UNIVERSITE LUMIERE DE BUJUMBURA

Faculté des Sciences et Technologies

informatique de Gestion:BAC+3

Campus MUTANGA

Cours: Théorie de l'information

Msc. MBAZUMUTIMA Khalfan

Contents

1 Introduction aux systèmes de communication

1.1 Introduction

La théorie des communications s'intéresse aux moyens de transmettre une information depuis la source jusqu'à un utilisateur à travers un canal. La nature de la source peut être très variée. Il peut s'agir par exemple d'une voix, d'un signal électromagnétique ou d'une séquence de symboles binaires. Le canal peut être une ligne téléphonique, une liaison radio, un support magnétique ou optique. La transmission peut se faire dans l'espace ou dans le temps. Le codeur représente l'ensemble des opérations effectuées sur la sortie de la source avant la transmission. Ces opérations peuvent être par exemple la modulation, la compression, le brouillage, l'ajout de redondance pour combattre les effets du bruit, ou encore l'adaptation à des contraintes de spectre. Elles ont pour but de rendre la sortie de la source compatible avec le canal. Enfin le décodeur doit être capable, à partir de la sortie du canal, de restituer de façon acceptable l'information fournie par la source.

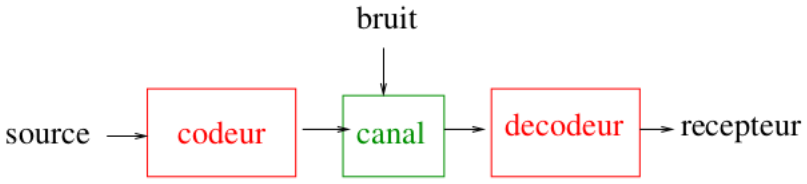


Figure 1.1: Un système de communication

La théorie de l'information a été créée par C. E. Shannon dans les années 40. Il s'agit d'une théorie mathématique qui décrit les aspects les plus fondamen-

taux des systèmes de communication. Elle consiste en l'élaboration et l'étude de modèles pour la source et le canal qui utilisent différents outils comme les probabilités et les automates finis.
Dans ce cours, nous étudieront certains de ces modèles qui, bien que considérablement plus simples que les sources et les canaux physiques, permettent de donner une bonne approximation de leur comportement.
Pour simplifier, on étudiera séparément les modèles de sources et les modèles de canaux ainsi que leurs codages respectifs.
– Le but du codeur de source est de représenter la sortie de la source en une séquence binaire, et cela de façon la plus économique possible. – Le but du codeur de canal et de son décodeur est de reproduire le plus fidèlement possible cette séquence binaire malgré le passage à travers le canal bruité.

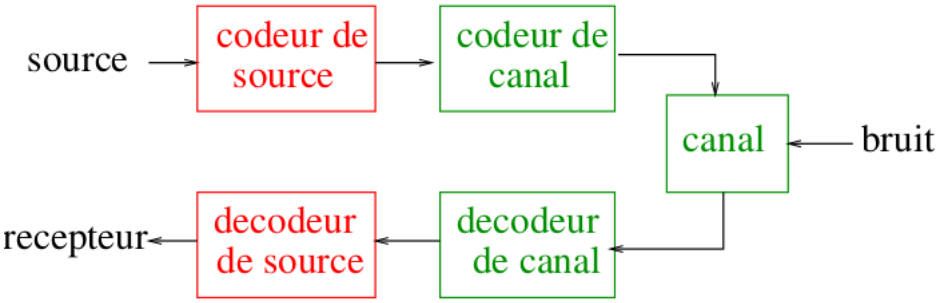


Figure 1.2: Codeur de source et codeur de canal

Cette séparation entre codage de source et codage de canal n'implique en pratique aucune limitation sur les performances du système complet.

1.1.1 Sources et codage de source

Parmi les classes possibles de modèles de source, nous nous intéresserons plus particulièrement aux sources discrètes sans mémoire.
La sortie d'une telle source est une séquence de lettres tirées dans un alphabet fini $A = \{a_1, ..., a_n\}$. Chaque lettre de la séquence est choisie aléatoirement d'après une loi de probabilité p indépendante du temps.Pour toute lettre a, $p(a)$ est la probabilité pour que cette lettre soit choisie. Il s'agit d'un réel compris entre 0 et 1. On a $\sum_{a \in A} p(a) = 1$. La donnée de $p(a_1), ..., p(a_n)$ définit la probabilité discrète p sur A. Il peut sembler étonnant de modéliser une source d'information à l'aide d'une variable aléatoire. Nous allons donner un exemple qui permet de se convaincre de l'utilité de tels modèles.

Exemple : Soit une source d'information qui fournit comme information l'une des quatre lettres a_1, a_2, a_3, a_4 . Supposons que le codage de source transforme

cette information discrète en symboles binaires. Nous donnons deux exemples de codage différents.

Codage 1	Codage 2
$a_1 \to 00$	$a_1 \to 0$
$a_2 \to 01$	$a_2 \to 10$
$a_3 \to 10$	$a_3 \to 110$
$a_3 \to 11$	$a_3 \to 111$

Si les quatre lettres sont équiprobables, la première méthode de codage est meilleure. Elle nécessite en effet deux symboles par lettre en moyenne tandis que la deuxième méthode nécessite $\frac{1}{4} + 2 * \frac{1}{4} + 3 * \frac{1}{4} + 3 * \frac{1}{4} = 2,25$ symboles par lettres. En revanche, si l'on a une source dont la distribution de probabilités est

$$p\left(a_1\right) = \tfrac{1}{2}, p\left(a_2\right) = \tfrac{1}{4}, p\left(a_3\right) = p\left(a_4\right) = \tfrac{1}{8}$$

la longueur moyenne d'un symbole codé par la première méthode est toujours 2 tandis que celle d'un symbole codé par la deuxième méthode est
$\frac{1}{2} + 2 * \frac{1}{4} + 3 * \frac{1}{4} + 3 * \frac{1}{8} = 1,75$.
Le deuxième codage réussit donc à coder quatre symboles avec moins de deux bits. Il a réalisé une compression. Pour coder correctement une source, il est donc important de connaître son comportement statistique.

