UNIVERSITÉ DE NANCY

FACULTÉ DES SCIENCES

# ÉTUDE ET SIMULATION SUR ORDINATEUR DE DIFFÉRENTS PROCÉDÉS DE CODAGE POUR UN CANAL BINAIRE AVEC EFFAÇAGE

THÈSE

présentée à la

FACULTÉ DES SCIENCES
DE L'UNIVERSITÉ DE NANCY

pour obtenir le grade de

**DOCTEUR DE SPÉCIALITÉ (MATHÉMATIQUES)**

par

**Françoise MATHÉ**

soutenue le **20 Mai 1969**
devant la Commission d'Examen



JURY :
J. LEGRAS Président
M. DEPAIX, M. FÉIX } Examinateurs
MARGUINAUD Invité

UNIVERSITÉ DE NANCY

FACULTÉ DES SCIENCES

# ÉTUDE ET SIMULATION SUR ORDINATEUR DE DIFFÉRENTS PROCÉDÉS DE CODAGE POUR UN CANAL BINAIRE AVEC EFFAÇAGE

THÈSE

présentée à la

FACULTÉ DES SCIENCES
DE L'UNIVERSITÉ DE NANCY

pour obtenir le grade de

DOCTEUR DE SPÉCIALITÉ (MATHÉMATIQUES)

par

Françoise MATHÉ

soutenue le 20 Mai 1969
devant la Commission d'Examen

JURY :
J. LEGRAS Président
M. DEPAIX, M. FEIX } Examinateurs
MARGUINAUD Invité

UNIVERSITE DE NANCY — FACULTE DES SCIENCES

ANNEE SCOLAIRE 1968/69

DOYEN : M. AUBRY

ASSESSEUR : M. GAY

Doyens honoraires : MM. CORNUBERT - ROUBAULT.

Professeurs honoraires : MM. RAYBAUD - LAFFITTE - LERAY - JOLY - CAPELLE - GODEMENT - DUBREIL - L. SCHWARTZ - DIEUDONNE - DE MALLEMAN - LAPORTE - EICHBORN - LONGCHAMBON - LETORT - DODE - GAUTHIER - GOUDET - OLMER - CORNUBERT - CHAPELLE - GUERIN - WAHL.

Maîtres de Conférences honoraires : MM. LIENHART - PIERRET - Melle MATHIEU.

## PROFESSEURS

| | | |
|---|---|---|
| MM. | ROUBAULT | Géologie |
| | VEILLET | Biologie Animale |
| | BARRIOL | Chimie Théorique |
| | BIZETTE | Physique |
| | GUILLIEN | Electronique |
| | LEGRAS | Mécanique Rationnelle |
| | BOLFA | Minéralogie |
| | NICLAUSE | Chimie |
| | FAIVRE | Physique Appliquée |
| | AUBRY | Chimie Minérale |
| | COPPENS | Radiogéologie |
| | DUVAL | Chimie |
| | FRUHLING | Physique |
| | HILLY | Géologie |
| | LE GOFF | Génie Chimique |
| | SUHNER | Physique Expérimentale |
| | CHAPON | Chimie Biologique |
| | HEROLD | Chimie Minérale Industrielle |
| | SCHWARTZ B. | Exploitation Minière |
| * | MALAPRADE | Chimie |
| * | MANGENOT | Botanique |
| | GAYET | Physiologie |
| | HADNI | Physique |
| * | BASTICK | Chimie |
| | DUCHAUFOUR | Pédologie |
| | GARNIER | Agronomie |
| | NEEL | Chimie Organique Industrielle |
| | BERNARD | Géologie Appliquée |
| * | CHAMPIER | Physique |
| * | GAY | Chimie Biologique |
| | STEPHAN | Zoologie |
| * | CONDE | Zoologie |
| * | WERNER | Botanique |
| | EYMARD | Calcul différentiel & Intégral |
| | LEVISALLES | Chimie Organique |
| | FELDEN | Physique |
| * | GOSSE | Mécanique Physique |
| * | DAVOINE | Physique (ESMIN) |
| | HORN | Physique (1° cycle) |
| * | ROCCI | Géologie |
| * | Mme LUMER | Mathématiques |
| | DELPUECH | Chimie Physique |

| | |
|---|---|
| N... | Chimie Biologique |
| N... | Mécanique Appliquée |
| N... | Analyse Supérieure |
| N... | Méthodes Mathématiques de la Physique |
| N... | Mécanique Rationnelle. |

