Théorie de l'information

Françoise Briolle

Table des matières

Introduction

La théorie de l'information ou, de façon plus précise, la théorie statistique de la communication, est l'aboutissement des travaux d'un grand nombre de chercheurs (H. Nyquist, R.W.L. Hartley, D. Gabor, ...) sur l'utilisation optimale des moyens de transmission de l'information (téléphone, télégraphe, etc.). Le premier exposé synthétique de cette théorie est due à Claude E. Shannon, ingénieur aux Bell Telephone Laboratories. L'idée fondamentale est que l'information doit être transmise à l'aide d'un canal (ligne téléphonique, ondes hertziennes). On est alors conduit à étudier d'une part l'information proprement dite (quantité d'information, entropie d'une source d'information, etc.), d'autre part les propriétés des canaux (capacité, etc.), et enfin les relations qui existent entre l'information à transmettre et le canal employé en vue d'une utilisation optimale de celui-ci.
On peut ainsi considérer la théorie de l'information comme une théorie du signal au sens large. Elle intervient chaque fois qu'un signal est envoyé et reçu, et s'applique, par conséquent, aussi bien à la téléphonie, à la télégraphie et au radar qu'à la physiologie du système nerveux ou à la linguistique (la notion de canal se retrouve alors dans la chaîne formée par l'organe de phonation, les ondes sonores et l'organe auditif).
En fait, les concepts de base de la théorie de l'information sont d'une telle simplicité et d'une telle généralité qu'il est possible de les introduire dans n'importe quelle discipline, des mathématiques à la sociologie.

Bibliographie

C.E. SHANNON & W. WEAVER, *The Mathematical Theory of Communication*, Illini Books edition, 1963.
E. ROUBINE, *Introduction à la théorie de la communication, tome III*, Edition Masson, 1970.

Chapitre 1

Notions de probabilités

La théorie de l'information utilise des notions de probabilités dont nous rappelons ici quelques éléments de base.
Les théories mathématiques de probabilité utilisent des expériences, réelles ou imaginaires, comme le lancer d'une pièce, le jet de dé, le tirage de cartes, le comptage du nombre d'accidents de la route, etc.. Après une expérience, on observe un résultat dont on cherche certaines propriétés, comme par exemple, la fréquence d'apparition. La théorie des probabilités s'est imposée au 17^{eme} siècle dans l'étude des jeux de hasard (jeux de cartes, de dés, etc. ...)..

1.1 Notions fondamentales

1.1.1 Expérience aléatoire

Une *expérience aléatoire* se décrit par la donnée de l'ensemble des résultats possibles de l'expérience en question. On note ω un résultat de l'expérience et Ω l'espace de tous les résultats possibles.

Exemple

1. On lance une pièce. L'ensemble des résultats possibles est :
   $\Omega = \{$ P (pile), F (face) $\}$
2. On lance un dé et on s'intéresse au chiffre lu sur sa face supérieure :
   $\Omega = \{ 1, 2, 3, 4, 5, 6 \}$
3. On lance deux dés et on s'intéresse à leurs faces supérieures :
   $\Omega = \{\omega = (i,j), \quad 1 \leq i \leq 6, \quad 1 \leq j \leq 6\}$
4. On lance deux dés et on s'intéresse à la somme des deux faces supérieures :
   $\Omega = \{ 2, 3, \ldots, 12 \}$
5. On observe les points d'impacts d'électrons sur une cible plane :
   $\Omega = R^2$

1.1.2 Evénement

Un *événement* A est un ensemble de résultats ω d'une expérience aléatoire. L'événement A est un sous ensemble de Ω. Il se réalise si le résultat $\omega \in A$.

Exemple
On lance un dé et on s'intéresse au chiffre lu sur sa face supérieure. On considère l'événement : "le chiffre lu est pair".
Le sous-ensemble de Ω associé à cet événement est A = { 2, 4, 6 }.
Si le résultat ω de l'expérience est 2, 4 ou 6, l'événement A est réalisé.

1.1.3 Opérations logiques sur les événements aléatoires

1. Un *événement impossible*, qui ne se réalise jamais, sera noté comme l'*ensemble vide* $\emptyset$ dans Ω.

2. Un *événement certain*, qui se réalise toujours, sera noté Ω puisqu'il est réalisé quel que soit le résultat de l'expérience.

3. A tout événement A est associé son contraire, non A, noté A^c, qui est réalisé lorsque A ne l'est pas. L'ensemble correspondant à A^c est le *complémentaire* dans Ω de l'ensemble représentatif de A.

4. Pour tout couple d'événements A_1 et A_2, l'événement "A_1 et A_2" est celui qui se réalise si les événements A_1 et A_2 sont réalisés en même temps. L'événement "$A_3 = A_1$ et A_2" est représenté par les résultats ω qui appartiennent à la fois à A_1 et A_2, c'est-à-dire à l'*intersection* de ces ensembles. $A_3 = A_1 \cap A_2$.
   Les deux événements A_1 et A_2 sont *incompatibles* si $A_1 \cap A_2 = \emptyset$.

5. Pour tout couple d'événements A_1 et A_2, l'événement "A_1 ou A_2" est celui qui se réalise si les événements A_1 ou A_2 sont réalisés. L'événement "$A_3 = A_1$ ou A_2" est représenté par les résultats ω qui appartiennent ou à A_1 ou à A_2, c'est-à-dire à l'*union* de ces ensembles. $A_3 = A_1 \cup A_2$.

6. Si l'événement A *implique* la réalisation de l'événement B, alors A ne peut être réalisé sans que B ne le soit. Tous les résultats ω de A appartient aussi à B. L'ensemble A est *contenu* dans B.

1.2 Probabilité

1.2.1 La loi empirique des grands nombres

Faute de pouvoir prédire qu'une expérience donne lieu à un résultat déterminé, on veut associer à chaque événement un nombre représentant la probabilité qu'il se réalise.

La *fréquence de réalisation d'un événement A* au cours de N répétitions de l'expérience à laquelle il est lié s'écrit :

$$F_N(A) = \frac{N_A}{N}$$

N : nombre de répétitions de l'expérience,
N_A : nombre de réalisations de l'événement A au cours de N répétitions de l'expérience.

On vérifie expérimentalement que cette fréquence F_N fluctue de moins en moins lorsque N augmente. Ce résultat expérimental, connu sous le nom de la *loi empirique des grands nombres* permet de définir la *probabilité de l'événement A* comme :

$$P(A) = \lim_{N\to\infty} \frac{N_A}{N}$$

On vérifie facilement les propriétés suivantes :

1. P(A) ≥ 0,
2. $P(\Omega) = 1$,
3. $P(\sum_{i\in I} A_i) = \sum_{i\in I} P(A_i)$, pour des événements A_i incompatibles et dont les résultats sont dénombrables.

1.2.2 Définition et application au cas discret

Soit une expérience aléatoire dont les résultats sont représentés par l'ensemble Ω dénombrable (cas discret), on appelle *probabilité sur* Ω toute application P :

$$P(\Omega) \to [0,1],$$

telle que :

1. $P(\Omega) = 1$,
2. Pour toute famille dénombrable $(A_i)_{i\in I}$, deux à deux incompatibles, on a : $P(\sum_{i\in I} A_i) = \sum_{i\in I} P(A_i)$ (axiome de σ-additivité).

L'ensemble Ω muni de la loi P, noté (Ω,P) est appelé *espace probabilisé.*
Ainsi toute probabilité P pour un espace Ω dénombrable est entièrement déterminé par la suite $(p_\omega)_{\omega\in\Omega} \in [0,1]^\Omega$ satisfaisant la condition

$$\sum_{\omega\in\Omega} p_\omega = P(\Omega) = 1$$

Exemple

1. Considérons une expérience aléatoire dont l'ensemble des résultats Ω contient k éléments. On peut choisir $p_\omega = \frac{1}{k}$.
   Cette loi de probabilité est appelée *loi uniforme* car tous les éléments ont la même probabilité $\frac{1}{k}$.
2. Dans le cas particulier où $\Omega = N$, on peut choisir $\forall n \in \Omega,\ \ P_\lambda(n) = \frac{e^{-\lambda}\lambda_n}{n!}$.
   Cette loi de probabilité est appelée *loi de Poisson.*

1.2.3 Probabilité d'un événement

On considère une expérience aléatoire dont les résultats sont représentés par l'ensemble Ω dénombrable muni d'une probabilité P. Pour tout résultat $\omega \in \Omega$, on pose $p_\omega = P(\omega)$.
Tout événement A, inclus dans Ω, s'écrit sous la forme $A = \sum_{\omega \in A}\{\omega\}$. En utilisant l'axiome de σ-additivité :

$$P(A) = \sum_{\omega \in A} p_\omega$$

Exemple
Considérons l'expérience dans laquelle on lance deux fois une pièce et l'événement A : " Pile apparaît".
Les résultats que l'on peut obtenir sont : $\Omega = \{$PP, PF, FP, FF$\}$.
Les résultats de l'expérience pour lesquels A est réalisé sont : $A = \{$PP, PF et FP$\}$.
Donc $P(A) = p(PP) + p(PF) + p(FP) = \frac{1}{4} + \frac{1}{4} + \frac{1}{4} = \frac{3}{4}$.
Bien sur, la probabilité pour que l'événement A ne se produise pas (c'est-à-dire lorsque Pile n'apparaît jamais) est égale à $1 - p(A)$.

Propriétés

Ces propriétés de la probabilité d'un ou plusieurs événements sont liés à la σ-additivité de P.

1. $\forall A \in \Omega, P(A^c) = 1 - P(A)$
   Comme $A \cup A^c = \Omega$, la σ-additivité de P implique $P(\Omega) = P(A) + P(A^c) = 1$
2. $P(\emptyset) = 0$
   On applique l'égalité précédente à $A = \emptyset$ (ce qui implique $A^c = \Omega$).
3. $\forall A, B \in \Omega,\ P(A \cup B) = P(A) + P(B) - P(A \cap B)$
   Les événements A et $B - A$ sont incompatibles, car $A \cap B - A = \emptyset$. Donc $A \cup B = A \cup (B - A)$. Appliquons la σ-additivité de P, $P(A \cup B) = P(A) + P(A - B)$. Comme $B = (B - A) + (A \cap B)$, on en déduit $P(B - A) = P(B) - P(A \cap B)$ et donc la relation énoncée plus haut.

1.3 Conditionnement et indépendance

1.3.1 Probabilités conditionnelles

Exemple introductif

On considère une famille ayant deux enfants. Chaque enfant est désigné par la lettre F si c'est une fille et par la lettre G si c'est un garçon. L'espace des résultats Ω est donc { (F, F), (F, G), (G, F), (G, G) } ; (F, G) signifie que la fille est née avant le garçon.
On suppose qu'il y a équiprobabilité pour chaque élément de Ω .
Définissons les deux événements suivants :

- A = "la famille a deux garçons" : $P(A) = \frac{1}{4}$
- B = "la famille a au moins un garçon" : $P(B) = \frac{3}{4}$

On suppose que B est réalisé, c'est-à-dire que la famille a au moins un garçon. Les éventualités qui réalisent A sont donc dans $A \cap B$ et la probabilité de A lorsque B est réalisé, vaut

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Dans cet exemple, la probabilité d'avoir un deuxième garçon lorsqu'on en a déjà eu un (B est réalisé) vaut $\frac{1}{3}$.