1.1.2 Entropie d'une source discrète

Nous allons établir un lien entre l'information fournie par une source et la distribution de probabilité de la sortie de cette source. On considère en effet que l'apparition d'un événement peu probable apporte beaucoup d'information tandis que l'occurrence d'un événement certain ne fournit au contraire aucune information.
Si une lettre a a une probabilité $p(a)$ d'être tirée, son information propre est définie par

$$I\left(a\right) = -\log_2 p\left(a\right)$$

En particulier $I(a)$ vaut zéro si $p(a) = 1$.
La valeur moyenne de l'information propre calculée sur l'ensemble de l'alphabet revêt une grande importance. Il s'agit donc de l'espérance de la variable aléatoire I. Elle est appelée entropie de la source et est notée $H(A)$:

$$H\left(a\right) = -\sum_{a \in A} p\left(a\right) \log_2 p\left(a\right)$$

Si une source émet n lettres équiprobables (ou encore avec une loi de probabilité uniforme), son entropie est donc $\log_2(a)$. Si $n = 2^r$, son entropie est alors r. Or pour représenter 2^r lettres distinctes en binaires, r cases sont nécessaires. L'entropie d'une source est quelquefois donnée en bits/seconde. Si l'entropie d'une source discrète est H et si les lettres sont émises toutes les τs secondes, son entropie en bits/s est $H/\tau s$.

1.1.3 Autres modèles de source

On peut également distinguer, parmi les classes de modèles de sources, les sources discrètes avec mémoire, finie ou infinie. Une entropie peut être définie pour ces sources de façon analogue.
Enfin les sources non discrètes, ou sources continues, ont une grande importance dans les applications. La sortie d'une telle source sera une fonction continue du temps, par exemple une tension qu'il faut coder par une séquence discrète binaire. La fonction continue doit être décrite le plus fidèlement possible par la séquence binaire générée par le codeur de source. Le problème dans ce cas consiste à minimiser le nombre de symboles transmis pour un niveau de distorsion donné.

1.1.4 Canaux et codage de canal

Pour modéliser un canal de transmission, il est nécessaire de spécifier l'ensemble des entrées et l'ensemble des sorties possibles. La cas le plus simple est celui du canal discret sans mémoire. L'entrée est une lettre prise dans un alphabet fini $A = \{a_1, ..., a_n\}$ et la sortie est est une lettre prise dans un alphabet fini $B = \{b_1, ..., b_m\}$. Ces lettres sont émises en séquence, et, le canal est sans mémoire si chaque lettre de la séquence reçue ne dépend statistiquement que de la lettre émise de même position.
Ainsi un canal discret sans mémoire est entièrement décrit par la donnée des probabilités conditionnelles $p(b|a)$ pour toutes les lettres a de l'alphabet d'entrée et toutes les lettres b de l'alphabet de sortie. Nous allons revenir dans la section suivante sur cette notion de probabilité conditionnelle.

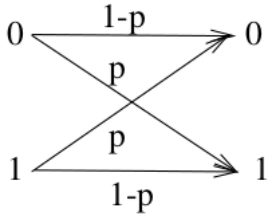


Figure 1.3: Le canal binaire symétrique

Exemple canal discret sans mémoire : Le plus connu est le canal binaire symétrique défini par $A = B = \{0, 1\}$ et dont les probabilités de transition sont représentées Figure 1.4. La probabilité pour qu'un symbole soit inchangé est $1 - p$, où p est un réel compris entre 0 et 1, et la probabilité pour qu'il soit changé est p.

On peut également considérer des canaux discrets à mémoire dans lesquels chaque lettre de la séquence de sortie peut dépendre de plusieurs lettres de la séquence d'entrée.

1.1.5 Canaux continus

Il existe une classe de modèles de canaux appelés canaux continus, beaucoup plus proches des canaux physiques. L'entrée et la sortie sont alors des fonctions continues du temps. Pour les canaux de cette classe, il est commode de séparer le codeur et le décodeur en deux parties, comme le montre la figure 4. La première partie du codeur, que nous appellerons codeur de canal discret, transforme une séquence binaire en une séquence de lettres d'un alphabet fini $A = \{a_1, ..., a_n\}$. La seconde partie du codeur, le modulateur de données digitales, envoie pendant un temps τ_c sur le canal une des fonctions de temps prédéfinies $s_1(t), ..., s_n(t)$. La durée τ_c est l'intervalle de temps séparant l'émission de deux lettres par le codeur de canal discret. L'ensemble de ces fonctions du temps mises bout à bout est converti à la sortie du canal par le démodulateur de données digitales en une séquence de lettres d'un alphabet de sortie $B = \{b_1, ..., b_m\}$ au rythme, là encore, d'une lettre toutes les τ_c secondes.

1.2 Mesure de l'information

Nous allons donner une mesure de la quantité d'information qui est adaptée à la description statistique des sources et des canaux. Les énoncés qui en résultent font appel aux probabilités discrètes. Nous allons en rappeler les notions principales.

1.2.1 Espace probabilisé discret

Nous considérerons des ensembles finis munis d'une probabilité discrète p. L'espace probabilisé est noté (A, p). La loi de probabilité est dite uniforme si $p(a) = \frac{1}{n}$, où $n = card(A)$, pour toute lettre a de A. Une variable aléatoire de (A, p) est une fonction de A dans un ensemble quelconque. Une variable aléatoire est réelle si l'espace d'arrivée est $\mathbb{R}$. L'espérance d'une variable aléatoire réelle v est le réel encore appelé moyenne de v.

$$E(v) = \sum_{a \in A} p(a)v(a)$$

1.2.2 Espace probabilisé joint. Probabilités conditionnelles

Pour modéliser un canal discret, nous considérons l'espace AxB produit des deux ensembles $A = \{a_1, ..., a_n\}$ et $B = \{b_1, ..., b_m\}$. Le produit est formé des couples (a, b) avec a dans A et b dans B. On munit cet ensemble d'une loi de probabilité discrète, notée p_{AB} , appelée loi de probabilité jointe de A et B. L'espace de probabilité joint est aussi noté AB.
La probabilité $p_{AB}(a, b)$ est la probabilité d'avoir simultanément a en entrée et b en sortie. On définit une loi de probabilité p_A sur A par

$$p_A(a) = \sum_{b \in B} p_{AB}(a, b)$$

On vérifie que c'est bien une loi de probabilité sur A. On définit une probabilité p_B sur B de façon similaire. Les deux lois p_A et p_B sont appelées lois marginales.

Nous définissons maintenant les lois conditionnelles. Soit a une lettre de A telle que $p(a) > 0$. La probabilité conditionnelle pour que l'on ait b en sortie sachant que l'on a a en entrée est définie par

$$p_{B|A}(b|a) = \frac{p_{AB}(a,b)}{p_A(a)}$$

On dit également qu'il s'agit de la probabilité conditionnelle pour que l'on ait $\{B = b\}$ sachant que $\{A = a\}$. Notez ici l'abus de notation car A et Bdésignent ici des variables aléatoires. De façon symétrique, on a

$$p_{A|B}(a|b) = \frac{p_{AB}(a,b)}{p_B(b)}$$

On dit que les événements $\{B = b\}$ et $\{A = a\}$ sont statistiquement indépendants si $p_{A|B}(a, b) = p_A(a)p_B(b)$. Lorsque cette égalité est vraie pour tout couple AB, alors les espaces A et B sont dits statistiquement indépendants. On parle alors d'espace probabilisé produit. Lorsqu'il n'y aura pas de confusion on notera p toutes les probabilités ci-dessus. Ainsi on notera $p(b|a)$ la probabilité conditionnelle pour que l'on ait $\{B = b\}$ sachant que $\{A = a\}$, et $p(a|b)$ la probabilité conditionnelle pour que l'on ait $\{B = b\}$ sachant que $\{A = a\}$. Attention à ces notations car par exemple si $A = B = 0, 1, p_{A|B}(0|1)$ peut très bien être différent de $p_{B|A}(0|1)$.