(*) professeur titulaire à titre personnel

## MAITRES DE CONFERENCES

| | |
|---|---|
| Mme BASTICK | Chimie P.C. EPINAL |
| MM. GUDEFIN | Physique |
| VUILLAUME | Psychophysiologie |
| FRENTZ | Biologie Animale |
| MARI | Chimie (ISIN) |
| AUROUZE | Géologie |
| DEVIOT | Physique du Solide |
| FLECHON | Physique P.C. |
| Mle HUET | Mathématiques C.B.G. |
| VIGNES | Metallurgie |
| BALESDENT | Thermodynamique, chimie appliquée (ENSIC) |
| BLAZY | Minéralogie Appliquée (ENSG) |
| JANOT | Physique P.C. EPINAL |
| JACQUIN | Pédologie et chimie agricoles |
| MAINARD | Physique M.P. |
| CACHAN | Entomologie appliquée (ENSA) |
| MARTIN | Chimie P.C. |
| PAULMIER | Mécanique Expérimentale |
| PEOTAS | Minéralogie |
| JOZEFOWICZ | Physico-chimie |
| JURAIN | Géologie C.B.G. |
| RIVAIL | Chimie appliquée (CUCES) |
| VILLERMAUX | Génie Chimique |
| METCHE | Biochimie appliquée (Brasserie) |
| PAIR | Mathématiques appliquées |
| BAUMANN | Physique 1° cycle |
| DURAND | Physique |
| GRANGE | Physique(ISIN) |
| DEPAIX | Probabilités et statistiques |
| RAVEREZ | Chimie (ENSIC) |
| ROQUES | Chimie Minérale |
| WEISSLINGER | Physique |
| GERL | Physique |
| HUSSON | Physique ENSEM |
| AMARIGLIO | Chimie Biologique |
| N... | Mécanique des fluides (ISIN) |
| N... | Mécanique (ISIN) |
| N... | Mathématiques |
| N... | Mathématiques P.C. |
| N... | Mathématiques C.B.G. |
| N... | Physiologie animale |
| N... | Mathématiques M.P. |
| N... | Exploitation minière (ENSMIN) |

Ce travail a été réalisé sous la direction de Monsieur FEIX, Maître de Recherche au C. N. R. S. ; je tiens à lui exprimer ma reconnaissance pour le soutien constant qu'il n'a cessé de m'accorder.

Je remercie Monsieur LEGRAS, Directeur de L'Institut Universitaire de Calcul Automatique de Nancy qui a bien voulu me faire l'honneur de présider ce jury.

Je remercie vivement Monsieur DEPAIX, chargé d'enseignement au I. U. C.A. ainsi qu'à Monsieur MARGUINAUD Ingénieur - Docteur à la Société A. E. R. O. pour avoir bien voulu participer au Jury.

J'exprime ma profonde reconnaissance à Monsieur BAUMANN qui nous a permis d'avoir accès à une I. B. M. 360.

Que Monsieur MARI, Directeur de l'I. S. I. N. , veuille bien trouver l'expression de ma gratitude pour l'accueil qu'il m'a réservée à l'I. S. I. N. et le soutien qu'il n'a cessé de m'accorder.

Ma gratitude va également à Monsieur POIRSON, au Personnel du Laboratoire de dessin et du Secrétariat de l'I. S. I. N. ainsi qu'à Mademoiselle DUSSAUSSOIS qui s'est chargée avec beaucoup d'amabilité de la dactylographie.