Définition

Soit (Ω,P) un espace probabilisé. Soient deux événements A et B tels que $P(B) \neq 0$. On appelle *probabilité conditionnelle* de A sachant B la quantité :

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Si $P(B) = 0$, alors l'événement B ne peut pas se produire et $P(A|B)$ non plus.
On remarque que l'on $A \to P(A|B)$ est une nouvelle probabilité sur Ω, notée $P(.|B)$ qui vaut 0 sur tous les événements incompatibles avec B.

Exemple 1
Soit une population de n personnes composée de f femmes et de b blonds. A est l'événement que la personne choisie au hasard soit une personne blonde et B qu'elle soit une femme.
$P(A) = \frac{b}{n}$ $\qquad$ $P(B) = \frac{f}{n}$
Soit f_b le nombre de femmes blondes. La probabilité pour que la personne choisie soit une femme blonde, que l'on note $P(A|B)$ est :
$P(A|B) = \frac{f_b}{f} = \frac{f_b}{n}\frac{n}{f} = \frac{P(A \cap B)}{P(B)}$

Exemple 2
Un carton contient 80 ampoules électriques dont 20 sont défectueuses. Une ampoule, sélectionnée au hasard ne fonctionne pas. Quelle est la probabilité pour que, si on choisit une nouvelle ampoule (sans remettre la première dans le carton), on tire encore une ampoule défectueuse.
A est l'événement d'obtenir une ampoule défectueuse au premier tirage :
$P(A) = \frac{20}{80} = \frac{1}{4}$.
B est l'événement pour lequel on tire au deuxième tirage une ampoule défectueuse.
$B|A$ est l'événement pour lequel on tire une ampoule défectueuse lorsque A s'est produit (il n'en reste que 19): $P(B|A) = \frac{19}{79}$.
On peut calculer $P(A \cap B)$ qui est la probabilité pour que les évenements A et B se produisent :
$P(A \cap B) = P(B|A)\ .\ P(A) = \frac{19}{79}\ .\ \frac{1}{4} = \frac{19}{316}$

1.3.2 Système exhaustif d'événements

Soit une suite dénombrable d'événements $(A_n)_{n\in I}$ et leur réunion $\cup_{n\in I} A_n$ qui désigne l'événement "A_1 ou A_2 ou ...". On conviendra de noter cette réunion $\sum_{n\in I} A_n$ lorsque les événements sont deux à deux incompatibles. Si les relations suivantes sont vérifiées :

1. $\forall (n,m) \in I^2,\ n \neq m \Rightarrow A_n \cap A_m = \emptyset$ (événements incompatibles deux à deux)
2. $\sum_{n\in I} A_n = \Omega$

alors les ensembles $A_n,\ n \in I$ forment une partition de Ω. Il est certain qu'un et un seul événement A_n parmi tous les $n \in I$ sera réalisé. Un tel système d'événements est appelé *système exhaustif d'événements.*

1.3.3 Formule de Bayes

Soit $(B_n)_{n\in I}$ un système exhaustif d'événements tel que $\forall n \in I,\ P(B_n) \neq 0$.
alors,

$$\forall A \in \Omega,\ P(A) \ = \ \sum_{n\in I} P(A|B_n)\ .\ P(B_n)$$

Démonstration. Comme les $B_n,\ n \in I$ sont deux à deux incompatibles, il en est de même des $A \cap B_n,\ n \in I$. Ainsi $A = \sum_{n\in I}(A \cap B_n)$ et
$P(A) = \sum_{n\in I} P(A \cap B_n) = \sum_{n\in I} P(A|B_n)\ .\ P(B_n)\ .$

1.3.4 Evénements indépendants

En général $P(A|B) \neq P(A)$.
On a l'équivalence $P(A|B) = P(A) \Rightarrow P(A \cap B) = P(A)\ .\ P(B)$.

Définition

Dans un espace probabilisé (Ω, P) deux événements A et B sont dits *indépendants* si $P(A \cap B) = P(A)\ .\ P(B)$.

Remarques

1. Un événement A est toujours indépendant de $\not\emptyset$ et Ω.
2. Si deux événements A et B sont indépendants, alors il en est de même pour A^c et B, de A et B^c, de A^c et B^c.

Exemple et généralisation

L'expérience consiste à jeter deux dés :

$$\Omega = \{\omega = (i,j),\ \ 1 \leq i \leq 6,\ \ 1 \leq j \leq 6\}$$

On s'intéresse aux deux événements suivants :

- A = "le premier dé donne un chiffre pair" : $P(A) = \frac{1}{2}$
- B = "le deuxième dé donne un chiffre pair" : $P(B) = \frac{1}{2}$

On voit facilement que $P(A \cap B) = \frac{1}{4}$ et que $P(A \cap B) = P(A)\ .\ P(B)$. Les événements A et B sont indépendants.
On considère maintenant l'événement :

- C = "la somme des deux chiffres lus est paire" : $P(C) = \frac{1}{2}$

On a $P(A \cap C) = \frac{1}{4} = P(A)\ .\ P(C)$. Donc A et C sont indépendants.
Par symétrie, il en est de même pour B et C. Donc A, B et C sont deux à deux indépendants.
Et pourtant $P(A \cap B \cap C) \neq P(A)\ .\ P(B)\ .\ P(C)$
Cet exemple montre que trois événements deux à deux indépendants ne sont pas "globalement" indépendants quand ils sont pris dans leur ensemble.

Définition

Dans un espace probabilisé (Ω,P), n événements $(n \geq 3)$ sont dits *indépendants dans leur ensemble*, si :

$$\forall k,\ \ 1 \leq k \leq n,\ \ P(A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_k}) = \prod_{j=1}^{k} P(A_{i_j})$$

1.4 Un exemple récapitulatif

1.4.1 Description de l'expérience et du problème

On dispose de deux dés, le dé a et le dé b. On jette les deux dés et on lit le nombre apparaissant sur la face supérieure du dé a, puis celui apparaissant sur la face supérieure du dé b.
Soit l'entier n appartenant à $\{1,\ 2,\ 3,\ 4,\ 5,\ 6\ \}$, on va répondre aux questions suivantes :

1. Quelle est la probabilité pour que le premier chiffre lu soit n ? Et celle que le deuxième chiffre lu soit n ?
2. Quelle est la probabilité d'obtenir un double ?
3. Quelle est la probabilité d'obtenir au moins un 6 ?
4. Quelle est la probabilité d'obtenir un double ou au moins un 6 ?

1.4.2 Solution

On va définir l'espace Ω des résultats :

$$\Omega \ = \ \{\omega_{i,j} \ = \ (i,j), \quad 1 \leq i \leq 6, \quad 1 \leq j \leq 6\}$$

$Card(\Omega) = 6\ .\ 6 = 36$. Comme les dés ne sont pas pipés, tous les résultats $\omega_{i,j}$ pour i et j dans $\{1,\ 2,\ 3,\ 4,\ 5,\ 6\ \}$ sont équiprobables, donc :

$$P(\omega_{i,j}) = \frac{1}{Card(\Omega)} = \frac{1}{36}$$

La loi de probabilité est uniforme.

1. Quelle est la probabilité pour que le premier chiffre lu soit n ? Et celle que le deuxième chiffre lu soit n ?
   Soit l'événement A_n "le premier chiffre lu est n".
   Il est clair que $A_n = \{\omega_{n,j} = (i,j), \quad 1 \leq j \leq 6\}$ et que $Card(A_n) = 6$. Comme le loi de probabilité est uniforme,

   $$P(A_n) = \frac{Card(A_n)}{Card(\Omega)} = \frac{1}{6}$$

   De même en posant B_n "le deuxième chiffre lu est n" et en appliquant le même raisonnement, on obtient :

   $$P(B_n) = \frac{1}{6}$$

2. Quelle est la probabilité d'obtenir un double ?
   Soit C l'événement "les deux chiffres lus sont égaux". Le sous-ensemble C de Ω est égal à : $C = \sum_{n=1}^{6}(A_n \cap B_n)$.

En effet les événements de la suite $(A_n \cap B_n)_{1 \leq n \leq 6}$ sont deux à deux incompatibles puisque $(A_n \cap B_n) = \{\omega_{n,n}\}$.
Puisque $P(\{\omega_{n,n}\}) = \frac{1}{36}$, on en déduit

$$P(C) = \sum_{n=1}^{6} P(\{\omega_{n,n}\}) = \frac{1}{6}$$

3. Quelle est la probabilité d'obtenir au moins un 6?
   Soit D l'événement "l'un au moins des chiffres lus est un 6". L'événement est décrit par l'ensemble $D = A_6 \cup B_6$. En remarquant que $P(A_6) = P(B_6) = \frac{1}{6}$ et que $P(A_6 \cap B_6) = \frac{1}{36}$, on calcule P(D) :

$$P(D) = P(A_6) + P(B_6) - P(A_6 \cap B_6) = \frac{11}{36}$$

4. Quelle est la probabilité d'obtenir un double ou au moins un 6?
   Soit E l'événement "les chiffres lus forment un double ou contiennent au moins un 6". L'événement est décrit par l'ensemble $E = C \cup D$. Remarquons que $C \cap D = \{\omega_{6,6}\}$ et donc que $P(C \cap D) = \frac{11}{36}$. Nous pouvons alors calculer P(E) :

$$P(E) = P(C) + P(D) - P(C \cap D) = \frac{4}{9}$$

Chapitre 2

Source discrète d'information

2.1 Quelques définitions

2.1.1 Sources d'information

On distingue deux types de sources d'information : les sources qui délivrent une information *analogique* et celles qui délivrent une information *discrète*.
L'information analogique peut être représentée par une fonction continue. C'est le cas du signal de parole, par exemple, qui est un signal acoustique c'est-a-dire une variation continue de la pression de l'air. Le signal de parole est représenté par la fonction continue p(t), de la variable "t", le temps. Cette fonction, à valeurs réelles, peut prendre une infinité de valeurs différentes, même si elles sont bornées par des extremas (-1 μPa, + 1 μPa, par exemple).
L'information délivrée par une source discrète est représentée par une fonction discrète qui n'aura de valeurs qu'à des instants précis ; ces valeurs seront quantifiées, donc en nombre fini.
Un morceau de musique gravé sur un disque vinyle est une information analogique. Le sillon gravé est continu d'un bout à l'autre du disque. Il est traduit, grâce à la cellule de lecture, en un signal électrique continu en fonction du temps et à valeurs réelles : ce signal peut prendre n'importe qu'elle valeur entre -1V et +1V.
Le même morceau de musique gravé sur un disque compact est une information discrète. Le signal de musique est échantillonné toutes les 22 μs : on ne connaît la valeur du signal qu'à des instants précis, toutes les 22 μs. Ces valeurs sont quantifiées sur 16 bits ; elles ont une précision relative puisqu'elles ne peuvent prendre que 2^{16} valeurs différentes.
Il est toujours possible de convertir une information analogique en une information discrète, mais ce n'est pas l'objet de ce cours.
Dans ce cours nous nous intéressons au codage de sources discrètes.