1.2.3 Incertitude et information

La notion d'information est déjà inhérente à celle de probabilité conditionnelle. Considérons les événements $\{A = a\}$ et $\{B = b\}$. La probabilité $p(a|b)$ peut être interprétée comme la modification apportée à la probabilité $p(a)$ de l'événement $\{A = a\}$ lorsque l'on reçoit l'information que l'événement $\{B = b\}$ s'est réalisé. Ainsi

– si $p(a|b) \leq p(a)$, l'incertitude sur a augmente,
– si $p(a|b) \geq p(a)$, l'incertitude sur a diminue.

On notera $I(a)$ l'incertitude sur a, encore appelée information propre de a :
$I(a) = -\log_2 p(a)$.

Ainsi l'information "b est réalisé" diminue l'incertitude sur a de la quantité :

$$I(a) - I(a|b) = \log_2 \frac{p(a|b)}{p(a)} \rightarrow \text{Cette quantité est appelée information mutuelle de } a \text{ et } b.$$

1.2.4 Information mutuelle. Information propre

On considère un espace probabilisé joint AB où $\{A = a_1, ..., a_n\}$ et $\{B = b_1, ..., b_m\}$.
L'information mutuelle entre les événements $\{A = a\}$ et $\{B = b\}$ est définie par

$$I(a; b) = \log_2 \frac{p(a|b)}{p(a)}$$

Par définition $p(a, b) = p(a|b)p(b) = p(b|a)p(a)$. Donc

$$I(a; b) = I(b; a) = \log_2 \frac{p(a,b)}{p(a)p(b)}$$

Nous allons discuter le signe de $I(a; b)$.
– $I(a; b) > 0$ signifie que si l'un des deux événements se réalise, alors la probabilité de l'autre augmente ;
– $I(a; b) < 0$ signifie que si l'un des deux événements se réalise, alors la probabilité de l'autre diminue ;
– $I(a; b) = 0$ signifie que les deux événements sont statistiquement indépendants.

Exemple: Considérons le canal binaire symétrique de probabilité de transition p avec des entrées notées a_1 , a_2 équiprobables et des sorties b_1 , b_2 .

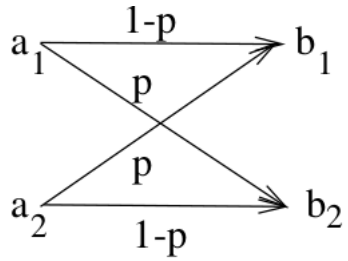


Figure 1.4: Le canal binaire symétrique

La matrice de transition, notée $\prod = (\prod_{ij})$, est définie par $\prod_{ij} = p(b_j|a_i)$.

La matrice de transition du canal binaire symétrique est donc

$$\begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$$

Puisque les entrées sont équiprobables, $p(a_1) = p(a_2) = \frac{1}{2}$. On en déduit la loi jointe :

$$p(a_1; b_1) = p(a_2; b_2) = \frac{1-p}{2}$$

$$p(a_1; b_2) = p(a_2; b_1) = \frac{p}{2}$$

On en déduit la loi marginale sur B : $p(b_1) = p(b_2) = \frac{1}{2}$. Ceci permet de calculer l'information mutuelle de chaque couple (a_i, b_j).

$$I(a_1; b_1) = I(a_2; b_2) = \log_2 2(1-p) = 1 + \log_2(1-p)$$

$$I(a_1; b_2) = I(a_2; b_1) = log_2 2p = 1 + log_2 p.$$

1.2.5 Information mutuelle moyenne. Entropie

L'information mutuelle moyenne de A et B dans l'espace probabilisé joint AB est définie par :

$$I(A; B) = \sum_{a \in A, b \in B} p(a, b) I(a; b)$$

donc

$$I(A; B) = \sum_{a \in A, b \in B} p(a, b) \log_2 \frac{p(a,b)}{p(a)p(b)}$$

On peut également définir la moyenne de l'information propre d'un espace probabilisé A. Cette moyenne s'appelle entropie de l'espace A :

$$H(a) = \sum_{a \in A} p(a) I(a) = -\sum_{a \in A} p(a) \log_2 p(a)$$

Enfin l'information propre conditionnelle est une variable aléatoire réelle et nous pouvons définir sa moyenne appelée entropie conditionnelle de A sachant B. Elle est définie sur l'espace de probabilité joint AB :

$$H(A|B) = -\sum_{a \in A, b \in B} p(a, b) \log_2 p(a|b)$$

On en déduit que $I(A; B) = H(A) - H(A|B)$.

2 Codage correcteur d'erreur

Les codes correcteurs ont été introduits pour corriger les erreurs de transmission ou de lecture de données numériques, ou les erreurs survenant au cours de leur inscription sur un support physique (bande, CD) ou encore lorsque les données subissent une altération sur le support de stockage. Voici quelques domaines ou ils sont appliques :

- transmissions spatiales ;
- minitel ;
- codes barres ;
- disque compact et DVD ;
- communications par internet.

Par codes, on peut entendre plusieurs concepts bien distincts : cryptographie (RSA,...) ; codes de compression (Huffman,...) ; codes correcteurs d'erreurs. Dans ce cours, on s'intéresse aux codes correcteurs d'erreur ; plus précisément à la famille des codes en bloc.

Lorsqu'on envoie un message à travers un canal de transmission des données (par exemple : en téléchargeant ce cours sur internet), des erreurs de transmission peuvent se produire. Le but est d'arriver à détecter, voire corriger des erreurs.

On se propose de "coder" chaque bloc du message initial en un bloc plus gros (avec des redondances d'information).

Exemple Code par adjonction d'un bit de parité (8, 9)
On découpe notre message initial en blocs de 8 bits.
On transforme ensuite chaque bloc en un bloc de 9 bits en ajoutant un bit à la fin de chaque bloc de telle sorte que la somme des bits des nouveaux blocs soit toujours paire.

Les trois principaux paramètres d'un code:

1. Dimension et longueur d'un code

   Terminologie et notations préliminaires :
   Un bloc de k bits sera indifféremment appelé **bloc**, **mot** ou **vecteur**. L'ensemble des mots de k bits sera noté $\{0,1\}^k$. On parlera indifféremment de **bits** ou de **lettres**.