# ETUDE ET SIMULATION SUR ORDINATEURS, DE DIFFERENTS PROCEDES DE CODAGE POUR UN CANAL BINAIRE AVEC EFFACAGE

SOMMAIRE

3° Temps de Calcul

a) Nombre d'opérations élémentaires

- Construction des Codes
- Décodage

b) Comparaison et limites d'Utilisation du Code dictionnaire

4° Conclusions

a) Les codes aléatoires sont bons pour n grand

b) "Super Canal"

c) Meilleur schéma d'un canal physique

VII - Bibliographie.

# I Introduction

La transmission de l'Information, que ce soit par la voix humaine, par les journaux, la radio, la télévision, ou par tout autre moyen, est un des aspects de notre vie de tous les jours. Nous pouvons ainsi communiquer des renseignements, des opinions, des idées ... . Devant une telle importance et une si grande variété de communications, une théorie est alors capitale.

## 1° Bases de la Théorie de l'Information

Il faut pouvoir mesurer et chiffrer l'information, la théorie doit donc être quantitative. Il sera nécessaire :

- de définir les ensembles sur lesquels nous travaillerons : ensemble de messages possibles.

- de ne pas vouloir donner une "valeur" au message transmis : ainsi si l'ensemble de travail est celui des mots, les deux messages : "l'Energie est égale à la masse multipliée par le carré de la vitesse de la lumière" et "il pleut à Nancy depuis le début du mois : c'est le climat normal de cette ville" auront la même importance au point de vue transmission de l'Information (l'un et l'autre seront formés de 21 mots).

- de définir une mesure de l'Information : soit une situation où N événements sont possibles ; la quantité moyenne d'information attachée à cette situation sera basée sur la possibilité pour ces événements de se réaliser ; et la connaissance des probabilités $P_1$ .... $P_N$ , à priori de ces éléments permettra de donner une mesure de la quantité moyenne d'information apportée par cette situation.

L'incertitude initiale liée à une telle situation sera caractérisée par la fonction [1] :

$$H = -\sum_{i=1}^{N} P_i \log_2 P_i$$

(le logarithme est le logarithme à base 2)

H est appelée entropie ou incertitude moyenne, ou quantité moyenne d'information. L'unité d'entropie est un nombre sans dimension : le bit.

## 2° Système de Communications.

Nous pouvons schématiser tout système de communications en le séparant en trois parties essentielles :

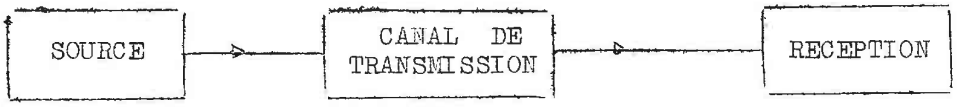



Fig. 1

- La source ou générateur de messages est un appareil qui sélecte et émet des suites de symboles appartenant à un certain alphabet X.

Nous supposerons notre système de communications discret. Toute suite de k symboles sera appelée séquence ou message.

X sera un ensemble fini de n éléments ou symboles, chacun d'entre eux ayant une probabilité $p(x_i)$ d'occurence. L'ensemble de ces probabilités définit entièrement la Source. Nous calculerons son entropie :

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

on appelle aussi H( X ) incertitude quand au caractère qui sera transmis

- à <u>la Réception</u> On reçoit des séquences formées de symboles qui peuvent être différents de ceux de la source. Ces symboles appartiennent à un alphabet Y , alphabet de la réception. Y est un ensemble fini comprenant m éléments ; chaque élément $y_j$ est reçu avec une probabilité $p(y_j)$. L'ensemble des probabilités $p(y_j)$ définit entièrment la réception. Son entropie est

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

On appelle aussi H (Y) incertitude quand au caractère qui sera reçu.