2.1.2 Sources discrètes d'information

1. Une source discrète d'information émet une information discrete, que l'on appelle *message* ou *signal*.

2. Un message est une suite, finie ou infinie, de *symboles*.

3. Les symboles que l'on utilisent pour composer le message sont en nombre fini. Ils constituent un *alphabet*.

   Exemple Lorsque j'écris ce texte, je produis un message discret, sous la forme d'un texte. Pour composer ce texte j'utilise un alphabet fini d'environ 45 symboles. Il est composé de 26 lettres, 10 chiffres et de signes typographiques comme l'espace, le point, la virgule, deux points, l'apostrophe, deux parenthèses, trois accents, etc..
   CECI EST UN MESSAGE écrit avec les 11 symboles suivants : C E I S T U N M A G espace. Ce message à une longueur de 19 symboles.

4. Lorsqu'on travaille avec un ordinateur, on utilise uniquement les symboles 0 et 1, que l'on appelle des bits (abréviation de **bi**nary digi**t**). Il faut donc traduire, c'est-à-dire *coder*, les symboles utilisés par la source en une séquence finie de 0 et 1. Cette séquence est appelée *mot-code*.
   Par exemple on peut traduire la lettre A par le mot-code 000110.

5. Les mots-code, bien que tous différents, peuvent avoir le même nombre de 0 et 1 pour représenter tous les symboles de l'alphabet. Le code est dit *code de longueur fixe*. Dans le cas contraire (mots-code de différentes longueurs), c'est un *code de longueur variable*.

6. On traduit les symboles de la source en séquences de 0 et 1 (mot-code) à l'aide d'un *dictionnaire*.

7. L'ensemble des mots-code utilisé pour représenter les différents symboles est appelé le *code* de la source.

   Exemple Un signal musical, représenté par un signal électrique dont l'amplitude varie entre -1 mV et $+1$ mV, est discrétisé. La fréquence d'échantillonnage F_e vaut 44.1 kHz ($T_e = 1/F_e = 22\ \mu s$) ; le signal est quantifié sur 16 bits soit $2^{16} = 65\ 536$ niveaux, (compris entre -1 mV et $+1$ mV). Toutes les 22 μs ($T_e = 1/F_e = 22\ \mu s$) on convertit la valeur réelle du signal en une valeur approchée égale à l'un des niveaux de quantification. Chaque niveau représente un symbole codé par un mot-code de 16 bits. Le signal discret est donc composé d'une succession de mots-code de 16 bits. C'est un code de longueur constante;

2.2 Propriétés statistiques de la source

Une source discrète d'information est un processus qui produit des messages aléatoires constitués d'une suite de symboles, provenant d'un alphabet fini. La suite des symboles, finie ou infinie, est aléatoire ; même si dans certain cas elle est prévisible, elle n'est jamais connue de façon certaine. C'est donc un *processus stochastique* .

2.2.1 Statistique de la source

Considérons une source qui émet des messages suffisamment longs, de longueur N, à partir de symboles A_i (i = 1, ..., n).

$$A_3A_5A_kA_jA_1A_lA_mA_2A_9 \ldots\ldots A_1A_4A_k$$

Nous pouvons mesurer :

- la probabilité d'apparition des symboles A_i dans le message de longueur N,

$$p(A_i) = \frac{\#A_i}{N} = p_i$$

  $\#A_i$: nombre de fois où le symbole A_i apparaît dans le message.
- la probabilité d'apparition des digrammes A_iA_j,

$$p(A_iA_j) = \frac{\#A_iA_j}{(N-1)}$$

- la probabilité d'apparition des trigrammes $A_iA_jA_k$, des 4-grammes, etc..

Les probabilités d'apparition des symboles A_i et des différents polygrammes décrivent la statistique de la source.

2.2.2 Source ergodique

La source est ergodique si les différents messages produits par la source ont les mêmes propriétés statistiques.
Par exemple, considérons la source dont les messages sont des textes français. La mesure statistique faite a partir de différents textes est relativement stable : la probabilité d'apparition du E est d'environ 0.148, du S 0.077, du N 0.071, du T 0.068, de l'espace séparateur de mot 0.184, etc.. On constate ensuite que certains groupes de lettres (polygrammes) sont plus fréquents que d'autres. Par exemple ON est 4 fois plus fréquent que NO ; Q est presque toujours suivi de la lettre U, P l'est souvent de H, X ne l'est jamais de Z. Il en est de même pour les trigrammes.
Cette source est ergodique.

2.2.3 Source stationnaire

La source est stationnaire si la mesure de sa statistique sur un message suffisament long ne dépend pas de l'endroit où commence la mesure.
Par exemple, la mesure de la statistique de la source qui produit des textes en français, est la même, que l'on commence la mesure au chapitre 1 ou au chapitre 5 d'un ouvrage.

2.2.4 Processus de Bernoulli (ou source sans mémoire)

La source est un processus sans mémoire si la production des différents symboles est indépendante des symboles précédemment émis.
Considérons la source qui produit deux symboles A et B avec les probabilités
$p(A) = p$ et $p(B) = q = (1 - p)$
Puisque c'est un processus de Bernoulli, les probabilités d'apparition des digrammes AA, AB, BA, BB sont égales à :
$p(AA) = p(A)p(A) = p^2$ $\qquad$ $p(BB) = p(B)p(B) = (1 - p)^2$
$p(AB) = p(BA) = p(A)p(B) = p(1 - p)$
On peut calculer de la même façon les probabilités de tous les polygrammes.
La statistique de la source est donc complètement définie par les probabilités d'apparition p_k des différents symboles A_k de son alphabet.

2.2.5 Source de Markov

La source est un processus de Markov lorsque la probabilité d'apparition des différents symboles A_j est conditionnée par le ou les symboles précedents.
C'est géneralement le cas des textes écrits dans des langues naturelles. Il y a dans tout texte usuel une influence plus ou moins forte entre lettres plus ou moins voisines. Par exemple en français, l'apparition d'un L conditionne l'apparition d'un E ou d'un A, mais interdit l'apparition d'un B. Avec une très bonne approximation une source de Markov sera un modèle approprié à la description de sources produisant des textes écrits. Mais il faut remarquer qu'il y a d'autres liaisons linguistiques que l'on doit également prendre en compte, celles de la syntaxe entre les mots, de la logique entre les propositions.
Soit une source de Markov d'ordre 1 définie sur un alphabet de n symboles A_k. La probabilité d'apparition des symboles A_j est conditionnée uniquement par le symbole émis précédemment A_k. Sa statistique est définie par :

- la probabilité d'apparition P_k de chacun des éléments A_k
- les probabilités p_{kj} d'apparition des symboles A_j lorsque la source a émis le symbole A_k. L'ensemble des probabilités p_{kj} constituent la matrice de transition de la chaîne de Markov.

La probabilité $p(k,j)$ d'apparition du digramme A_kA_j est égale à la probabilité d'apparition du symbole A_k multiplié par la probabilité d'apparition du symbole A_j conditionnée par A_k soit : $p(k,j) = P_k\ p_{kj}$
La probabilité $p(k,j,i)$ d'apparition du trigramme $A_kA_jA_i$, est le produit des probabilités de A_kA_j et de A_i conditionné par le symbole A_j, soit $p(k,j,i) = P_k\ p_{kj}\ p_{ji}$.
Pour une source de Markov d'ordre 2, sa statistique sera défine par l'ensemble P_k des probabilités d'apparition des symboles A_k, et des probabilités conditionnelles p_{kj} et p_{kji}.

2.3 Exemples de production de texte

Pour illustrer notre propos, nous allons produire différents textes a partir de sources discrètes qui ont des propriéteés différentes. Leur alphabet est composé de 27 symboles { A, B, ... Z, l'espace séparateur de mots }.

1. Production d'un texte a partir d'une source de Bernoulli où tous les symboles ont la même probabilité d'apparition :
   XFOML RXKHRJFFJUJ ZLPWCFWKCYJ FFJEYVKCQSGHYD QPAAM-KBZAACIBZLHJQD
2. La source est un processus de Bernoulli mais les symboles on les même probabilites d'apparition que celles mesurées dans un texte anglais :
   OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI ALHENHT-TPA OOBTTVA NAH BRL
3. La source est un processus de Markov d'ordre 1 (on prend en considération la fréquence d'apparition des symboles et des digrammes)
   ON IE ANTSOUTINYS ARE T INCTORE ST BE SDEAMY ACHIN D ILO-NASIVE TUCOOWE AT TEASONARE FUSO TIZIN ANDY TOBE
4. La source est un processus de Markov d'ordre 2 (prise en compte des trigrammes)
   IN NI IST LAT WHEY CRATICT FROURE BIRS GROCID PONDENOME OF DEMONSTRURES OF THE REPTAGIN IS REGOACTIONA OF CRE
5. Maintenant la source est un producteur de mots (= symboles) dont on prend en compte la fréquence d'apparition
   REPRESENTING AND SPEEDILY IS AN GOODAPT OR COME CAN DIFFERENT NATURAL HERE HE THE A INCAME THE TO OF EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE THESE
6. Cette source, producteur de mots, est maintenant un processus de Markov d'ordre 1. On impose une règle d'ordre 1 dans la structure de la phrase : un article est suivi d'un nom , un adjectif précède un nom, ...
   THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH WRITER THAT THE CHARACTER OF THIS POINT IS THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE TIME OF WHO EVER TOLD THE PROBLEM FOR AN UNEXPECTED

La ressemblance avec un texte anglais progresse lorsque l'on affine le modèle de la représentation de la source.

Chapitre 3

Mesure de l'information

Une information désigne, par définition, un ou plusieurs événements parmi un ensemble fini d'événements possibles.
Si, cherchant un document dans une pile de dossiers, on indique que ce document se trouve dans un dossier rouge, on donne une *information* qui réduira d'autant plus le temps de recherche que le nombre de dossiers rouges est plus restreint. Si on ajoute que le document est dans un petit dossier, on fournit une nouvelle information qui diminue encore le temps de recherche.

3.1 Une mesure de l'information

Une source sans mémoire produit un ensemble fini de n symboles dont nous connaissons les probabilités d'apparition p_1, p_2, ..., p_n. Pouvons nous définir une mesure de l'information contenue dans la source ?
Si une telle mesure existe, c'est une fonction des différentes probabilités des symboles. Elle doit avoir la forme $F(p_1, p_2, \ldots, p_n)$ et les propriétés suivantes :

1. F doit être continue en p_i,

2. F doit être une fonction positive, strictement monotone : plus on ajoute d'événements, plus l'information est riche.

3. $F(p_1, p_2, \ldots, p_n) = F(p_1, p_2, \ldots, p_n, 0)$
   Ajouter un événement impossible ($p_{n+1} = 0$) ne doit pas modifier la mesure de l'information contenue dans la source.

