise ratios)—these are called erasures. A Reed–Solomon code (like any MDS code) is able to correct twice as many erasures as errors, and any combination of errors and erasures can be corrected as long as the relation $2E + S \leq n - k$ is satisfied, where E is the number of errors and S is the number of erasures in the block.

For practical uses of Reed–Solomon codes, it is common to use a finite field F with 2^m elements. In this case, each symbol can be represented as an m -bit value. The sender sends the data points as encoded blocks, and the number of symbols in the encoded block is $n = 2^m - 1$. Thus a Reed–Solomon code operating on 8-bit symbols has $n = 2^8 - 1 = 255$ symbols per block. (This is a very popular value because of the prevalence of byte-oriented computer systems.) The number k , with $k < n$, of *data* symbols in the block is a design parameter. A commonly used code encodes $k = 223$ eight-bit data symbols plus 32 eight-bit parity symbols in an $n = 255$ -symbol block; this is denoted as a $(n, k) = (255, 223)$ code, and is capable of correcting up to 16 symbol errors per block.

The above properties of Reed–Solomon codes make them especially well-suited to applications where errors occur in bursts. This is because it does not matter to the code how many bits in a symbol are in error — if multiple bits in a symbol are corrupted it only counts as a single error. Conversely, if a data stream is not characterized by error bursts or drop-outs but by random single bit errors, a Reed–Solomon code is usually a poor choice compared to a binary code.

The Reed–Solomon code, like the convolutional code, is a transparent code. This means that if the channel symbols have been inverted somewhere along the line, the decoders will still operate. The result will be the inversion of the original data. However, the Reed–Solomon code loses its transparency when the code is shortened. The "missing" bits in a shortened code need to be filled by either zeros or ones, depending on whether the data is complemented or not. (To put it another way, if the symbols are inverted, then the zero-fill needs to be inverted to a one-fill.) For this reason it is mandatory that the sense of the data (i.e., true or complemented) be resolved before Reed–Solomon decoding.

Error correction algorithms

Theoretical decoder

Reed & Solomon (1960) described a theoretical decoder that corrected errors by finding the most popular message polynomial. The decoder for a RS (n,k) code would look at all possible subsets of k symbols from the set of n symbols that were received. For the code to be correctable in general, at least k symbols had to be received correctly, and k symbols are needed to interpolate the message polynomial. The decoder would interpolate a message polynomial for each subset, and it would keep track of the resulting polynomial candidates. The most popular message is the corrected result. Unfortunately, there are a lot of subsets, so the algorithm is impractical. The number of subsets is the binomial coefficient, $\binom{n}{k} = \frac{n!}{(n-k)!k!}$, and the number of subsets is infeasible for even modest codes. For a (255,249) code that can correct 3 errors, the naive theoretical decoder would examine 359 billion subsets. The RS code needed a practical decoder.

Peterson decoder

Main article: Peterson–Gorenstein–Zierler algorithm

Peterson (1960) developed a practical decoder based on syndrome decoding. (Welch 1997, p. 10) Berlekamp (below) would improve on that decoder.

Syndrome decoding

The transmitted message is viewed as the coefficients of a polynomial $s(x)$ that is divisible by a generator polynomial $g(x)$. Welch (1997, p. 5)

$$s(x) = \sum_{i=0}^{n-1} c_i x^i$$

$$g(x) = \prod_{j=1}^{n-k} (x - \alpha^j),$$

where α is a primitive root.

Since $s(x)$ is divisible by generator $g(x)$, it follows that

$$s(\alpha^i) = 0, \quad i = 1, 2, \dots, n - k$$

The transmitted polynomial is corrupted in transit by an error polynomial $e(x)$ to produce the received polynomial $r(x)$.

$$r(x) = s(x) + e(x)$$

$$e(x) = \sum_{i=0}^{n-1} e_i x^i$$

where e_i is the coefficient for the i -th power of x . Coefficient e_i will be zero if there is no error at that power of x and nonzero if there is an error. If there are ν errors at distinct powers i_k of x , then

$$e(x) = \sum_{k=1}^{\nu} e_{i_k} x^{i_k}$$

The goal of the decoder is to find ν , the positions i_k , and the error values at those positions.