- <u>Le Canal</u> transmet les messages émis par la source. Nous le supposerons bruyant ; donc à chaque couple de symboles $x_i$ émis, $y_j$ reçu est attaché la probabilité :

$P ( x_i , y_j)$ = Probabilité d'avoir $x_i$ émis et $y_j$ reçu.

nous considérons aussi les probabilités conditionnelles :

$P( x_i \longleftarrow y_j)$ = probabilité pour que $y_j$ ayant été reçu, on ait envoyé $x_i$

$P( x_i \longrightarrow y_j)$ = probabilité pour que $x_i$ ayant été envoyé, on reçoive $y_j$

Ces probabilités sont liées aux probabilités $P( x_i)$, $P(y_j)$ et $P( x_i , y_j)$ par les relations :

$$P( x_i , y_j) = P ( x_i) \cdot P( x_i \longrightarrow y_j) = P(y_j) \cdot P ( x_i \longleftarrow y_j)$$

Donc l'ensemble des probabilités $P( x_i)$ , $P( y_j)$ , $P( x_i, y_j)$ suffisent pour définir complétement le système de communications.

Nous pouvons calculer les différentes entropies

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

= entropie relative au couple Source et Réception, ou incertitude moyenne quand au moment où X sera transmis et Y reçu.

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

= entropie conditionnelle de la Source vue de la Réception, ou incertitude moyenne du récepteur du message quant à ce qui a été réellement envoyé.

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

= entropie conditionnelle d'incertitude ou incertitude moyenne de la personne qui envoie quant à ce qui sera reçu.

Etant données les relations existant entre les probabilités nous avons :

$$H(X, Y) = H(X) + H(X \longrightarrow Y) = H(Y) + H(X \longleftarrow Y)$$

3° Capacité - Téhorème de Shannon.

On définit alors pour un tel système l'information moyenne transmise, ou rythme de transmission de l'information, ou information relative (cf RENYI)

$$\boxed{R = H(X) + H(Y) - H(X, Y)}$$

Et soit $\tau_i$ la durée du symbole $x_i$ ; c'est le temps pendant lequel $x_i$ est transmis et c'est en général le même, $\tau$ pour tous les symboles $x_i$ ; on appelle alors <u>capacité du canal</u> la grandeur C telle que

$$\boxed{C = \frac{1}{\tau} \max (R)}$$ , ce maximum étant pris par rapport à la variation des probabilités $P(x_k)$ des symboles de la source ; on suppose ici que ces probabilités sont indépendantes.

$\tau$ étant le même pour tous, nous le prendrons dans la suite égal à l'unité.

Soient des messages de longueur $n$ : ils sont formés de $n$ symboles et soit $N(n)$ le nombre de messages différents de longueur $n$ émis par la source, une autre définition de $C$ est [2] :

$$\boxed{C = \frac{1}{\tau} \lim_{n \longrightarrow \infty} \left( \frac{1}{n} \log_2 N(n) \right)}$$

On montre [3] qu'il existe des canaux qui n'ont pas de Capacité. Pour la plupart des canaux "raisonnables" $C$ existe ; nous voulons chercher à réduire l'importance des erreurs pour ces canaux. Nous savons que cela est possibles et le <u>Théorème de Shannon</u> nous dit dans quelles conditions :

> Soit un canal de transmission de Capacité C, si on veut envoyer des messages de n éléments à un rythme R de transmission tel que $R < C$ , alors pour n grand, il est possible de transmettre en rendant la probabilité de mal reconnaître un message aussi faible que possible. Si $R \geqslant C$ , ceci est impossible.

4° Le Codage est le moyen qui nous permettra de minimiser l'erreur : on codera un message émis par la source ; la réception connaît la méthode utilisée pour ce codage et celui-ci est fait de telle sorte qu'à la réception, après décodage, on retrouve le message émis, la probabilité de mal interpréter étant aussi faible que possible.

Nous ne considérerons que le cas d'une source ayant deux symboles différents. Ces symboles seront noté 0 et 1. Le canal de transmission utilisé sera dit binaire et les symboles appelés digits.