4. F doit être minimale, c'est-a-dire nulle, si un seul évenement peut se produire. Il n'y a pas d'incertitude, la sortie de la source peut être prédite à coup sur et de notre point de vue, elle ne contient pas d'information.

5. La fonction F doit être maximale si tous les événements sont equiprobables. C'est dans ce cas où la sortie de la source est la plus incertaine.

6. Soient deux sources, A et B, dont la mesure de l'information vaut respectivement F(A) et F(B). La mesure de l'information contenue dans la source B, lorsque l'on connaît A est $F(B|A)$. Alors, la mesure de l'information de la source AB, produit des deux sources A et B, doit être égale à l'information de la source A plus l'information contenue dans B, lorsqu'on connaît l'information de la source A :

$$F(AB) = F(A) + F(B|A)$$

Si les deux sources sont indépendantes, alors :

$$F(AB) = F(A) + F(B)$$

Ce sont les définitions données par Shannon pour définir la mesure de la quantité d'information contenue dans une source.

3.2 Quantité d'information d'un message

3.2.1 Définition

On veut évaluer par un nombre I l'information relative à un événement E.
Sur le plan pratique, une information est d'autant plus intéressante est rare : si on vous dit qu'il a fait beau a Marseille au mois de juin, l'information n'a pas beaucoup d'intérêt ; par contre, si on vous annonce qu'il y a eut un orage de grêle le 19 juin, l'information retient votre attention. On va donc attribuer un nombre I d'autant plus grand que l'événement est rare, peu probable.
D'autre part, soient les informations I_1 et I_2 qui désignent les événements E_1 et E_2 ; si E_1 et E_2 sont indépendants, alors l'information qui désigne ces deux événements vaut $I = I_1 + I_2$

Soit une expérience aléatoire qui produit n résultats contenus dans l'ensemble $\Omega = \{\omega_k\}$ tel que $Card(\Omega) = n$. Considérons l'événement E qui se réalise pour les résultats qui appartiennent à l'ensemble $A \subset \Omega$ tel que $Card(A) = f$.
La *quantité d'information de l'événement E* est par définition :

$$I = -log_a(\frac{f}{n})$$

Remarque
La quantité d'information calculée en utilisant la fonction log_2 (a = 2) s'exprime en **bit**, abréviation de **bi**nary uni**t**. C'est généralement l'unité qu'on utilise lorsque les sources produisent des messages avec un alphabet de deux symboles {0 et 1}.
Lorsqu'on utilise des logarithmes Nepperiens ln (a = e) la quantité d'information s'exprime en **nat**.
Si on utilse des logarithmes décimaux, elle s'exprime en **decit** ou en **Hartley**.

Soit une source aléatoire S sans mémoire qui produit des messages avec les n symboles $\{A_k\}$, $k = 1, \ldots, n$ dont les probabilités d'apparition valent $p(A_k) = p_k$. Ces probabilités sont telles que $0 \leq p_k \leq 1$ et $p_1 + p_2 + \ldots + p_n = 1$.
La *quantité d'information du symbole* A_k vaut :

$$I(A_k) = -log_a(p_k)$$

La source sans mémoire produit un message M composé de L symboles. La probabilité d'apparition de ce message, $p(M)$ est le produit des probabilités des L symboles qui le composent.
La *quantité d'information du message* M vaut :

$$I(M) = -log_a(p(M))$$

3.2.2 Exemples

Exemple 1
Soit une lettre choisie au hasard parmi les 26 lettres de l'alphabet. La probabilité de tirer cette lettre est $p = \frac{1}{26}$. La quantité d'information relative à cet événement vaut $I = log_2(\frac{1}{26}) = 4.7$ bits

Exemple 2
On range régulièrement 64 points sur une grille de 8 colonnes et 8 lignes. Soit E_j l'événement "le point choisit aléatoirement sur la grille est sur la j^{ieme} colonne" et F_k l'événement "le point est sur la k^{ieme} ligne"
$p(E_j) = p(F_k) = \frac{1}{8}$
$I(E_j) = I(F_k) = 3$ *bits*
La probabilité pour que le point soit sur la j^{ieme} ligne et la k^{ieme} colonne vaut $\frac{1}{64}$.
Donc $I(E_j \cap F_k) = -log_2(\frac{1}{64}) = 6$ *bits* $= I(E_j) + I(F_k)$
Remarque: $I(E_j \cap F_k) = I(E_j) + I(F_k)$ car les événements E_j et F_k sont indépendants.

Exemple 3
Recherche d'un document dans une pile de dossiers.
Soit N le nombre total de dossiers, n_1 le nombre de dossiers rouges, n_2 le nombre de dossiers petits et n le nombre de dossiers petits et rouges.
Indiquer que le document se trouve dans un dossier rouge donne une information qui vaut $I_1 = log_2(N/n_1)$ *bits*.
Indiquer que le document est dans un dossier petit vaut $I_2 = log_2(N/n_2)$ *bits*.
Enfin, indiquer que le document se trouve dans un dossier petit et rouge vaut $I_3 = log_2(N/n)$ *bits*.
On remarque que :
$log_2(\frac{N}{n}) = log_2(\frac{N}{n_1}\frac{n_1}{n}) = log_2(\frac{N}{n_1}) + log_2(\frac{n_1}{n})$.
On peut donc dire que $log_2(\frac{n_1}{n})$ est la quantité d'information supplémentaire que l'on donne "dans un dossier petit" lorsque l'information "dans un dossier rouge" est déjà connue.

En faisant un calcul similaire, on peut écrire que :
$log_2(\frac{N}{n}) = log_2(\frac{N}{n_2}\frac{n_2}{n}) = log_2(\frac{N}{n_2}) + log_2(\frac{n_2}{n})$.
Alors $log_2(\frac{n_2}{n})$ est la quantité d'information supplémentaire que l'on donne "dans un dossier rouge" lorsque l'information "dans un dossier petit" est déjà connue.
Ainsi, si l'on a 800 dossiers dont 200 sont petits, 50 rouges et 20 qui sont à la fois petits et rouges, l'information "dans un dossier rouge" vaut 4 bits, l'information "dans un dossier petit" vaut 2 bits et "dans un dossier petit et rouge" vaut 5.2 bits (et non 6 bits car les événements "dans un dossier petit" et "dans un dossier rouge" ne sont pas indépendants). Lorsqu'on dispose de deux informations, la quantité d'information totale n'est pas nécessairement la somme des quantités d'information.

3.3 Quantité d'information d'une source

3.3.1 Quantité d'information d'une source sans mémoire

Définition

Soit une expérience ou une source S, qui produit les symboles A_i (i=1, ..., n) et $p(A_i) = p_i$ avec $0 \leq p_i \leq 1$ et $p_1 + p_2 + \ldots + p_n = 1$.
Shannon définit la *mesure de l'information de la source* comme la valeur moyenne des quantités d'information de chacun des symboles produits par la source.

$$H(S) = E\{I(A_i)\} = -\sum_{k=1}^{n} p_k \; log_2(p_k)$$

Cette fonction est appelée *entropie* en mécanique quantique et en thermodynamique. C'est une mesure de l'incertitude ou du désordre de la source. Du point de vue de la théorie de l'information, c'est une mesure de la quantité d'information contenue dans la source.

Exemple
Soient trois sources (A, B, C) dont l'alphabet est limité à deux symboles.

A_1	A_2	B_1	B_2	C_1	C_2
0.5	0.5	0.99	0.01	0.3	0.7

Les événements produits par la source A sont équiprobables. Cette source contient beaucoup d'incertitude ; on ne peut prédire quel sera le symbole de sortie : il y a une chance sur deux pour que ce soit le symbole A_1 ou A_2.
La source B produit 99 fois sur 100 l'événement B_1. Elle ne contient pas beaucoup d'incertitude : l'état de la source est très probablement B_1.
Quant à la source C, elle est entre les deux.
La quantité d'information (entropie) de chacune de ces sources vaut respectivement H(A) = 1 bit, H(B) = 0.08 bit et H(C) = 0.88 bit.

Remarque
Claude Shannon a choisi d'utiliser l'entropie comme mesure de l'information, mais il aurait pu utiliser d'autres fonctions, comme l'entropie quadratique introduite par Renyi, la complexité, ou d'autres fonctions plus exotiques. Il se trouve que l'entropie est une mesure globale de la statistique de la source, qui peut donner des informations sur sa production.

Propriétés

1. La fonction $H(p_1, p_2, \ldots, p_n)$ est toujours positive ou nulle

$$H(p_1, p_2, \ldots, p_n) = -\sum_{k=1}^{n} p_k \, lg(p_k) \geq 0$$

   En effet $0 \leq p_k \leq 1$ donc $-lg(p_k) \geq 0$

2. La fonction $H(p_1, p_2, \ldots, p_n) = 0$ si et seulement si tous les p_k sont nuls sauf p_i qui vaut 1.
   Quelque soit la base du logarithme utilisé, $p_k \, lg(p_k) = 0$ si $p_k = 0$ et $lg(1) = 0$
   La sortie de la source est connue d'avance et égale à A_i ; elle ne contient pas d'incertitude. Son entropie, mesure de l'incertitude, est nulle.
3. La fonction $H(p_1, p_2, \ldots, p_n)$ est maximale et égale à $lg(n)$ lorsque tous les p_k ont une même valeur $p_k = \frac{1}{n}$.
   Tous les symboles ont la même probabilité d'appartion (équiprobable) ; la sortie de la source peut être l'un de ces symboles et il est impossible de prédire lequel. La source contient beaucoup d'incertitude ; son entropie est maximale et égale à $lg(n)$.
4. Considérons maintenant deux sources A et B, qui produisent respectivements les symboles A_i, (i = 1, ..., n) et B_j, (j = 1, ..., m), dont les probabilités d'apparition sont respectivement p_i et q_j.
   Soit H(A) et H(B) les quantités d'information des sources A et B. Si les deux sources sont indépendantes, l'entropie de le source AB, qui émet les symboles A_iB_j vaut :

$$H(AB) = H(A) + H(B)$$

   Démonstration : les deux sources A et B sont indépendantes. La probabilité pour que le couple d'événements A_kB_l se produise est égale à $\pi_{kl} = p_k \;\; q_l$. Calculons l'entropie de la source qui émet les symboles A_kB_l. Cette source est le produit de la source A par la source B.

$$H(AB) = -\sum_{k=1}^{n}\sum_{l=1}^{n} \pi_{kl} \, lg(\pi_{kl}) = -\sum_{k=1}^{n}\sum_{l=1}^{n} p_k \, q_l \, lg(p_k \, q_l)$$

$$H(AB) = -\sum_{k=1}^{n}\sum_{l=1}^{n} p_k \, q_l \, (lg(p_k) + (lg(q_l)) = -\sum_{k=1}^{n} p_k \, lg(p_k) - \sum_{l=1}^{n} q_l \, lg(q_l)$$

$$H(AB) = H(A) + H(B)$$

3.3.2 Entropie d'une source de Markov

Soit une chaîne de Markoff d'ordre 1, définie sur un alaphabet fini,
$\{A_k$, k = 1, ...,n$\}$.
Sa statistique est définie par la probabilité P_k d'apparition de chaque symboleA_k et la matrice de transition p_{kj}, qui indique la probabilité d'apparition du symbole B_j lorsque le symbole A_k vient d'être produit.
Calculons l'entropie de la source, lorsqu'elle a émis le symbole A_k, pour arriver dans n'importe quel état A_i, i=1, ..., n.

$$H_k = -\sum_{i=1}^{n} p_{ik}\ lg(p_{ik})$$

Donc l'entropie de la source, lorqu'elle est dans un état A_k quelconque, pour arriver dans un état A_i quelconque, c'est-à-dire lorsqu'elle évolue d'un pas est :

$$H = \sum_{k=1}^{n} P_k\ H_k$$

L'entropie d'une source de Markoff est donc :

$$H = -\sum_{l=1}^{n}\sum_{i=1}^{n} P_k\ p_{ik}\ lg(p_{ik})$$

Chapitre 4

Codage

4.1 Introduction

L'objet du codage est de traduire les messages émis par la source sous la forme d'une suite de 0 et de 1.
Les messages produits par la source sont une suite de symboles A_k provenant d'un alphabet fini de n symboles $\{A_1, \dots A_n\}$. Ils ont la forme suivante :

$$\dots A_i A_j A_k \dots A_l A_m A_p A_q \dots$$

A chaque symbole on attribue un *mot-code*, c'est-à-dire une séquence finie de 0 et de 1.
$A_k \longleftrightarrow 0011$ $A_l \longleftrightarrow 010$ etc.