The syndromes S_j are defined as

$$\begin{aligned} S_j &= r(\alpha^j) = s(\alpha^j) + e(\alpha^j) = 0 + e(\alpha^j) = e(\alpha^j), \quad j = 1, 2, \dots, n - k \\ &= \sum_{k=1}^{\nu} e_{i_k} (\alpha^j)^{i_k} \end{aligned}$$

The advantage of looking at the syndromes is that the message polynomial drops out.

Error locators and error values

For convenience, define the **error locators** X_k and **error values** Y_k as:

$$X_k = \alpha^{i_k}, \quad Y_k = e_{i_k}$$

Then the syndromes can be written in terms of the error locators and error values as

$$S_j = \sum_{k=1}^{\nu} Y_k X_k^j$$

The syndromes give a system of $n - k \geq 2\nu$ equations in 2ν unknowns, but that system of equations is nonlinear in the X_k and does not have an obvious solution. However, if the X_k were known (see below), then the syndrome equations provide a linear system of equations that can easily be solved for the Y_k error values.

$$\begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_\nu^1 \\ X_1^2 & X_2^2 & \cdots & X_\nu^2 \\ \vdots & \vdots & & \vdots \\ X_1^{n-k} & X_2^{n-k} & \cdots & X_\nu^{n-k} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_\nu \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_{n-k} \end{bmatrix}$$

Consequently, the problem is finding the X_k .

Error locator polynomial

Peterson found a linear recurrence relation that gave rise to a system of linear equations. (Welch 1997, p. 10) Solving those equations identifies the error locations.

Define the **error locator polynomial** $\Lambda(x)$ as

$$\Lambda(x) = \prod_{k=1}^{\nu} (1 - xX_k) = 1 + \Lambda_1 x^1 + \Lambda_2 x^2 + \cdots + \Lambda_\nu x^\nu$$

The zeros of $\Lambda(x)$ are the reciprocals X_k^{-1} :

$$\Lambda(X_k^{-1}) = 0$$

$$\Lambda(X_k^{-1}) = 1 + \Lambda_1 X_k^{-1} + \Lambda_2 X_k^{-2} + \cdots + \Lambda_\nu X_k^{-\nu} = 0$$

Multiply both sides by $Y_k X_k^{j+\nu}$ and it will still be zero.

$$Y_k X_k^{j+\nu} \Lambda(X_k^{-1}) = 0.$$

Hence $Y_k X_k^{j+\nu} + \Lambda_1 Y_k X_k^{j+\nu} X_k^{-1} + \Lambda_2 Y_k X_k^{j+\nu} X_k^{-2} + \cdots + \Lambda_\nu Y_k X_k^{j+\nu} X_k^{-\nu} = 0$,
and so $Y_k X_k^{j+\nu} + \Lambda_1 Y_k X_k^{j+\nu-1} + \Lambda_2 Y_k X_k^{j+\nu-2} + \cdots + \Lambda_\nu Y_k X_k^j = 0$

Sum for $k = 1$ to ν

$$\sum_{k=1}^{\nu} (Y_k X_k^{j+\nu} + \Lambda_1 Y_k X_k^{j+\nu-1} + \Lambda_2 Y_k X_k^{j+\nu-2} + \cdots + \Lambda_{\nu} Y_k X_k^j) = 0$$

$$\sum_{k=1}^{\nu} (Y_k X_k^{j+\nu}) + \Lambda_1 \sum_{k=1}^{\nu} (Y_k X_k^{j+\nu-1}) + \Lambda_2 \sum_{k=1}^{\nu} (Y_k X_k^{j+\nu-2}) + \cdots + \Lambda_{\nu} \sum_{k=1}^{\nu} (Y_k X_k^j) = 0$$

This reduces to

$$S_{j+\nu} + \Lambda_1 S_{j+\nu-1} + \cdots + \Lambda_{\nu-1} S_{j+1} + \Lambda_{\nu} S_j = 0$$

$$S_j \Lambda_{\nu} + S_{j+1} \Lambda_{\nu-1} + \cdots + S_{j+\nu-1} \Lambda_1 = -S_{j+\nu}$$

This yields a system of linear equations that can be solved for the coefficients Λ_i of the error location polynomial:

$$\begin{bmatrix} S_1 & S_2 & \cdots & S_{\nu} \\ S_2 & S_3 & \cdots & S_{\nu+1} \\ \vdots & \vdots & & \vdots \\ S_{\nu} & S_{\nu+1} & \cdots & S_{2\nu-1} \end{bmatrix} \begin{bmatrix} \Lambda_{\nu} \\ \Lambda_{\nu-1} \\ \vdots \\ \Lambda_1 \end{bmatrix} = \begin{bmatrix} -S_{\nu+1} \\ -S_{\nu+2} \\ \vdots \\ -S_{\nu+\nu} \end{bmatrix}$$

Obtain the error locations from the error locator polynomial

Use the coefficients Λ_i found in the last step to build the error location polynomial. The roots of the error location polynomial can be found by exhaustive search. The error locators (and hence the error locations) can be found from those roots.