Distance de deux mots ; définition : La source binaire émet des mots de n digits. Soient deux messages (ou mots) émis $\boxed{V_1}$ et $\boxed{V_2}$. Et soit le mot $\boxed{W} = \boxed{V_1} \oplus \boxed{V_2}$ où $\oplus$ désigne la somme modulo 2 de $\boxed{V_1}$ et $\boxed{V_2}$ ; on appellera distance de $\boxed{V_1}$ et $\boxed{V_2}$ le nombre de symboles égaux à 1 de $\boxed{W}$.

Nous allons considérer dans ce qui suit deux types de codage binaire :

a) La source émet des messages de longueur n qui sont envoyés à travers le canal ; si nous voulons un bon codage il faut que R rythme de transmission soit inférieur à C ; il doit donc y avoir au maximum $2^{nC}$ messages différents ($C = \max R = \lim_{n\to\infty} \left(\frac{1}{n} \log_2 N(n)\right)$)

Ces messages seront répertoriés dans un "dictionnaire" ou cahier de code connu par l'émission et la réception ; le décodage sera une simple comparaison du message avec tous les messages du dictionnaire : on cherchera le mot le plus "proche" (distance minimum) du message reçu. C'est le code "dictionnaire".

b) Codage linéaire Tout mot de code est un vecteur de longueur n

Définition : Soit W un espace vectoriel de dimension n, défini sur le corps K . Ici K sera le corps des entiers modulo 2 ; on appelle code linéaire G (n, k) un sous espace vectoriel de W, de dimension k. Tout vecteur de G sera un mot de code et une combinaison linéaire de k vecteurs indépendants formés sur K.

Le corps K est le corps des entiers modulo 2.
Donc toutes les opérations se feront modulo 2.

On appelle la matrice $[G]$ dont les lignes sont les k vecteurs de génération de G, matrice de Génération du Code.

Pour engendrer tout le code G nous avons k coefficients qui peuvent prendre chacun deux valeurs, G a donc $2^k$ mots de code de longueur n différents et le rythme R de transmission est égal à $\frac{k}{n}$. Ces k coefficients sont appelés digits d'information ; ils forment un vecteur $[I]$. Tout mot de code $[V]$ de longueur n sera tel que :

$[V] = [I] \cdot [G]$ (modulo 2) $[I]$ et $[V]$ sont des vecteurs lignes.

Le codage se fait de la manière suivante : $[G]$ matrice de génération est connue par l'émission et la réception ; la source émet un mot $[I]$ de longueur k ; on calcule le message $[V]$ de longueur n qui sera envoyé ; à la réception on cherchera à retrouver le mot $[I]$ émis. Pour $R < C$ nous pourrons le faire avec une probabilité de mal reconnaître $[I]$ aussi faible que possible, si n est suffisamment grand.

Nous nous proposons d'étudier un certain canal de transmission : canal binaire à effacement ou B. E. C. ; cette étude sera faite en simulant le canal, la source, la réception sur ordinateur, ainsi que les méthodes de codage et décodage utilisées.

On cherchera dans quelles mesures il est possible d'améliorer le codage.

# II Définition du Système de communications simulé.

## 1° Canal binaire à effaçage - Capacité.

Le canal binaire à effaçage ou B. E. C. est un canal qui a deux symboles d'entrée et trois de sortie.

La source a les symboles 0 et 1

La réception 0, 1 et E

Si un symbole (0 ou 1 ) est envoyé, la probabilité pour que ce même symbole (0 ou 1) soit reçu est 1 - P ; la probabilité pour que l'on reçoive E est P. La figure 2 précise les différentes probabilités conditionnelles. Dans la suite nous parlerons indifféremment de symboles ou de digits pour 0 et 1 ; E sera l'effaçage.