On appelle *dictionnaire* l'ensemble des correspondances symbole $\longleftrightarrow$ mot code.
Il permet de traduire le message, qui sera repésenté comme une suite de 0 et de 1 (codage):

$$\dots 00010101101 \dots 11001010101 \dots$$

et de traduire une séquence de 0 et 1 en une suite de symboles de l'alphabet (décodage) :

$$\dots A_i A_j A_k \dots A_l A_m A_p A_q \dots$$

Pour être exploitable, le code doit être *déchiffrable* sans ambiguïté : chaque séquence de 0 et de 1 doit permettre, en utilisant le dictionnaire, de reconstituer un et un seul message, celui émis par la source.
Il est généralement souhaitable que le code soit le "plus court possible" afin que la transmission du message codé soit la plus rapide possible.

4.2 Classification des codes

4.2.1 Définition

Une source émet des messages avec 4 symboles qui sont codés par les six codes suivants. Nous allons utiliser cet exemple pour expliquer la notion de classement d'un code.

	code 1	*code* 2	*code* 3	*code* 4	*code* 5	*code* 6
A_1	00	00	0	0	0	1
A_2	01	01	1	10	01	01
A_3	00	10	00	110	011	001
A_4	11	11	11	111	0111	0001

1. *Code de longueur fixe*
   Lorsque les mots-code ont la même longueur (même nombre de 0 et de 1) pour tous les symboles de l'alphabet, le code est *de longueur fixe.*
   Exemple : les codes 1 et 2 du tableau ci-dessus sont de longueur 2.

2. *Code de longueur variable*
   Un code de *longueur variable* est un code dont les mots-code ne sont pas tous de même longueur. Exemple : les codes 3, 4, 5 et 6 du tableau ci-dessus.

3. *Code univoque*
   Un code *univoque* est un code dont chaque mot-code est distinct de tous les autres mots-code. C'est le cas des codes 2 à 6 du tableau ci-dessus. Un contre exemple : le code 1 pour lequel le codage des symboles A_1 et A_3 sont identiques.

4. *Code sans préfixe*
   Un code est *sans préfixe* si aucun mot-code n'est le préfixe (début) d'un autre mot-code.
   Exemple : les codes 2, 4 et 6 du tableau ci-dessus sont sans préfixe.
   Le code 5 n'est pas sans préfixe ; en effet le mot-code 01 représentant le symbole A_2 a pour préfixe 0, mot-code du symbole A_1. Tous les mots-code de ce code débutent par le préfixe 0.

5. *Code irréductible*
   Un code est *irréductible* lorsque le dictionnaire permet de reconstituer sans ambiguïté le message. Pour chaque message de longueur finie émis par la source, la séquence codée ne coïncide jamais avec un autre message. On remarque que le code 3 n'est pas irréductible puisque la séquence binaire 1001 peut correspondre aux messages $A_2A_3A_2$ ou $A_2A_1A_1A_2$.
   Une condition suffisante (mais non nécessaire) pour qu'un code soit irréductible est qu'aucun de ses mots-code ne soit le préfixe d'un autre mot-code. Les codes 2, 4, et 6 du tableau ci-dessus sont irréductibles. Le code 5, qui n'est pas sans préfixe, est aussi irréductible.

6. *Code instantané*
   Un code irréductible est dit *code instantané* si la fin de tout mot-code est identifiable sans examen des symboles 0 et 1 qui composent le mot-code suivant. Les code instantanés sont des codes sans préfixe.

7. *Code optimal*
   Un code est dit *optimal* s'il est instantané et présente une longueur moyenne L minimale. Pour calculer la longueur moyenne du code, il faut connaître la probabilité d'occurence de chaque symbole.

4.2.2 Théorème de Kraft

Soit une source discrète sans mémoire produisant des messages avec un alphabet de N symboles $\{A_i\}_{(i=1,...N)}$. Ces symboles sont codés par des séquences binaires $\{x_i\}_{(i=1,...N)}$ dont les longueurs sont respectivement $l_1, l_2, \ldots, l_N$.
Une condition nécessaire et suffisante d'existence d'un code binaire instantané est qu'il vérifie l'inégalité de Kraft :

$$K = \sum_{k=1}^{N} 2^{-l_k} \leq 1$$

On remarquera que l'inégalité de Kraft nous assure qu'il existe bien un code instantané, déchiffrable dont la longueur des mots-code satisfait l'inégalité en question. Elle ne nous dit rien sur la façon d'engendrer un tel code, pas plus qu'elle nous garantit qu'un code satisfaisant à cette inégalité est ipso facto décodable de façon unique.

4.3 Durée transmission du message

On cherche à minimiser la durée (ou le coût) de transmission d'un message. Le but est de produire la plus courte possible séquence binaire représentant le message . Pour cela on va prendre en compte la fréquence d'apparition des symboles A_i. L'idée sous-jacente est de coder par des mots brefs les symboles qui apparaissent souvent alors que les symboles qui apparaissent quasiment jamais seront codés par des mots plus longs.

4.3.1 Exemple introductif

Supposons une source sans mémoire, sur un alphabet de 4 symboles, dont la statistique est la suivante :

	A_1	A_2	A_3	A_4
p_k	1/2	1/4	1/8	1/8

Examinons les deux codes suivants :

	A_1	A_2	A_3	A_4
code 1	00	01	10	11
code 2	0	10	110	111

Le premier code à une longueur de mot code égale à deux bits, quelque soit le symbole représenté. C'est un code de longueur constante. Le coût de transmission de ce code est de 2 bits/symbole.

Calculons, pour le code 2, le coût moyen de chaque symbole. En moyenne, après un temps assez long,

- il y a 1 chance sur 2 (1/2) pour que le symbole A_1 apparaisse ; il est codé sur 1 bit ;
- il y a 1 chance sur 4 (1/4) pour que le symbole A_2 apparaisse ; il est codé sur 2 bits ;
- il y a 1 chance sur 8 (1/8) pour que le symbole A_3 apparaisse ; il est codé sur 3 bits ;
- il y a 1 chance sur 8 (1/8) pour que le symbole A_4 apparaisse ; il est codé sur 3 bits ;

Donc, en moyenne, le coût de transmission du message est de :

$$\frac{1}{2}(1) + \frac{1}{4}(2) + \frac{1}{8}(3) + \frac{1}{8}(3) = 1.75\ bits/symbole.$$

Pour cette source, dont la statistique est donnée plus haut, il est plus intéressant, du point de vue de la rapidité de transmission des messages, d'utiliser le code 2. On gagne en *moyenne* 12.5%[1] en coût (ou en temps) de transmission des messages. Nous pouvons vérifier que ce code est irréductible, c'est-à-dire qu'il ne permet qu'une seule interprétation de la séquence binaire.

Pour une autre statistique de la source, le code 2 n'est pas forcément le plus intéressant. Par exemple, dans le cas d'une production équi répartie des symboles :

	A_1	A_2	A_3	A_4
p_k	1/4	1/4	1/4	1/4

La longueur moyenne du code 2 est de :

$$\frac{1}{4}(1) + \frac{1}{4}(2) + \frac{1}{4}(3) + \frac{1}{4}(3) = 2.25\ bits/symbole.$$

La longueur moyenne du code 1 n'est que de 2 bits/symbole.

1. $\frac{(2-1.75)}{2} = 0.125$

4.3.2 Définitions

1. Soit une source S sans mémoire et un alphabet $\{A_i\}$ avec $(i = 1, \ldots, n)$ dont les probabilités d'occurence sont $p(A_i) = p_i$. Les symboles sont codés par des séquences binaires $\{x_i\}$ de longueurs respectives l_i.
   La *longueur moyenne du code* est :

$$L = \sum_{i=1}^{n} p_i \; l_i$$

   Le paramètre L représente le nombre moyen de bits par symbole utilisé dans le processus de codage.

2. On définit l'*efficacité d'un code* η de la façon suivante :

$$\eta = \frac{L_{min}}{L}$$

   où L_{min} est la valeur minimale que peut prendre L. Lorsque η approche la valeur 1, on dit que le code est efficace.

3. La *redondance* γ d'un code a pour expression :

$$\gamma = 1 - \eta$$

4.4 Le théorème de Shannon

C'est en 1948 que Claude Shannon, chercheur aux Bell Laboratories, publie un article intitulé "The Mathematical Theory of Information".[2]
Cet article comporte une vingtaine de théorèmes, qui ont permis le développement d'une branche nouvelle de la physique mathématique, la théorie de la communication. Même si les démontrations ne sont pas toujours rigoureuses (il aura fallut attendre les travaux de A. I. Khinchin et de B. McMillan), les travaux de Shannon ont permis le développement de recherches nouvelles en communication.
Nous exposons ici le théorème fondamental, qui s'énonce en deux parties. Ce théorème permet de réaliser un codage efficace de sources discrètes, sur un alphabet fini, stationnaires et ergodiques.

2. C. E. Shannon, Bell Syytem Technical Journal, **27**, 379-423 ; 623-656 (1948)

4.4.1 Théorème 1

Considérons une source discrète, sur un alphabet fini de n symboles, stationnaire et ergodique qui produit N messages (C) de longueur s ($N = n^s$). Ils sont de la forme : $A_i A_j \dots A_k A_l$ avec s symboles. Soit $p(C)$ la probabilité d'obtenir un tel message.

Pour un $\epsilon > 0$ et un $\nu > 0$, aussi petit qu'on souhaite, et pour un nombre s suffisamment grand, tous les messages de la forme (C) peuvent être divisés en deux groupes avec les propriétés suivantes :

1. La probabilité $p(C)$ de n'importe quel message du premier groupe vérifie l'inégalité
$$|\frac{lg_2(\frac{1}{p(C)})}{s} - H| < \nu$$
2. La somme des probabilités de toutes les messages du second groupe est inférieure à ϵ.