Calculate the error values

Once the error locations are known, the error values can be determined and corrected.

Berlekamp–Massey decoder

Main article: BCH code

Transmit $s(x)$ divisible by generating polynomial $g(x)$.

Received $r(x)$ will have errors: $r(x) = s(x) + e(x)$.

The received polynomial is evaluated at the roots of $g(x)$; these values are the syndromes. Since $g(x)$ divides $s(x)$, $s(x)$ has no effect on the syndromes and is essentially eliminated. Only $e(x)$ affects the syndromes, and the goal becomes using the syndromes to determine $e(x)$ (which can be used to reconstruct $s(x) = r(x) - e(x)$).

The Berlekamp–Massey algorithm is used to find the error location polynomial.

The roots of the error location polynomial identify the nonzero terms of $e(x)$. The roots are usually found using Chien search.

Once the error locations are known, the error values can be found using Forney's algorithm.

With the locations and the values, the message is reconstructed.

Decoding in frequency domain (sketch)

The above algorithms are presented in the time domain. Decoding in the frequency domain, using Fourier transform techniques, can offer computational and implementation advantages. (Hong & Vetterli 1995)

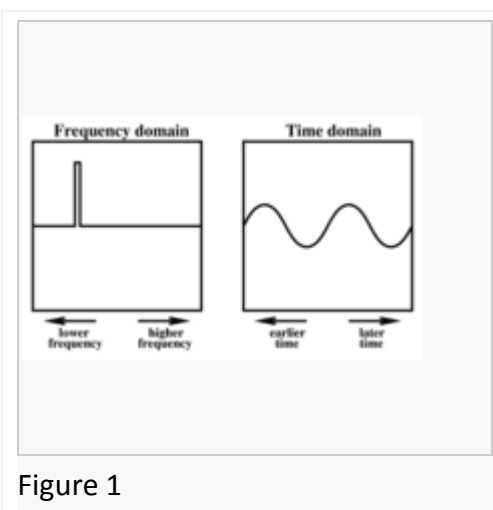
The following is a sketch of the main idea behind this error correction technique.

By definition, a code word of a Reed–Solomon code is given by the sequence of values of a low-degree polynomial over a finite field. A key fact for the error correction algorithm is that the *values* and the *coefficients* of a polynomial are related by the discrete Fourier transform.

The purpose of a Fourier transform is to convert a signal from a time domain to a frequency domain or vice versa. In case of the Fourier transform over a finite field, the frequency domain signal corresponds to the coefficients of a polynomial, and the time domain signal correspond to the values of the same polynomial.

As shown in Figures 1 and 2, an isolated value in the frequency domain corresponds to a smooth wave in the time domain. The wavelength depends on the location of the isolated value.

Conversely, as shown in Figures 3 and 4, an isolated value in the time domain corresponds to a smooth wave in the frequency domain.



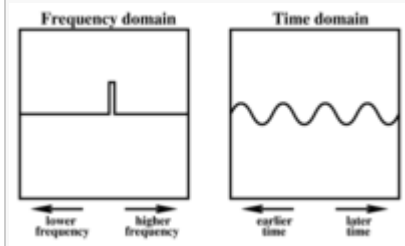


Figure 2

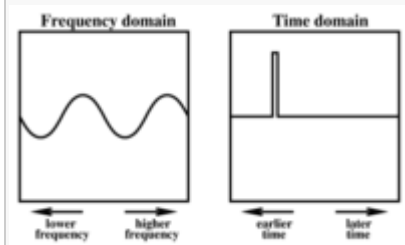


Figure 3

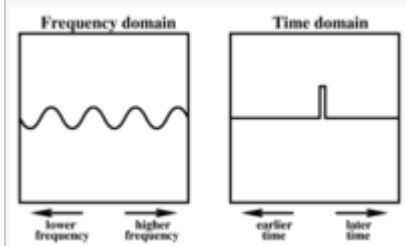


Figure 4

In a Reed–Solomon code, the frequency domain is divided into two regions as shown in Figure 5: a left (low-frequency) region of length k , and a right (high-frequency) region of length $n - k$. A data word is then embedded into the left region (corresponding to the k coefficients of a polynomial of degree at most $k - 1$), while the right region is filled with zeros. The result is Fourier transformed into the time domain, yielding a code word that is composed only of low frequencies. In the absence of errors, a code word can be decoded by reverse Fourier transforming it back into the frequency domain.