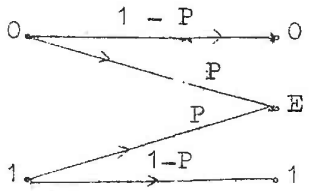



Fig. 2

Soit $\alpha$ = probabilité d'émission de 1

$1-\alpha$ = probabilité d'émission de 0

Calculons le rythme de transmission R ( $\alpha$ ) et cherchons son maximum C quand $\alpha$ varie

Mettons dans un tableau les probabilités $P(x_i , y_j)$

RECEPTION

| SOURCE Symboles | 0 | 1 | E | $P(x_i)$ |
|---|---|---|---|---|
| 0 | $(1-P)(1-\alpha)$ | 0 | $(1-\alpha)P$ | $1-\alpha$ |
| 1 | 0 | $\alpha(1-P)$ | P | $\alpha$ |
| $Py_j$ | $(1-P)(1-\alpha)$ | $\alpha(1-P)$ | P | |

$$H(X) = -(1-\alpha)\log_2(1-\alpha) - \alpha\log_2(\alpha)$$

$$H(Y) = -(1-P)(1-\alpha)\log_2(1-\alpha)(1-P) - \alpha(1-P)\log_2\alpha(1-P) - P\log_2 P$$

$$H(X,Y) = -(1-P)(1-\alpha)\log_2(1-P)(1-\alpha) - \alpha(1-P)\log_2\alpha(1-P) - (1-\alpha)P\log_2(1-\alpha)P - \alpha P\log_2\alpha P$$

$$R(\alpha) = H(X) + H(Y) - H(X,Y) = -(1-\alpha)\log_2(1-\alpha) - \alpha\log_2\alpha + (1-\alpha)P\log_2(1-\alpha)P + \alpha P\log_2\alpha P - P\log_2 P$$

$$R(\alpha) = (P-1)\left\{(1-\alpha)\log_2(1-\alpha) + \alpha\log_2\alpha\right\}$$

$R(\alpha)$ est maximum pour $R'(\alpha) = 0$

$$R'(\alpha) = (P-1)\left\{\log_2\alpha - \log_2(1-\alpha)\right\}$$

$$R'(\alpha) = 0 \quad \text{pour} \quad \frac{\alpha}{1-\alpha} = 1 \quad \text{soit} \quad \alpha = \frac{1}{2}$$

alors $$\boxed{R\left(\frac{1}{2}\right) = C = 1 - P}$$

Nous trouvons le résultat intéressant : La capacité est égale au pourcentage des digits d'entrée qui ne sont pas effacés et ceci nous permet de commenter le théorème de Shannon en nous en montrant clairement sa puissance : on envoie N digits ; N (1 - P) sont reçus , N P perdus après effaçage ; sans codage on récupère bien N(1 - P) digits mais au départ nous ne savons pas quels seront ceux qui vont être perdus et le hasard frappe aveuglement parmi les N digits.

Shannon dit : on peut déchiffrer en envoyant un peu moins que N (1 - P) digits plus une redondance d'un peu plus de N P digits. C'est à dire que dans un certain sens on peut s'arranger pour que ce soient les N(1 - P) digits bien déterminés qui puissent être retrouvés. En fait, le codage permet de préciser sur les N digits, les N(1 - P) qui seront conservés.

2° Codage.

Nous allons chercher par le codage à réduire le nombre de messages mal reçus ; nous savons que cela est possible pour n grand et $R < C$.

Nous utiliserons un code linéaire G ( n , k ) de matrice de génération $[G]$ . Par des combinaisons de lignes et de colonnes sur $[G]$ nous obtenons la matrice $[G']$ de la forme

$$[G'] = \begin{bmatrix} 1 & \cdots & 0 & b_{11} & \cdots & b_{1,\,n-k} \\ & \ddots & & & & \\ 0 & & 1 & b_{i1} & \cdots & b_{i,\,n-k} \\ & & & & & \\ & & & b_{k1} & \cdots & b_{k,\,n-k} \end{bmatrix} = \begin{bmatrix} [I_k] & , & [B] \end{bmatrix}$$

où $[I_k]$ est la matrice unité de rang k

$[G']$ est appelé forme réduite de $[G]$ et les k vecteurs lignes de $[G']$ sont indépendants et engendrent le même espace vectoriel G . La matrice de génération utilisée sera $[G']$ plutôt que $[G]$ .