En d'autre termes, tous les messages du premier groupe ont une probabilité d'apparition $p(C)$ comprise entre $2^{-s(H+\nu)}$ et $2^{-s(H-\nu)}$ et les messages du deuxième groupe n'apparaissent pratiquement jamais.
Prenons l'exemple d'une source utilisant l'alphabet {0,1} et d'entropie H = 0.7. ($0 \leq H \leq 1$) produisant des messages de longueur 10. Choisissons $\nu = \epsilon = 10^{-5}$. D'après le théorème, les messages appartenant au premier groupe ont une probabilté d'apparition $p(C)$ qui est comprise entre 7.812 10^{-3} et 7.813 10^{-3} soit approximativement $2^{-7} = 2^{-sH}$. La probabilité d'apparition des messages du deuxième groupe vaut $p_2(C) = 10^{-5}$.

4.4.2 Théorème 2

Rangeons maintenant les messages du premier groupe en ordre décroissant de probabilité $p(C)$. Nous allons sélectionner un nombre $N_s(\lambda)$ de messages tel que la somme des probabilités des $N_s(\lambda)$ messages sélectionnés soit inférieure à λ ($0 < \lambda < 1$).

$$\lim_{s\to\infty} \frac{lg\ N_s(\lambda)}{s} = H$$

Ce théorème indique que la limite de $N_s(\lambda)$ ne dépend pas de λ quand s croît, si et seulement si λ reste constant. Soit M cette limite.

Ce théorème permet de dire :

1. Il y a un nombre M de messages de longueur s, qui ont une grande probabilité $p(C)$ d'apparition(cf. th.1). M vaut environ 2^{sH} (cf. th. 2). Nous remarquons que M, le nombre de ces messages, est inférieur à N ($N = 2^s$), le nombre total de messages de longueur s que l'on peut produire avec 2 symboles.
2. Il y a un nombre P de messages, qui ont une probabilité d'apparition très faible (cf. th. 1). Le nombre P de ces messages vaut :
$P = N - M = N - 2^{sH} = 2^{slg(n)} - 2^{sH} = 2^{s(lg(n)-H)}$.

4.4.3 Applications au codage

Considérons une source discrète, stationnaire et ergodique, qui utilise un alphabet de deux symboles {a, b}. L'entropie H de la source est calculée en utilisant un logarithme en base 2. Elle est toujours positive et inférieur ou égale à 1.
Considérons tous les messages de s symboles de la forme :

$$aababbab, \ldots, ababbb \qquad \text{séquence à s symboles}$$

Le nombre total de ces messages de longueur s symboles est $N = 2^s$. Si on veut représenter tous les messages de longueur s, il faudra une sucession de 0 et 1 de longueur 2^s, car nous devons utilser un code de 1 bit par symbole ($a \longleftrightarrow 0$ et $b \longleftrightarrow 1$).

Supposons que l'entropie de la source soit inférieure à 1 (les deux symboles ne sont pas equiprobables). Le théorème de Shannon nous dit que :

1. il y a M messages qui apparaissent souvent, avec une probalilité supérieure à 2^{-sH}. Le nombre de ces messages est inférieur ou égal à $M = 2^{sH}$. M est inférieur à $N = 2^s$, le nombre total de messages possible de longueur s.
2. il y a un nombre P de messages qui n'apparaissent pratiquement jamais, avec une probabilité inférieur à ϵ. Le nombre de ces messages est $P = N - M = 2^{s(1-H)}$.

On va donc prendre un risque ϵ, en ignorant les P messages qui n'apparaissent pratiquement jamais. Ce risque est faible puisqu'on peut choisir ϵ aussi petit que l'on veut.
On va donc coder uniquement les M messages qui apparaissent souvent. Comme $M = 2^{sH}$ il suffit de sH symboles pour coder les M messages de s symboles, soit H bits/symboles. La compression réalisée ainsi est de (1 - H).
Shannon nous propose une formulation génerale.

4.4.4 Théorème 3

Soit une source S, ergodique et stationnaire, d'entropie $H(S)$. Il existe un code de *longueur minimale* qui permet de traduire les symboles $\{A_i\}$ en séquences binaires $\{x_i\}$ de longueur l_i. La longueur minimale du code est

$$L_{min} = H(S)$$

4.5 Codage entropique

On appelle *codage entropique* l'élaboration d'un codage dont la longueur moyenne des mots-code reflète l'entropie d'une source discrète sans mémoire. Nous allons étudier deux exemples de codage entropique.

4.5.1 Codage de Shannon-Fano

On obtient un codage efficace en appliquant le procédure suivante, connue sous la forme d'*algorithme de Shannon-Fano*.

1. Lister les symboles de la source par probabilités décroissantes.
2. Partager l'ensemble en deux sous-ensembles aussi équilibrés que possible au sens de la sommation des probabilités élémentaires des symboles.
3. Répéter le processus de partage en assurant au mieux l'équilibre jusqu'à ce que l'opération devienne impossible.

A_i	$p(A_i)$	étape 1	étape 2	étape 3	étape 4	Code
A_1	0.30	0	0			00
A_2	0.25	0	1			01
A_3	0.20	1	0			10
A_4	0.12	1	1	0		110
A_5	0.08	1	1	1	0	1110
A_6	0.05	1	1	1	1	1111

L'entropie de cette source vaut H(S) = 2.36 bit/symbole. La longueur moyenne du code construit avec l'algorithme de Shannon-Fano est de 2.38 bit/symbole. L'efficacité de ce code est de 99 %.

4.5.2 Codage de Huffman

Le codage de Huffman produit généralement un code optimal. C'est le plus efficace des codages. La méthode est la suivante.

1. Lister les symboles de la source par probabilités décroissantes.
2. Attribuer 0 et 1 aux symboles dont les probabilités sont les plus faibles. Ces deux symboles correspondent à un symbole unique dont la probabilité est la somme des deux probabilités.
3. Répéter cette procédure jusqu'à obtenir un symbole dont la probabilité vaut 1.

A_i	$p(A_i)$	étape 1	étape 2	étape 3	étape 4	Code
A_1	0.30				× **1** *0.55*	**1**1
A_2	0.25				× **0**	× **1**0
A_3	0.20			× **0**		**0**0
A_4	0.12		× **0** *0.25*	*0.45* **1**0 ×		× **0**10
A_5	0.08	× **1** *0.13*	**1**1 ×	**1**11		**0**111
A_6	0.05	× **0**	**1**0	**1**10		**0**110

Comme le code précedent, la longueur moyenne du code construit avec l'algorithme de Huffman est de 2.38 bit/symbole. L'efficacité de ce code est de 99 %. En général, l'algorithme de Huffman aboutit à des codes plus efficace que ceux construit avec l'algorithme de Fano.

Chapitre 5

Canaux discrets sans mémoire

5.1 Canal de transmission

5.1.1 Définitions

On appelle *canal de transmission* le support ou le milieu qui achemine le message entre l'emetteur et le récepteur. On n'envoie à travers ce canal que des signaux binaires (0 et 1, point et trait, etc.). Si le canal est imparfait (canal bruyant), la réception d'un 1, par exemple, ne permettra pas de conclure à l'envoi d'un 1 en toute certitude, mais seulement avec une certaine probabilité.

Un *canal discret sans mémoire* (SDM) peut être représenté comme un modèle statistique d'entrée X et de sortie Y.

$$X \quad \longrightarrow \quad P(x_i|y_j) \quad \longrightarrow \quad Y$$

L'entrée X se compose de m symboles $\{ x_1, x_2, \ldots, x_m \}$ dont les probabilités d'apparition $P(x_i)$ sont supposés connues.
La sortie Y se compose de n symboles $\{ y_1, y_2, \ldots, y_n \}$
Chaque correspondance possible entrée/sortie est définie par une probabilité conditionnelle $P(y_j|x_i)$ d'obtenir y_j lorsque x_i a été émis.
On appelle cette probabilité conditionnelle $P(y_j|x_i)$ la *probabilité de transition du canal.*

Le canal est *discret* lorsque les alphabet de X et Y sont finis. Il est *sans mémoire* lorsque le symbole de sortie fourni par le canal ne dépend que du dernier symbole reçu en entrée, indépendamment de tous les symboles d'entrée précédents.

5.1.2 Matrice de transition d'un canal

On définit un canal de façon complète en spécifiant l'ensemble de ses probabilités de transition. On définit la matrice des probabilités de transition [P(Y—X) appelée *matrice du canal.*

$$[\mathbf{P}(\mathbf{Y}|\mathbf{X})] = \begin{pmatrix} P(y_1|x_1) & P(y_2|x_1) & \dots & P(y_n|x_1) \\ P(y_1|x_2) & P(y_2|x_2) & \dots & P(y_n|x_2) \\ \dots & \dots & \dots & \dots \\ P(y_1|x_m) & P(y_2|x_m) & \dots & P(y_n|x_m) \end{pmatrix}$$

Comme chaque symbole appliqué à l'entrée du canal produit un symbole de sortie, on en déduit que :

$$\sum_{j=1}^{n} P(y_j|x_i) \;=\; 1 \quad \textit{pour tout } i$$

5.1.3 Quelques relations matricielles

1. Calcul des probabilités de la sortie $P(y_j)$
   Si on présente les probabilités de la variable d'entrée X comme un vecteur :

   $$[\mathbf{P}(\mathbf{X})] = \begin{pmatrix} P(x_1) & P(x_2) & \dots & P(x_m) \end{pmatrix}$$

   ainsi que celles de la variable de sortie Y :

   $$[\mathbf{P}(\mathbf{Y})] = \begin{pmatrix} P(y_1) & P(y_2) & \dots & P(y_n) \end{pmatrix}$$

   On obtient la relation matricielle, qui permet de calculer les probabilités de la sortie $P(y_j)$ en fonction des probabilités de la variable d'entrée et des probabilités de transition du canal :

   $$P(y_j) \;=\; \sum_{i=1}^{m} P(x_i)\; P(y_j|x_i)$$

   Soit, sous forme matricielle :

   $$[P(Y)] \;=\; [P(X)]\; [P(Y|X)]$$

2. Calcul des probabilités conjointes de transmettre x_i et de recevoir y_j, $P(x_i,y_j)$.
   On peut calculer ces probabilités conjointes $P(x_i,y_j)$ par la relation :

   $$P(x_i,y_j) \;=\; P(x_i)\; P(y_j|x_i)$$

   Si l'on représente $[P(X)]_d$ sous la forme d'une matrice diagonale :

   $$[\mathbf{P}(\mathbf{X})]_{\mathbf{d}} = \begin{pmatrix} P(x_1) & 0 & \dots & 0 \\ 0 & P(x_2) & \dots & 0 \\ \dots & \dots & \dots\dots & \\ 0 & 0 & \dots & P(x_m) \end{pmatrix}$$

   On peut alors écrire les probabilités conjointes $P(x_i,y_j)$ sous la forme matricielle [P(X,Y)] obtenue par la relation :

   $$[P(X,Y)] \;=\; [P(X)]_d\; [P(Y|X)]$$

5.1.4 Canaux remarquables

Canal sans perte

Dans un canal sans perte, aucune information issue de la source n'est perdue lors de la transmission. La matrice de transition d'un tel canal ne possède qu'un élément non nul par colonne.