   Un mot m de k bits sera noté $m_1m_2...m_k$, ou éventuellement

   $$\begin{pmatrix} m_1 \\ m_2 \\ ... \\ m_k \end{pmatrix}$$

   Le principe du codage est le suivant : après avoir découpé notre message en blocs de k bits, on va appliquer un même algorithme sur chaque bloc :
   a) ou bien en rajoutant des bits de contrôle à la fin de chaque bloc
   b) ou bien en modifiant complètement les blocs, mais en évitant que deux blocs différents soient transformés en un même bloc.

   D'où: Un code est une application injective ϕ: $\{0,1\}^k \to \{0,1\}^n$
   Le paramêtre k est appelé la dimension du code ϕ et le paramêtre n est appelé la longueur du code : on dit que ϕ est un code de paramètres (k, n).
   Si de plus pour tout mot m de $\{0,1\}^k$,m est un préfixe de $\phi(m)$ (c'est à dire si l'application de ϕ consiste seulement à rajouter des bits de contrôle), on dira que ϕ est un code systématique.

2. L'algorithme de codage ou décodage
   L'algorithme de codage ou de décodage doit être suffisamment rapide.

3. La distance minimale
   La distance minimale d'un code quantifie donc sa qualité vis à vis du point 1. C'est un paramètre important. En abrégé, "un code de dimension k, de longueur n et de distance minimale d" se dira "un code de paramètres (k, n, d)" ou même "un code(k, n, d)".

   Exemple Prenons l'exemple d'un code de répétition pure $(1, 3)$. Son image est $C = 000, 111$ donc sa distance minimale est $d(000, 111) = 3$

Note:

- La notation $[n, k, d_{min}]$ sera utilisée pour dénoter les paramètres d'un code en bloc de taille n, qui code k bits et poss'ede une distance minimale d_{min}.

- Le taux du code (rendement) est $\frac{n}{k}$ c'est à dire le nombre de bits d'information par bits codés.

Principe général :

- Chaque suite de bits (trame) à transmettre est augmentée par une autre suite de bit dite de redondance ou de contrôle.

- Pour chaque suite de k bits transmis, on ajoute r bits. On dit alors que l'on utilise un code $C(n, k)$ avec $n = k + r$.

- À la réception, on effectue l'opération inverse et les bits ajoutés permettent d'effectuer des contrôles à l'arrivée.

Il existe deux catégories de code :

- les codes détecteurs d'erreurs,

  Le CRC (Cycle Redundancy Check)

- Les codes détecteur et correcteur d'erreurs.

  Le code de Hamming

2.1 Codes et distance de Hamming

Les messages transmis sont supposés découpés en blocs (ou mots) de longueur n écrits avec l'alphabet $\{0, 1\}^n$. Un code(binaire) est un sous-ensemble C de l'ensemble$\{0, 1\}^n$ de tous les mots possibles. On dit que n est la longueur de C. La distance de Hamming entre deux mots $x = (x_1, ..., x_n)$ et $y = (y1, ..., yn)$, que l'on notera $d(x, y)$, est le nombre d'indices i tels que $x_i \neq y_i$. C'est bien une distance sur $\{0, 1\}^n$. La distance minimum du code C est le minimum des $d(x, y)$ pour x et y des mots différents de C (on suppose que C a au moins 2 mots !). On la notera toujours d

Définitions

La **distance de Hamming**, dans le cas binaire ($\mathbb{F}_2$) entre deux vecteurs x et y de dimension n correspond au nombre de composantes pour lequel ces deux vecteurs différent.

$$d(x, y) = |\{i : x_i \neq y_i, 0 \leq i \leq n\}|$$

Soit un code C, sa distance minimale de Hamming, d_{min}, est définie comme la distance minimum entre toutes les paires de mots de code de C.

Un code de distance minimale d_{min} est susceptible de corriger $t = [(dmin - 1)/2]$ erreurs. Plus précisément, si le mot y reçu après transmission comporte au plus t composantes erronées, il est possible de déterminer sans ambiguïte le mot de code émis c.

Exemple: Codage

$$0 \rightarrow 000$$

$$1 \rightarrow 111$$

- La distance minimum est $d_{min} = 3$,
- Le nombre d'erreur maximum détectable est de $d_{min} - 1 = 2$ erreurs.
- Le nombre d'erreur maximum corrigeable est de $t = [(d_{min} - 1)/2] = 1$ erreur.

2.1.1 Structure d'un mode de code de Hamming

- les m bits du message à transmettre et les n bits de contrôle de parité.
- longueur totale :$2^n - 1$
- longueur du messages : $m = (2^n - 1) - n$
  $\Rightarrow$ on parle de code (x, y) où $x = n + m$ et $y = m$.
- Le codage de Hamming se base sur le calcul de parité,
- Au lieu de rajouter un seul bit,rajouter plusieurs bit de parité,
- Chaque bit de Contrôle est une fonction de plusieurs bits d'information.

2.1.2 Principe du code de Hamming

- Soit un mot de code (7,4)→rajouter 3 bits de contrôles notes $C_0C_1C_2$
- et l'information de départ est sur 4 bits $m = U_0U_1U_2U_3$
- les bits de contrôles sont insérés dans l'information de la façon suivante :

  Ils prennent les positions 2^i pour $i \in \{0, 2, 3, ...\}$

  et les bits d'informations prennent les autres positions

C_0	C_1	U_0	C_2	U_1	U_2	U_3
2^0	2^1	3	2^2	5	6	7
1	2		4			

2.1.3 Calcul des bits de controle

- Chaque bit de l'information possède une position dans le mot de code final,
- Écrire cette position en puissance de 2

Exemple: U_0 est dans la position $3 = 1 + 2 = 2^0 + 2^1$
U_1 est dans la position $5 = 1 + 4 = 2^0 + 2^2$
U_2 est dans la position $6 = 2 + 4 = 2^1 + 2^2$
U_3 est dans la position $7 = 1 + 2 + 4 = 2^0 + 2^1 + 2^2$

- Un bit de l'information ayant la position J participe au calcul du bit de contrôle ayant la position 2^i si 2^i existe dans la decompostion de J en puissance de 2,
- Dans l'exemple précédant:

  La position de C_0 est 2^0 →donc les bits U_0, U_1, U_3 participent dans le calcul de C_0 →$C_0 = U_0 + U_1 + U_3$

  La position de C_1 est 2^1 →donc les bits U_0, U_2, U_3 participent dans le calcul de C_1 → $C_1 = U_0 + U_2 + U_3$

  La position de C_2 est 2^2 →donc les bits U_2, U_2, U_3 participent dans le calcul de C_2 → $C_2 = U_1 + U_2 + U_3$
- **Application:** Pour l'information 1010,trouver le code de hamming correspond