Soit $[I]$ un mot d'information émis et $[V]$ le message envoyé :

$$[V] = [I]\,[G'] = [I]\,\begin{bmatrix} [I_k] & , & [B] \end{bmatrix} = \begin{bmatrix} [I] & , & [C] \end{bmatrix}$$

où le vecteur ligne $[C]$ est composé des k digits calculés de la manière suivante :

$[C] = [I] \cdot [B]$ ces opérations se faisant modulo 2.

Soit $[A]$ la matrice transposée de $[B]$

$[A]$ est dite <u>matrice de parité du code</u> ; elle a k colonnes et n - k lignes toutes composées de digits ( $a_{ij} = 0$ ou 1) ; elle permet de calculer le vecteur $[C]$ ; qui avec le vecteur $[I]$ d'information forme le message envoyé.

Posons $m = n - k$

$\boxed{I} = \left[ I_1 \ \ldots \ I_k \right]$ sont les digits d'informations

$\boxed{C} = \left[ C_1 \ \ldots \ C_m \right]$ sont dits digits de parité

$\boxed{A}$ étant la matrice de parité du code G (n , k) utilisé

$\boxed{A} = \left[ a_{ij} \right]$ à k colonnes et m lignes

et $$C_i = \sum_{j=1}^{k} a_{ij} \ I_j \qquad i = 1, \ \ldots \ , \ m \ (\text{mod } 2)$$

Un message ainsi codé est envoyé à travers le canal. A la réception certains digits sont effacés. Soient $k_1$ parmi les digits d'information et $m_1$ parmi ceux de parité. Pour décoder nous aurons à résoudre un système linéaire de $m - m_1$ équations à $k_1$ inconnues sur le corps des entiers modulo 2.

A partir de cette considération nous pouvons retrouver la capacité C du canal : P étant la probabilité d'effaçage, la valeur moyenne de $k_1$ est Pk, celle de $m_1$ , P(n - k). Il faut donc pour pouvoir résoudre le système précédent que $m - m_1 \geqslant Pk$

Soit $n - k - P(n - k) \geqslant Pk$

ou $$\frac{n - k}{k} \geqslant \frac{P}{1 - P}$$

soit $$\frac{n}{k} \geqslant 1 + \frac{P}{1 - P} = \frac{1}{1 - P}$$

Or le rythme de transmission est $R = \frac{k}{n}$

On aura donc $\boxed{R \leqslant 1 - P}$ ce qui permet de retrouver C sachant que $R \leqslant C$.

Lorsque nous aurons pu résoudre le système de $m - m_1$ équations à $k_1$ inconnues, nous aurons retrouvé le message de longueur k émis par la source.

## III Simulation du B. E. C.

1° Emission et Codage.

Toute information émise est une suite de k digits. On suppose que les probabilités d'émission de 0 et 1 sont les mêmes : soit $\frac{1}{2}$ ; dans ce cas R , rythme de transmission pourra être maximum et égal à C.

On veut envoyer cette information à travers le canal ; il y aura des digits effacés et la probabilité d'effaçage est P. Nous utiliserons un codage linéaire où les k premiers digits du mot codé sont ceux d'information ; si nous voulons un codage efficace, il faudra prendre m = n - k digits de parité, tel que $\frac{k}{n} < 1 - P$.

Pour effectuer le codage on se donne la matrice $[A]$ de parité du code ; $[A]$ dépend du type de codage linéaire étudié et à m lignes et k colonnes . $[A]$ est connue par le codeur et le décodeur et ne contient que des 0 ou des 1.