Now consider a code word containing a single error, as shown in red in Figure 6. The effect of this error in the frequency domain is a smooth, single-frequency wave in the right region, called the *syndrome* of the error. The error location can be determined by determining the frequency of the syndrome signal.

Similarly, if two or more errors are introduced in the code word, the syndrome will be a signal composed of two or more frequencies, as shown in Figure 7. As long as it is possible to determine the frequencies of which the syndrome is composed, it is possible to determine the error locations. Notice that the error *locations* depend only on the *frequencies* of these waves, whereas the error *magnitudes* depend on their amplitudes and phase.

The problem of determining the error locations has therefore been reduced to the problem of finding, given a sequence of $n - k$ values, the smallest set of elementary waves into which these values can be decomposed. It is known from digital signal processing that this problem is equivalent to finding the roots of the minimal polynomial of the sequence, or equivalently, of finding the shortest linear feedback shift register (LFSR) for the sequence. The latter problem can either be solved inefficiently by solving a system of linear equations, or more efficiently by the Berlekamp–Massey algorithm.

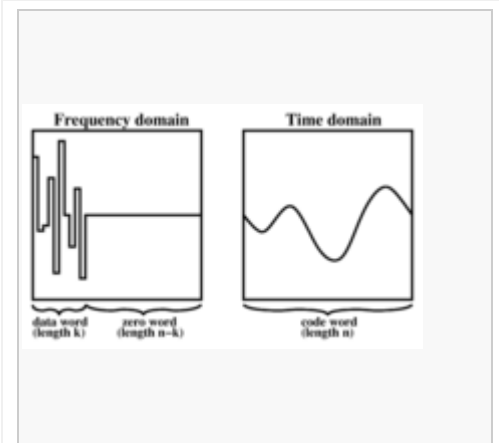


Figure 5

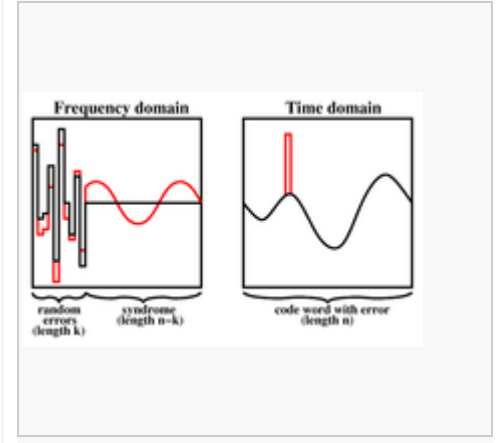


Figure 6

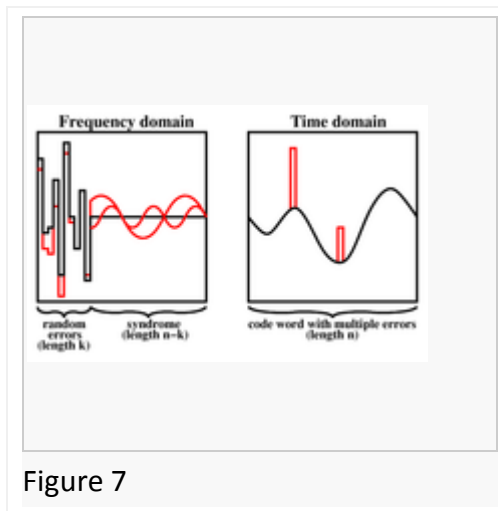


Figure 7

Decoding beyond the error-correction bound

The Singleton bound states that the minimum distance d of a linear block code of size (n, k) is upper-bounded by $n - k + 1$. The distance d was usually understood to limit the error-correction capability to $\lfloor d/2 \rfloor$. The Reed–Solomon code achieves this bound with equality, and can thus correct up to $\lfloor (n - k + 1)/2 \rfloor$ errors. However, this error-correction bound is not exact.