Canal déterministe

Dans ce cas, chaque élément x_i de l'entrée correspond à un seul élément y_j de la sortie. La matrice de transition du canal ne possède qu'un élément non nul par ligne.

Canal sans bruit

On dit qu'un canal est sans bruit s'il est à la fois sans perte et déterministe. La matrice de transition ne possède qu'un élément non nul par ligne et par colonne.

Canal binaire symétrique

La matrice de transition d'un canal binaire symétrique est de la forme :

$$[\mathbf{P}(\mathbf{Y}|\mathbf{X})] = \begin{pmatrix} 1-\epsilon & \epsilon \\ \epsilon & 1-\epsilon \end{pmatrix}$$

Le canal est dit symétrique car la probabilité de recevoir un 1, si l'on a émis un 0, est la même que celle de recevoir un 0, lorsqu'un 1 a été émis : elle vaut ϵ.

5.2 Information mutuelle

Soit une source aléatoire, ergodique et stationnaire X, qui représente l'entrée d'un canal de transmission. Elle produit les événements (symboles)
x_i (i = 1, ..., m) dont les probabilités d'occurence sont $p(x_i)$ avec $0 \leq p(x_i) \leq 1$ et $p(x_1) + p(x_2) + \ldots + p(x_m) = 1$
L'entropie de cette source est l'incertitude moyenne relative à l'entrée du canal. Elle vaut :

$$H(X) = -\sum_{i=1}^{m} p(x_i)\ log_2(p(x_i))$$

Soit une deuxième source Y qui représente la sortie du canal et produit les événements (symboles) y_j (j=1, ..., n) dont les probabilités d'occurence sont $p(y_j)$ avec $0 \leq p(y_j) \leq 1$ et $p(y_1) + p(y_2) + \ldots + p(y_n) = 1$.
L'entropie de cette source est l'incertitude moyenne relative à la sortie du canal. Elle vaut :

$$H(Y) = -\sum_{j=1}^{n} p(y_j)\ log_2(p(y_j))$$

5.2.1 Entropies conditionnelle et conjointe

Il est possible en utilisant les probabilités $p(x_i)$ des symboles d'entrée et $p(y_j)$ des symboles de sortie ainsi que les probabilités conjointes $p(x_i,y_j)$ de définir diverses fonctions d'entropie relative à un canal de transmission à m entrées et n sorties.

1. $H(X|Y)$, *entropie de X conditionnée par Y*, est une mesure de l'incertitude sur l'entrée du canal X, une fois que l'on a observé la sortie Y. Cette fonction est aussi appelée *ambiguïté* de X par rapport à Y.

$$H(X|Y) = -\sum_{j=1}^{n}\sum_{i=1}^{m} p(x_i,y_j)\ log_2 p(x_i|y_j)$$

2. $H(Y|X)$, *entropie de Y conditionnée par X*, est une mesure de l'incertitude moyenne sur la sortie Y du canal sachant que X a été transmis.

$$H(Y|X) = -\sum_{j=1}^{n}\sum_{i=1}^{m} p(y_j,x_i)\ log_2 p(y_j|x_i)$$

3. $H(X,Y)$, l'*entropie conjointe*, est une mesure de l'incertitude d'ensemble relative à l'ensemble du canal de transmission.

$$H(X,Y) = -\sum_{j=1}^{n}\sum_{i=1}^{m} p(x_i,y_j)\ log_2 p(x_i,y_j)$$

Il existe deux relations intéressantes entre ces diverses quantités :

$$\begin{aligned} H(X,Y) &= H(X|Y) + H(Y) \\ H(X,Y) &= H(Y|X) + H(X) \end{aligned}$$

5.2.2 Information mutuelle

On définit l'*information mutuelle* $I(X;Y)$ de la façon suivante :

$$I(X;Y) = H(X) - H(X|Y)$$

Comme $H(X)$ représente l'incertitude relative à l'entrée du canal avant que l'on ait observé sa sortie et puisque $H(X|Y)$ représente l'incertitude sur l'entrée après que la sortie a été observée, $I(X;Y)$ mesure le gain d'incertitude relatif à l'entrée du canal qui résulte de l'observation de sa sortie.

Propriétés de $I(X;Y)$

1. $I(X;Y) = I(X;Y)$
2. $I(X;Y) \geq 0$
3. $I(X;Y) = H(Y) + H(Y|X)$
4. $I(X;Y) = H(X) + H(Y) + H(X,Y)$

5.3 Capacité d'un canal

5.3.1 Capacité par symbole d'un canal

L'information mutuelle $I(X;Y)$ dépend de $H(X)$ et $H(X|Y)$. La matrice des probabilités de transition $H(X|Y)$ est donnée : elle caractérise le canal. Par contre il existe de nombreuses lois de probabilité $\{p(x_i)\}$ des symboles d'entrée $\{x_i\}$. Parmi toutes ces lois de probabilité possible, il en existe une telle que la quantité $I(X;Y)$ est maximale.
La *capacité par symbole* d'un canal discret sans mémoire a pour expression :

$$C_s \ = \ max_{p(x_i)} \ I(X;Y) \quad b/symbole$$

La capacité du canal n'est fonction que des probabilités de transition $H(X|Y)$.

5.3.2 Capacité par seconde d'un canal

Si r représente le débit du canal en symboles par seconde, le débit maximal d'information du canal a pour valeur rC_s. On appelle cette quantité la *capacité par seconde du canal*. Elle est notée C et s'exprime en bit/seconde.

$$C \ = \ r \ C_s \quad b/s$$

5.3.3 Capacité de canaux remarquables

Canal sans perte

Pour un canal sans perte, on sait que $H(X|Y) = 0$ donc

$$I(X;Y) \ = \ H(X)$$

Ainsi l'information mutuelle est égale à l'entropie de l'entrée et on ne perd aucune information lors du transfert. En Conséquence la capacité du canal est égale à :

$$C_s \ = \ max_{p(x_i)} \ H(X) \ = \ log_2(m)$$

où m est le nombre de symboles de l'entrée X.

Canal déterministe

Pour un canal déterministe, on sait que $H(Y|X) = 0$, donc :

$$I(X;Y) \ = \ H(Y)$$

Ainsi l'information mutuelle est égale à l'entropie de la sortie. En Conséquence la capacité du canal est égale à :

$$C_s \ = \ max_{p(x_i)} \ H(Y) \ = \ log_2(n)$$

où n est le nombre de symboles de la sortie Y.

Canal sans bruit

Un canal sans bruit est un canal sans perte et déterministe. On en déduit :

$$I(X;Y) \;=\; H(X) \;=\; H(Y)$$

La capacité du canal vaut donc :

$$C_s \;=\; log_2(m) \;=\; log_2(n)$$

Pour un tel canal, le nombre m de symboles d'entrée est égal au nombre n de symboles de sortie.

Canal binaire symétrique

Dans la cas du CBS, l'information mutuelle a pour expression :

$$I(X;Y) \;=\; H(Y) \;+\; p\ log_2(p) \;+\; (1-p)\ log_2(1-p)$$

et la capacité du canal vaut :

$$C_s \;=\; 1 \;+\; p\ log_2(p) \;+\; (1-p)\ log_2(1-p)$$

Chapitre 6

Code correcteur d'erreur

6.1 Introduction

Considérons une source discrète S sans mémoire, d'entropie $H(S)$, qui émet des messages en utilisant un alphabet fini $\{A_i\}$, (i = 1, ..., n).
Chaque symbole est codé, c'est-à-dire représenté par une séquence binaire appelée mot-code. Nous avons vu qu'il est possible, grâce à des techniques de codage entropique (Shannon-Fano, Huffman, ...), de construire des codes de longueur moyenne proche de la longueur minimale $L_{min} = H(S)$.
Lorsqu'une séquence binaire est transmise par un canal de transmission, des erreurs peuvent se produirent : un "1" peut être reçu comme un "0" et inversement.
Nous allons introduire des bits supplémentaires au mot-code suivant une loi donnée. Ces bits supplémentaires, appelés bits de parité, introduisent une redondance au message binaire. Cette redondance doit permettre de détecter et de corriger les erreurs de transmission.
Dans ce chapitre nous considérons des codes dont les mots-code sont de longueur fixe k, construits sur un alphabet binaire $F_2 = \{0,1\}$.

6.2 Les codes en blocs linéaires

6.2.1 Définition

On appelle *codage en blocs*, l'opération qui consiste à associer à chaque mot-code de longueur k un mot de longueur n ($n > k$). Les $p = n - k$ bits supplémentaires associés au mot-code sont appelés *bits de parité*.

Les mots-code sont des séquences binaires (alphabet $\{0,1\}$) de longueur k. Il y a donc 2^k mots-code différents. Ils appartiennent à l'ensemble F_2^k.
Pour réaliser le codage en blocs linéaire, on ajoute p bits de parité aux mots-code de façon à obtenir des mots-blocs de longueur n. Il n'y a donc que 2^k mots-blocs différents, bien que leur longueur est n. Ils forment donc un sous-ensemble de l'ensemble F_2^n.

Un codage en blocs est donc une application g de l'ensemble F_2^k constitués par les mots-code de longueur k vers l'ensemble F_2^n

$$F_2^k \rightarrow F_2^n \qquad \textit{tel que} \qquad m \rightarrow c = g(m)$$

où $m = [\, m_0, m_1, \ldots, m_i, \ldots, m_{k-1} \,]$, mot-code de longueur k
et $c = [\, c_0, c_1, \ldots, c_j, \ldots, c_{n-1} \,]$ mot-bloc de longueur n.

Le code en blocs linéaire est noté : *C(n,k).*
Le rapport k/n est appelé le *rendement* du code ou *taux de remplissage.* La différence $(n-k)$ représente le nombre d'éléments binaires de redondance introduit par le codage.

6.2.2 Addition et multiplication dans le corps F_2

Pour les codes en blocs, les opérations de codage et décodage sont réalisées par addition et multiplication entre éléments binaires, dont nous rappelons les résultats dans la tableau ci-dessous.

a	b	a+b	ab
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

6.2.3 Matrice génératrice d'un code en blocs linéaires

Les mots-code m font partis de l'ensemble F_2^k.
Soit ($e_0, e_1, \ldots, e_i, \ \ldots, e_{k-1}$) une base de cet ensemble. Les mots-code s'écrivent sous la forme :

$$m = \sum_{i=0}^{k-1} m_i \; e_i$$

Le codage en bloc est une application $g : F_2^k \rightarrow F_2^n$.
Un mot-bloc c associé à m est égal à :

$$c \;=\; g(m) \;=\; \sum_{i=0}^{k-1} m_i \; g(e_i)$$

Soit ($e'_0, e'_1, \ldots, e'_j, \ \ldots, e'_{n-1}$) une base de F_2^n,

$$g(e_i) \;=\; \sum_{j=0}^{n-1} g_{i,j} \; e'_j$$

La matrice G associée à l'application linéaire g est une matrice de dimension (k lignes, n colonnes) :

$$\mathbf{G} = \begin{pmatrix} g_{0,0} & \cdots & g_{0,j} & \cdots & g_{0,n-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ g_{i,0} & \cdots & g_{i,j} & \cdots & g_{i,n-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ g_{k-1,0} & \cdots & g_{k-1,j} & \cdots & g_{k-1,n-1} \end{pmatrix}$$

La matrice G de dimension (k,n) est appelée *matrice génératrice du code*. Au mot-code m elle associe le mot-bloc c par la relation matricielle :

$$m \ = \ c\, G$$

Remarque
Cette matrice n'est pas unique. En effet, en permutant les vecteurs de la base $(e'_0, e'_1, \ldots, e'_j, \ \ldots, e'_{n-1})$ ou de la base $(e_0, e_1, \ldots, e_i, \ \ldots, e_{k-1})$, on obtient une nouvelle matrice G' dont les colonnes et les lignes ont été permutées.