  $C_0 = U_0 + U_1 + U_3 = 1 + 0 + 0 = 1$

  $C_1 = U_0 + U_2 + U_3 = 1 + 1 + 0 = 0$

  $C_2 = U_1 + U_2 + U_3 = 0 + 1 + 0 = 1$

- Donc l'information a envoyé:1011010

2.1.4 Détection d'erreur

- A la réception du message,recalculer les bits de contrôle de la même manière que lors de l'émission.
- Si égalité alors passage au bit suivant
- Si non incrémenter un compteur C par la position du bit de contrôle
- Après avoir recalculer tous les bits de contrôle:

  Si le compteur est égale à zéro alors pas d'erreur

  Sinon il indique le numéro du bit erroné
- **Exemple:**Si on envoie le message 10110101 et on reçoit le message 1011000,normalement il y a une erreur.
- Pour la détecter:

  $C'_0 = U_0 + U_1 + U_3 = 1 + 0 + 0 = 1$ correcte,$C = 0$

  $C'_1 = U_0 + U_2 + U_3 = 1 + 0 + 0 = 1$,erreur $C = 2$

  $C'_2 = U_2 + U_2 + U_3 = 0 + 0 + 0 = 0$,erreur $C = 2 + 4$

Donc le bit errone est le bit N^0 6

Exercices 1: Est ce qu'il y a une erreur dans le mot suivant:0110101?

Exercice 2 Soit un code de hamming sur 15 bits 110110111101101.

- Quels sont les bits de controle?
- Quel est le message recu?
- Est ce le message recu corresponr au message transmis?

  Sinon quel est le message transmis?

2.1.5 Émission pour un contrôle de parité pair

On souhaite envoyer le message 1010, compléter le mot de Hamming correspondant :

1	0	1	_	1	_	_
7	6	5	4	3	2	1

1	0	1	$\underline{0}$	1	-	-

- C2 vaut 0 pour pouvoir rendre pair 1 + 0 + 1 (les bits d'indices 7, 6, 5),

- C_1 vaut 1 pour pouvoir rendre pair 1 + 0 + 0 (les bits d'indices 7, 6, 3),

1	0	1	$\underline{0}$	1	$\underline{1}$	-

- C_0 vaut 0 pour pouvoir rendre pair 1 + 1 + 0 (les bits d'indice 7, 5, 3),

1	0	1	$\underline{0}$	1	$\underline{1}$	$\underline{0}$

2.2 Le CRC

2.2.1 Vérification polynomiale

Rappel: Une information en binaire peut être écrit sous la forme polynomial suivant les puissances de 2

$(1110)_2 = 1 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0$

Dans le cas général:

$(u_k u_{k-1}.....u_1 u_0) = u_{k-1}X^k + u_k X^{k-1} + ... + u_1 X^1 + u_0 X^0$ avec $u_i \in [0, 1]$

Exemple: La suite 1100101 est représenté par le polynôme:

$1100101 = 1.X^6 + 1.X^5 + 0.X^4 + 0.X^3 + 1.X^2 + 0.X^1 + 1.X^0$

2.2.2 Le calcul du CRC

- On choisit un un polynôme appelé polynôme générateur $G(X)$ de degré n
  Exemple:$x^4 + x^2 + x \rightarrow$ Polynôme générateur de degré 4
- Soit une information sur m bits représentée sous la forme d'un polynôme $M(X)$ de degré $(m - 1)$

- Pour calculer le CRC:
  - multiplier $M(X)$ par X^n(n est le degré du polynôme générateur)
  - effectuer une division de $X^n.M(X) par G(X)$,
  - on obtient le quotient $Q(X)$ et le reste $R(X)$
    $X^n M(X) = Q(X)G(X) + R(X)$
  - Le CRC correspond au reste de la division $R(X)$
- Donc l'information a envoye est egale à $X^n.M(X).R(X)$

Calcul du CRC avec des additions successives

- On choisit un polynôme générateur puis on le transforme en un mot binaire.
- Exemple : avec le polynôme générateur $x^4 + x^2 + x$, on obtient 10110
- On ajoute m zéros au mot binaire à transmettre où m est le degré du polynôme générateur.
- Exemple : on souhaite transmettre le mot 11100111 en utilisant le polynôme générateur $x^4 + x^2 + x$, on obtient alors 111001110000.
- On va ajouter itérativement à ce mot, le mot correspondant au polynôme générateur jusqu'à ce que le mot obtenu soit inférieur au polynôme générateur. Ce mot obtenu correspond au CRC à ajouter au mot avant de l'émettre.
- On effectue donc une division euclidienne dans laquelle on ne tient pas compte du quotient.

1	1	1	0	0	1	1	1	0	0	0	0
1	0	1	1	0							
0	1	0	1	0	1						
	1	0	1	1	0						
	0	0	0	1	1	1	1	0			
				1	0	1	1	0			
				0	1	0	0	0	0		
					1	0	1	1	0		
					0	0	1	1	0	0	0
							1	0	1	1	0
							0	1	1	1	0

- Le CRC est donc 1110 et le mot à transmettre 11100111 1110.

- Réception d'un mot :

1	1	1	0	0	1	1	1	1	1	1	0
1	0	1	1	0							
0	1	0	1	0	1						
	1	0	1	1	0						
	0	0	0	1	1	1	1	1			
				1	0	1	1	0			
				0	1	0	0	1	1		
					1	0	1	1	0		
					0	0	1	0	1	1	0
							1	0	1	1	0
							0	0	0	0	0

- Le reste de la division est nulle, il n'y a donc pas d'erreur.

Exercices : On utilisera le polynôme générateur $x^4 + x^2 + x$.
1. On souhaite transmettre le message suivant :1111011101, quel sera le CRC à ajouter ?
2. Même question avec le mot 1100010101.
3. Je viens de recevoir les messages suivants : 1111000101010, 11000101010110, sont-ils corrects ?