In 1999, Madhu Sudan and Venkatesan Guruswami at MIT published “Improved Decoding of Reed–Solomon and Algebraic-Geometry Codes” introducing an algorithm that allowed for the correction of errors beyond half the minimum distance of the code. It applies to Reed–Solomon codes and more generally to algebraic geometric codes. This algorithm produces a list of codewords (it is a list-decoding algorithm) and is based on interpolation and factorization of polynomials over $GF(2^m)$ and its extensions.

Soft-decoding

The algebraic decoding methods described above are hard-decision methods, which means that for every symbol a hard decision is made about its value.^[7] The advent of LDPC and turbo codes, which employ iterated soft-decision belief propagation decoding methods to achieve error-correction performance close to the theoretical limit, has spurred interest in applying soft-decision decoding to conventional algebraic codes. In 2003, Ralf Koetter and Alexander Vardy presented a polynomial-time soft-decision algebraic list-decoding algorithm for RS codes, which was based upon the work by Sudan and Guruswami.^[8]

Applications

Data storage

Reed–Solomon coding is very widely used in mass storage systems to correct the burst errors associated with media defects.

Reed–Solomon coding is a key component of the compact disc. It was the first use of strong error correction coding in a mass-produced consumer product, and DAT and DVD use similar schemes. In the CD, two layers of Reed–Solomon coding separated by a 28-way convolutional interleaver yields a scheme called Cross-Interleaved Reed Solomon Coding

(CIRC). The first element of a CIRC decoder is a relatively weak inner (32,28) Reed–Solomon code, shortened from a (255,251) code with 8-bit symbols. This code can correct up to 2 byte errors per 32-byte block. More importantly, it flags as erasures any uncorrectable blocks, i.e., blocks with more than 2 byte errors. The decoded 28-byte blocks, with erasure indications, are then spread by the deinterleaver to different blocks of the (28,24) outer code. Thanks to the deinterleaving, an erased 28-byte block from the inner code becomes a single erased byte in each of 28 outer code blocks. The outer code easily corrects this, since it can handle up to 4 such erasures per block.

The result is a CIRC that can completely correct error bursts up to 4000 bits, or about 2.5 mm on the disc surface. This code is so strong that most CD playback errors are almost certainly caused by tracking errors that cause the laser to jump track, not by uncorrectable error bursts.^[9]

Another product which incorporates Reed–Solomon coding is the Nintendo e-Reader. This is a video-game delivery system which uses a two-dimensional barcode printed on trading cards. The cards are scanned using a device which attaches to Nintendo's Game Boy Advance game system.

Reed–Solomon error correction is also used in archive files which are commonly posted accompanying multimedia files on USENET. The Distributed online storage service Wuala also makes use of Reed–Solomon when breaking up files.

Data transmission

Specialized forms of Reed–Solomon codes, specifically Cauchy-RS and Vandermonde-RS, can be used to overcome the unreliable nature of data transmission over erasure channels. The encoding process assumes a code of $RS(N, K)$ which results in N codewords of length N symbols each storing K symbols of data, being generated, that are then sent over an erasure channel.

Any combination of K codewords received at the other end is enough to reconstruct all of the N codewords. The code rate is generally set to $1/2$ unless the channel's erasure likelihood can be adequately modelled and is seen to be less. In conclusion, N is usually $2K$, meaning that at least half of all the codewords sent must be received in order to reconstruct all of the codewords sent.

Reed–Solomon codes are also used in xDSL systems and CCSDS's Space Communications Protocol Specifications as a form of forward error correction.

Bar code

Paper bar codes such as PDF-417, PostBar, MaxiCode, Datamatrix, QR Code, and Aztec Code use Reed–Solomon error correction to allow correct reading even if a portion of the bar code is damaged. When the bar code scanner cannot recognize a bar code symbol, it will treat it as an erasure.

Satellite transmission

One significant application of Reed–Solomon coding was to encode the digital pictures sent back by the Voyager space probe.

Voyager introduced Reed–Solomon coding concatenated with convolutional codes, a practice that has since become very widespread in deep space and satellite (e.g., direct digital broadcasting) communications.

Viterbi decoders tend to produce errors in short bursts. Correcting these burst errors is a job best done by short or simplified Reed–Solomon codes.

Modern versions of concatenated Reed–Solomon/Viterbi-decoded convolutional coding were and are used on the Mars Pathfinder, Galileo, Mars Exploration Rover and Cassini missions, where they perform within about 1–1.5 dB of the ultimate limit imposed by the Shannon capacity.