On peut donc écrire la matrice génératrice sous la forme :

$$\mathbf{G} = (I_k,P) = \begin{pmatrix} 1 & 0 & \ldots & 0 & p_{0,1} & \cdots & p_{0,i} & \cdots & p_{0,n-k} \\ 0 & 1 & \ldots & 0 & p_{1,1} & \cdots & p_{1,i} & \cdots & p_{1,n-k} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \ldots & 1 & p_{k-1,1} & \cdots & p_{k-1,i} & \cdots & p_{k-1,n-k} \end{pmatrix}$$

Où I_k est la matrice identité de dimension (k,k) et P, de dimension $(k,n-k)$ qui permet de calculer les bits de parité. Ainsi le mot-bloc sécrit sous la forme :

$$c = [m,mP]$$

Exemple
Soit un code en blocs linéaires C(3,2) tel que :

$mot-code$	$mot-bloc$
00	000
01	011
10	101
11	110

Ce code consiste à ajouter un bit de parité tel que la somme des bits du mot-bloc soit nulle. Pour écrire la matrice génératrice du code, on considère :
une base canonique de F_2^2 : $e_0 \ = \ (1,0)$ et $e_1 \ = \ (0,1)$
une base canonique de F_2^3 : $e'_0 \ = \ (1,0,0)$, $e'_1 \ = \ (0,1,0)$ et $e'_3 \ = \ (0,0,1)$
On peut écrire :

$$g(e_0) \rightarrow 1\ 0\ 1 \ = \ 1\ e'_0 \ + \ 0\ e'_1 \ + \ 1\ e'_2$$
$$g(e_1) \rightarrow 0\ 1\ 1 \ = \ 0\ e'_0 \ + \ 1\ e'_1 \ + \ 1\ e'_2$$

La matrice génératrice du code s'écrit :

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

On peut intervertir les lignes et les colonnes de cette matrice pour obtenir G' qui est aussi génératrice du code bloc.

$$\mathbf{G'} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

6.3 Code dual et matrice de contrôle de parité

6.3.1 Définition

1. Deux vecteurs $x = (x_0, x_1, \ldots, x_i, \ldots, x_{n-1})$ et $y = (y_0, y_1, \ldots, y_i, \ldots, y_{n-1})$ sont *orthogonaux* si leur produit scalaire est nul :

$$< x.y > = x\ y^T = \sum_{i=0}^{n-1} x_i\ y_i = 0$$

2. A chaque code en blocs linéaire $C(n,k)$ on peut associer un code en blocs linéaires *dual*. Tous les mots du *code dual* sont perpendiculaires à tous les mots-bloc du code. Les mots-blocs sont des vecteurs de dimension n, donc les mots du code dual sont aussi de dimension n et forment un sous-ensemble de F_2^n. Comme il y a 2^k mots-bloc, il y a donc 2^{n-k} mots du code dual.

6.3.2 Contrôle de parité

Soit H la matrice génératrice du code dual. Cette matrice est de dimension $(n-k,\ n)$. Tout mot du code dual s'écrit $y = u\ H$.
Soit c un mot-bloc, orthogonal à tout mot du code dual. On a donc la relation : $\forall y,\ \ c\ y^T = 0$ ou encore $\forall u,\ \ c\ H^T\ u^T = 0$.

Propriété

Tout mot-bloc c du code $C(n,k)$ est orthogonal aux mots du code dual :

$$c\ H^T = 0$$

Définition

La matrice H génératrice du code dual est appelée *matrice de contrôle de parité* du code $C(n,k)$.

Propriété

Tout mot-bloc $c = m\,G$ du code $C(n,k)$ est orthogonal aux mots du code dual. $\forall c,\ c\,.\,H^{T} = 0$ ou encore $\forall m,\ m\,.\,G\,.\,H^{T} = 0$.

$$G\,H^{T} = 0$$

En utilisant l'expression de $G = (I_k,P)$, on en déduit l'expression de H, matrice de contrôle de parité :

$$H = (P^{T},I_{n-k})$$

Exemple
Soit un code $C(7,3)$ définit par sa matrice géneératrice G :

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Sa matrice de contrôle de parité est égale à :

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

Au mot-code m =[1 0 1] on associe le mot-bloc c = [1 0 1 0 0 1 1].

6.3.3 Principe de décodage

Soit c le mot-bloc émis et r le mot reçu. Supposons que le mot c soit transmis avec des erreurs :

$$r = c + e$$

e est un vecteur de dimension n dont les composantes binaires représentent les éventuelles erreurs de transmission. Une composante du vecteur e égale à 1 indique la présence d'une erreur de transmission sur la composante correspondante du mot c.
La détection d'erreur se fait en utilisant le principe d'orthogonalité de la matrice de contrôle de parité avec les mots-blocs du code $C(n,k)$.
Pour cela on calcule le *syndrome* s, qui est un vecteur de dimension $n-k$:

$$s = r\,H^{T} = (c+e)\,H^{T} = e\,H^{T}$$

Le syndrome s est nul si, et seulement si, r est un mot-bloc du code $C(n,k)$. Un syndrome non nul implique la présence d'erreur(s) de transmission.
Notons toutefois qu'un syndrome nul n'implique pas forcément l'absence d'erreur de transmission car le mot r peut appartenir au code $C(n,k)$. Il esiste donc des configurations d'erreurs indétectables.

6.3.4 Règle de décodage

Définitions

1. Soit un vecteur v composé d'éléments égaux à 1 ou 0. On appelle *poids de Hamming du mot* v, noté $P_H(\ )$, le nombre déléments non nuls de ce mot.
2. Soient deux mots u et v on appelle *distance de Hamming entre deux mots*, notée $d_H(u,v)$ le nombre d'emplacements où les deux mots possèdent des éléments binaires différents.

$$d_H(u,v) \ = \ P_H(u+v)$$

Exemple
Soient deux mots $v = [\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ ]$ et $u = [\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ ]$, leur distance de Hamming $d_H(u,v)$ est égale à 2.
$P_H(u+v) = P_H([\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ ]) = 2$.

Règle de décodage

En présence d'erreurs de transmission (syndrome s non nul) la règle de décodage va consister à rechercher un mot-bloc ĉ le plus vraissemblable, c'est-à-dire celui qui est à la distance de Hamming minimale du mot reçu r

$$\hat{c} \quad \text{tel que} \quad d_H(r,\hat{c}) \leq \ d_H(r,c) \quad \forall c \not= \hat{c} \in C$$

Exemple
Soit un code linéaire $C(6,3)$ de matrice génératrice G :

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

et de matrice de contrôle de parité H :

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

Les 8 configurations du syndrome s et les configurations d'erreurs associées e, de poids minimal sont représentées ci-dessous :

Syndrome	Configuration d'erreur
000	000000
001	000001
010	000010
100	000100
110	001000
101	010000
011	100000
111	001001

6.4 Pouvoir de détection et de correction des codes en blocs

Pour décoder les codes en blocs linéaire on recherche les mots-blocs du code qui sont à la distance de Hamming minimale du mot reçu. La distance de Hamming entre les mots-bloc du code est un paramètre pertinent pour évaluer la performance du code.

6.4.1 Définition

On appelle *distance minimale d'un code*, notée d_{min} la distance minimale entre ses mots-blocs :

$$d_{min} \ = \ Min \ d_H(c_i, \, c_j) \quad (c_i, \, c_j \in C)$$

En tenant compte du fait que le distance entre deux mots est égale au poids de leur somme, la distance minimale d'un code est égale au poids minimal de ses mots-blocs non nuls.

$$d_{min} \ = \ Min \ P_H(c_i) \quad (c_i \in C, \, c_i \neq 0)$$

6.4.2 Pouvoir de détection et de correction d'un code en blocs

Si un code en blocs linéaire possède une distance minimale égale à d_{min}, il peut *détecter* $(d_{min} - 1)$ erreurs dans un mot de n éléments binaires et *corriger* toutes les configurations de t erreurs avec $t = \frac{d_{min}-1}{2}$.

6.5 Quelques exemples de codes en blocs

6.5.1 Code de parité

Ce code est de la forme C(n,k) avec $n = k + 1$.
$c = [\, m_0, m_1, \ldots, m_i, \ldots, m_{k-1}, c_k \,]$, avec $c_k = \sum_{i=0}^{k-1} m_i$ (modulo 2).
La distance minimale de ce code est 2. Il ne permet pas la correction d'erreur mais permet de détecter toutes les erreurs en nombre impair dans un mots de n éléments binaires.
La matrice génératrice du code est :

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & \ldots & 0 & 1 \\ 0 & 1 & \ldots & 0 & 1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & 1 & 1 \end{pmatrix}$$

La matrice de contrôle de parité H se réduit à un vecteur :

$$\mathrm{H} = [1\ 1 \ldots 1]$$

6.5.2 Code à répétition

Pour ce code C(n,k), k = 1 et n = 2M + 1. Chaque élément binaire est répété un nombre impair de fois. La distance minimale entre les deux mots codes de ce code bloc est d_{min} = (2M + 1). Son pouvoir de correction est de t = 2M erreurs.
La matrice génératrice du code est :

$$\mathrm{G} = [1\ 1 \ldots 1]$$

La matrice de contrôle de parité H est :

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

Le décodage de ce code est extrèmement simple. Il suffit de déterminer le poids du mot reçcu : s'il $P_H(r) \geq (M + 1)$ alors $m = 1$; Autrement $m = 0$.

6.5.3 Code de Hamming

Pour un code de Hamming, les colonnes de la matrice de contrôle de parité sont les représentations binaires des nombres de 1 à N. Comme chaque colonne est constituée de $m = n - k$ éléments binaires, les paramètres du code de Hamming sont :

$$n = 2^m - 1 \qquad\qquad k = 2^m - m - 1$$

La distance minimale d'un code de Hamming est de 3, quelle que soit la valeur des paramètres n et k.

Soit le code de Hamming C(7, 4) avec n-k = 3.
La matrice de contrôle de parité H est :

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

La matrice génératrice du code est :

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Ce code permet de détecter deux erreurs et d'en corriger une.