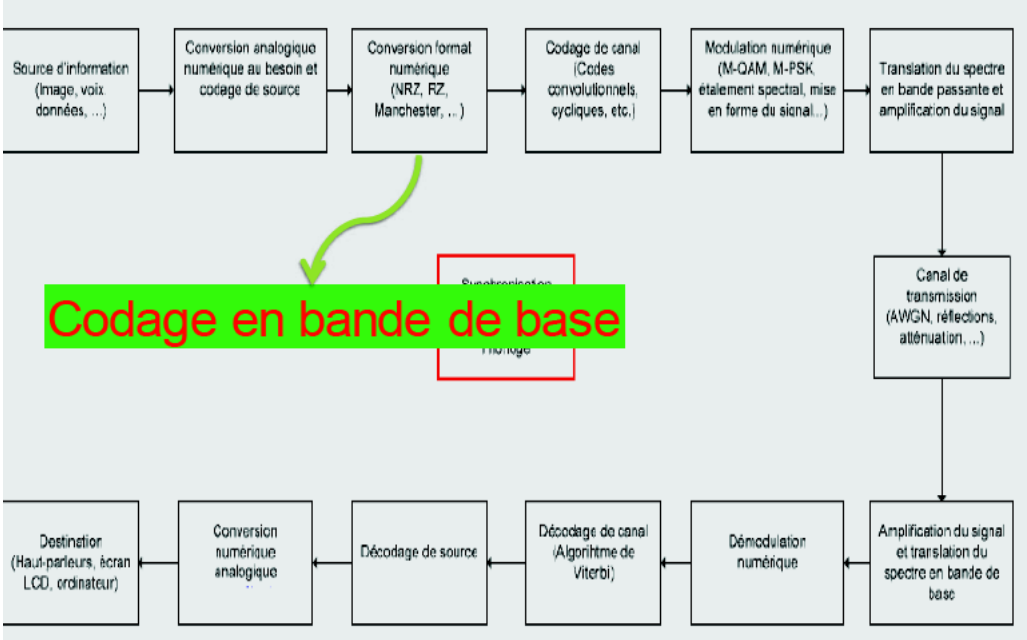
3 Communications numériques

3.1 Introduction

En communications numériques,la source émet un message discret : on entend par là que le message prend ses valeurs dans un ensemble dénombrable, le plus souvent fini, de valeurs. Typiquement une suite de K bits ne peut prendre que 2K valeurs. Afin d'être transmise, la suite des données d'information émise parla source est associée, par le modulateur, a un signal qui subit a travers le canal des perturbations.
La transmission d'une information numérique passe par la création d'un signal qui peut être considéré de deux points de vues:sous son aspect temporel,ou sous son aspect fréquentiel.

Afin de numériser un signal analogique, on passe par trois étapes : l'échantillonnage, la quantification et le codage.
Les modes de transmission numérique sont:

3.2 Transmission en bande de base



Dans ce type de transmission l'information est émise sous sa forme initiale (numérique) avec une amplification et éventuellement une codification.

Elle est surtout utilisée pour les transmissions courte (Ethernet, série, etc.)

Le principe dans ce type de transmission est de définir un niveau de tension ou une transition entre deux tensions afin de coder le signal numérique.

1. **Code RZ**

   Au niveau des composants de transmission dans l'ordinateur,les informations binaires sont codees de facon basique:

   Un signal a 1 est code sous un signal compris entre 2 et $5V$

   Un signal a 0 est code autour de $0V$

   Ce type de codage qui est le plus simple reste localisée $\grave{a}$ l'intérieur de la carte mère n'est pas adaptée $\grave{a}$ une transmission filaire dans la mesure $o\grave{u}$ un signal 0 est très sensible $\grave{a}$ toute perturbation électrique.De plus,dans ce type de transmission un signal nul peut $\grave{a}$ la fois représenter la transmission d'un 0 mais également aa l'absence de transmission.Ce qui fait que la reconnaissance d'un message avec ce type de codage reste problématique.Pour palier ces différents problèmes,d'autres codes plus évolutifs ont été crées.

2. **Code NRZ**

   Le codage NRZ(Non retour a zéro) code le 1 par un signal positif,le bit 0 par un signal négatif.

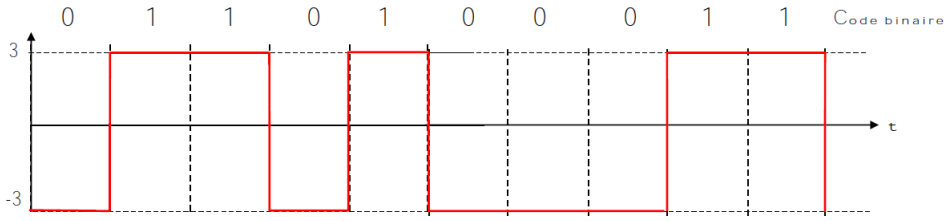


Figure 3.1: NRZ

3. **Code NRZI**

   Le code NRZI(No Return to Zero Inverted) est en fait une variante inversée du code NRZ.

   Un bit 0 est code par une tension positive,un bit *à* 1 par une tension négative.Il s'agit du type de transmission utilisée entre autre sur une ligne RS232 avec des niveaux de tensions de + ou - 12 volts.

   L'inconvénient de ces types de codage réside dans la direction et reconnaissance de longues chaines de 0 ou de 1.

4. **Code Manchester**

   Le code Manchester ou code biphasé cherche *à* amener une réponse au problème précédent .Ce code est bas sur une variation du signal.Il s'agit d'observer du signal entre le début et la fin du temps élémentaire.

   - Le bit 1 est code par une variation de +V a -V
   - le bit 0 est code par une variation de -V a +V

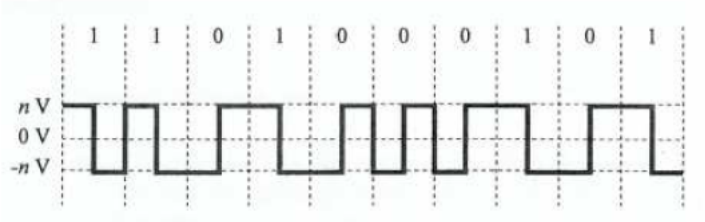


Figure 3.2: Manchester

Ce code est celui adopte pour les réseaux Ethernet

5. **Code Manchester différentiel**

   De la meme facon, ce code est basee sur les transitions du signal.

   - Le bit zéro est codé par une transition en début du temps élémentaire
   - Le bit 1 est codé de la même façon par une transition en milieu du temps élémentaire

   La transition réalisé dans tous les cas en milieu du temps élémentaire permet de garder une synchronisation entre l'émetteur et le récepteur.

   Ce code est celui utilisee dans la norme 802.5

6. **Code Miller**

   Le code de Miller est egalement base sur une codification a partir des transitions du signal

   - Le bit 0 est codee par l'absence de transition pendant le temps elementaire
   - Le bit 1 est codee par une transition

   Pour éviter le problème de synchronisation lié aux longues séquences de 0,une transition en cas de succession de 0 est réalisée en début de chaque temps élémentaire

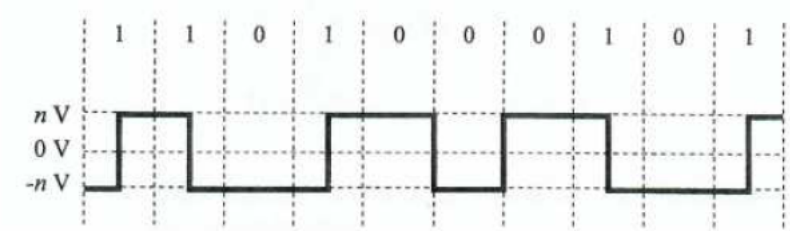


Figure 3.3: Codage Miller

3.2.1 Caractéristique d'un canal de transmission

1. **Bande passante**

   La principale caractéristique d'une voie de transmission (câble,fibre optique,etc) est sa bande passante.C'est l'intervalle de fréquences a l'intérieur duquel les signaux seront correctement transmis.

   Pour transmettre des signaux numériques il faut que la ligne de transmission possède une grande bande passante.Les signaux analogiques utilisent une bande passante plus étroite.Le RTC offre un intervalle de fréquence de $300à3400Hz$,ce qui limite la bande passante a $3.1KHz$.

2. **La valence d'une voie**

   Un code associe *à* une valeur physique (un signal électrique) a une valeur logique (un signal binaire).La valence notée V est le nombre de valeurs que peut prendre l'état physique *à* un instant t.

   Par exemple on parle de bivalence si le signal peut prendre deux valeurs physiques de tension +V et –V.

3. **Le moment élémentaire**

   C'est la durée élémentaire pendant laquelle il est nécessaire d'émettre le signal physique sur le câble afin qu'il soit reconnu par le récepteur.Ce temps s'exprime en secondes.On parle également de temps d'horloge.Le moment est noté T_m.

4. **La vitesse de modulation** C'est le nombre de valeurs physiques emises par secondes.La vitesse de modulation(ou Rapidité de modulation) se note R_m et s'exprime en bauds.

   $$R_m = \frac{1}{T_m}$$

   La vitesse de modulation correspond au nombre d'états physiques que l'interface peut émettre par secondes

5. **Le débit binaire**

   Egalement appele de transmission,c'est le nombre de valeurs logiques transmises par secondes.Il est D et s'exprime en $bits/s$.

   $$D = R_m \log_2 V$$

6. **Capacité d'un canal de transmission**

   On définit la capacité C d'un canal de transmission comme étant la vitesse maximale de transfert des informations transportées. La capacité maximale théorique d'un canal de communication en présence de bruit est donnée par la relation suivante :

   $$C = B_p . \log_2 \left[1 + \left(\frac{P_S}{P_B}\right)\right]$$

   Avec C la capacité maximale en bps, B_P la bande passante du canal en Hz et $\left(\frac{P_S}{P_B}\right)$ le rapport entre puissance moyenne du signal et la puissance moyenne du bruit.
   Pour un système de transmission multi-niveaux la capacité du canal de transmission est donnée par la formule :

   $$C = 2.B_p . \log_2 V$$

7. **débit d'une transmission**

   On distingue le débit binaire (nombre de bits par seconde, noté D) et le débit de symboles (nombre de symboles par secondes, noté R)débit binaire :il se calcule à partir de la durée T_b d'un bit

3.3 Modulation numérique

Compromis entre l'efficacité spectrale, la puissance et le taux d'erreurs

Lors de construction d'un système des communications, trouver le meilleur compromis entre les divers paramètres de système est fondamental. Les objectifs du constructeur peuvent être les suivants:

- maximiser l'efficacité spectrale;
- minimiser le taux d'erreurs pas bit;
- minimiser la puissance émise;
- minimiser la bande passante;
- améliorer la qualité de service, c'est-a-dire accepter le maximum utilisateurs avec le minimum d'interférences crées entre eux;
- minimiser la complexité du système, etc.

Dans cette partie du cours nous nous concentrons sur le compromis entre l'efficacité spectrale $Db = B$, la probabilité d'erreurs par bit P_b et la puissance du signal émis. Très souvent, P_b est remplace par le BER et la puissance du signal émis par le SNR; c'est équivalent. Le but est de maximiser $D_b = B$ et de minimiser le BER et le SNR. Comme il est difficile d'optimiser ses trois paramètres au même temps, nous allons les considérer par paires, en

fixant le troisième paramètres.
La question est jusqu'ou l'optimisation est possible. La théorie de l'information est une matière qui étudie les limites théoriques des systèmes de communication. Voici deux exemples importants. Considérons la paire $D_b/B - SNR$, supposant la transmission sur le canal gaussien. La limite théorique dans ce cas est donnée par la théorème de Shannon:

$$\frac{C}{B} = \log_2\left(1 + \frac{E_b}{N_o}\left(\frac{C}{B}\right)\right)$$

ou $\frac{E_b}{N_o}$ est le SNR, et $C = B$ est l'efficacité spectrale maximale (quand le débit binaire Db est égale á la capacité théorique C, étant la valeur maximale possible).

Pour simplifier, supposons dans la suite que la probabilité d'erreurs pas bit Pb est

xe. Alors, nous avons *à* trouver le compromis entre l'efficacité spectrale P_b/B et la puissance du signal émis (SNR pour P_b donnée).Le choix du modulateur/démodulateur est dicté par ce compromis.

Dans les communications numériques, on utilise la modulation d'amplitude et la modulation de phase. Ces deux modulations peuvent être utilisées separément, mais dans ce cas elles sont difficiles à générer à l'émetteur et difficiles à détecter au récepteur. Donc, en pratique, nous utilisons ces deux modulations d'une manière qu'elles dépendent l'une de l'autre: le signal à émettre (qui est complexe) est separé en deux composants, *I ("In-phase")* et *Q ("quadrature")* , qui correspondent aux parties réelle et imaginaire du signal.

3.4 Modulation/démodulation dans la chaîne de communication

Les techniques de modulations sont utilisées pour adapter le signal à la bande transposée. Il existe trois méthodes possibles:

- modulation d'amplitude
- modulation de fréquence
- modulation de phase

Dans les communications numériques, on utilise la modulation d'amplitude et la modulation de phase. Ces deux modulations peuvent être utilisées séparément, mais dans ce cas elles sont difficiles à générer à l'émetteur et difficiles à détecter au récepteur. Donc, en pratique, nous utilisons ces deux modulations d'une manière qu'elles dépendent l'une de l'autre: le signal à émettre (qui est complexe) est séparé en deux composants, *I ("In-phase")* et *Q ("quadrature")*, qui correspondent aux parties réelle et imaginaire du signal.

Définition: La modulation s'appelle M-aire, si chaque symbole émis peut prendre M valeurs possibles. Dans la plupart des cas, le symbole est formé à partir d'un vecteur de k bits, ce qui conduit à $M = 2^k$.

Définition: Le modulateur est un module de la chaîne de communication qui forme des symboles a_k (en général,complexes), en fonction des vecteurs de bits à son entrée. Le démodulateur est un module situé à la réception, qui estime les valeurs des bits correspondant au symboles a_k , en ayant une estimation des a_k à son entrée.

Lors du design du modulateur, deux choses principales sont à déterminer:
- type de modulation ou la constellation dans la plan complexe,
- étiquettes des points de la constellation (mapping).

Le type de modulation est défini par le fait si le système est plutôt limité en puissance ou en bande passante.
En ce qui concerne le démodulateur, il est placé derrière le module de prise de décision ou il le remplace. Pour démoduler, on définit les régions de décision sur la plane complexe et on accorde au vecteur des bits estimé la valeur correspondante.

3.5 Types des modulations

Nous allons maintenant présenter les types de modulation les plus répandues en utilisant fig: 3.4 comme illustration

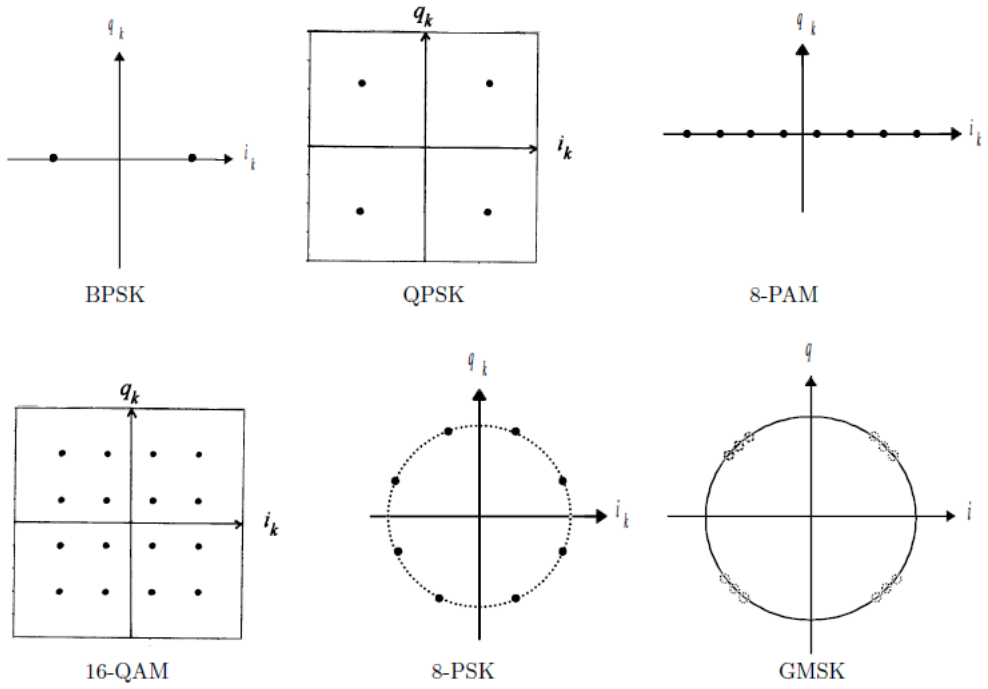


Figure 3.4: Exemples des modulations.

BPSK (Bipolar Phase Shift Keying) - la modulation la plus simple. Nous avons $a_k = \{\pm 1\}$. Chaque symbole est réel.

QPSK ou 4-QAM (Quadriple Phase Shift Keying ou Quadrature Amplitude Modulation avec $M = 4$). Les symboles a_k prennent les valeurs $1+j, 1-j, -1+j, -1-j$. Chaque symbole est généré à partir d'un vecteur de 2 bits.

M-PAM (Pulse Amplitude Modulation avec M symboles) : $a_k = \{\pm 1, \pm 2, ... \pm M/2\}$. Les symboles sont réels. Comme nous avons vu dans les cours précédents, cette modulation n'est pas très efficace quand M est grand, puisqu'elle prend beaucoup de puissance émise. Notons que BPSK est un cas particulier de M-PAM.

M-QAM (Quadrature Amplitude Modulation avec M symboles): $a_k = n + jm$, ou $n, m \in Z$, $n, m \leq \sqrt{M}$. Ceci est une modulation largement utilisé dans les systèmes avec des limitations en bande passante. Quand la constellation est grande (M grand), elle peut consommer beaucoup de puissance. QPSK est un cas particulier. D'habitude,$M = 4, 16, 64, 256$, parce que, pour les constellations carrées, les voies I et Q peuvent être indépendantes.

M-PSK (Pulse Shift Keing modulation avec M symboles) : $ak = n + jm$ ou l'amplitude de a_k est égale à 1, $||a_k|| = \sqrt{n^2 + m^2} = 1$. Autrement dit, les points de la constellation sont situées sur un cercle unitaire autour du zéro. Ceci est une modulation bien adaptée aux systèmes avec des limitations en puissance émise (la puissance émise par symbole est constante et égale à 1). BPSK et QPSK sont des cas particuliers. On rencontre également 8-PSK, mais rarement plus, car avec le nombre des points la probabilité d'erreurs par symbole augmente.

GMSK (Gaussian Message Shift Keying modulation) : une modulation à phase continue et amplitude constante.

3.6 Comparaison des modulations diverses

Faisons la comparaison entre les modulations présentées, en comparant leurs probabilités d'erreur par bruit minimales et leurs efficacités spectrales. Soit α le coefficient d'arrondi du filtre du cos surélevé utilisé. Pour la BPSK nous avons:

$$\frac{D_b}{B} = \frac{1}{1+\alpha}, P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

Pour la M-PAM, la puissance du signal reçu vaut:

$$\frac{1}{2}\sigma^2 \frac{2E_h}{T_S} = \frac{\left(M^2-1\right)E_h}{T_S}$$

Sachant que $E_b = PT_b$, nous obtenons

$$P_{b,min} = 2\frac{M-1}{M\log_2 M} Q\left(\sqrt{\frac{6\log_2 M}{M^2-1}\frac{E_b}{N_0}}\right)$$

Le résultat en bande transposée est le même que celui pour la bande de base. La M-PAM ne rapport aucun gain particulier pour P_b en bande transposée. L'efficacité spectrale est la moitié de celle obtenue en bande de base.

$$\frac{D_b}{B} = \frac{\log_2 M}{1+\alpha}$$

Pour la M-QAM, sous condition d'indépendance des voies I et Q,

$$P_b = P_{b,I} = P_{b,Q}$$

La puissance moyenne par symbole est

$$P = \frac{M-1}{3}\frac{E_h}{T_S}$$

qui conduit à

$$P_{b,min} = 4\frac{\sqrt{M}-1}{\sqrt{M}\log_2 M}Q\left(\sqrt{\frac{3\log_2 M}{M-1}\frac{E_b}{N_0}}\right)$$

Comme le spectre de l'enveloppe constante occupe la même largeur de bande que les spectre des composants I et Q,

$$\frac{D_b}{B} = \frac{\log_2 M}{1+\alpha}$$

Pour la M-PSK, nous avons

$$\frac{D_b}{B} = \frac{\log_2 M}{1+\alpha}$$

et

$$P_{b,min} = \frac{2}{\log_2 M}Q\left(\sqrt{2\log_2 M\frac{E_b}{N_0}}\sin\frac{\pi}{M}\right)$$