Soit $[I]$ le vecteur information ; $[I]$ a k digits

On calcule le vecteur $[C]$ des m digits de parité :

$[C] = [A] . [I]$

où $[C]$ et $[I]$ sont des vecteur colonnes et les opérations effectuées modulo 2 . Le message émis $[I]$ est gardé en mémoire dans le vecteur $[U]$

Le message envoyé à travers le canal sera le vecteur $[I]$ suivi du vecteur $[C]$

2° Transmission

Au cours de la transmission certains digits sont effacés : soient $k_1$ dans $[I]$ et $m_1$ dans $[C]$ . On suppose que l'effaçage est aléatoire, donc qu'il n'y a pas de liaisons entre les digits du message.

3° Décodage et Réception

On essaie de retrouver le vecteur $|\overline{I}|$ émis, donc les digits effacés dans $|\overline{I}|$ , connaissant la matrice $|\overline{A}|$ de parité du code.

Soient $I_1 \ldots\ldots I_k$ les digits d'information

$C_1 \ldots\ldots C_m$ ceux de parité

Ils sont liés par la relation :

$$|\overline{C}| = |\overline{A}|\,|\overline{I}| \quad \text{(modulo 2)}$$

soit

$$\boxed{C_j = \sum_{i=1}^{k} a_{ji}\, I_i \qquad j = 1, \ldots, m \quad \text{(mod. 2)}} \qquad (1)$$

nous avons $m - m_1$ digits de parité non effacés

soient $I_{l_1} \ldots\ldots I_{l_{k_1}}$ les digits d'information effacés

$C_{l_1} \ldots\ldots C_{l_{m-m_1}}$ les digits de parité non effacés

posons $m_2 = m - m_1$

et ne gardons dans les équations (1) que celles pour lesquelles $C_j$ n'est pas effacé. Cela nous donne le système d'équations :

$$\begin{cases} C_{l_j} = \sum_{i=1}^{k} a_{l_j,i}\, I_i \quad \text{(mod 2)} \\ j = 1, \ldots, m_2 \\ l_j \text{ est l'indice de } C_{l_j} \text{ digit de parité non effacé} \end{cases}$$

Dans ces équations certains des $I_i$ sont inconnus et en changeant la forme d'écriture nous pouvons les faire apparaître :

$$\begin{cases} C_{l_j} = \sum_{i=1}^{k_1} \left( a_{l_j, l_i} \; I_{l_i} \right) + \sum_{s=1}^{k-k_1} \left( a_{l_j, l_s} \cdot I_{l_s} \right) \pmod 2 \\ j = 1, \ldots, m_2 \\ l_i = \text{indice de } I_{l_i} \quad \text{digit d'information effacé} \\ l_s = \text{indice de } I_{l_s} \quad \text{digit d'information non effacé} \end{cases}$$

en faisant passer d'un même côté du signe égal tout ce qui est connu, on obtient :

$$\begin{cases} \sum_{i=1}^{k_1} \left( a_{l_j, l_i} \cdot I_{l_i} \right) = C_{l_j} + \sum_{s=1}^{k-k_1} \left( a_{l_j, l_s} \cdot I_{l_s} \right) \pmod 2 \\ j = 1, \ldots, m_2 \end{cases}$$

posons

$$\begin{cases} C'_j = C_{l_j} + \sum_{s=1}^{k-k_1} \left( a_{l_j, l_s} \cdot I_{l_s} \right) \pmod 2 \\ a'_{j,i} = a_{l_j, l_i} \end{cases}$$

nous obtenons le système (2) de $m_2$ équations à $k_1$ inconnues $I_{l_1} \ldots I_{l_{k_1}}$ sur le corps des entiers modulo 2.

$$\boxed{\sum_{i=1}^{k_1} \left( a'_{j\,i} \cdot I_{l_i} \right) = C'_j \pmod 2 \quad j = 1, \ldots, m_2} \qquad (2)$$

Nous allons chercher à résoudre ce système : nous savons qu'il a une solution : les valeurs initiales des digits $I_k$ lorsqu'ils ont été émis. Donc le système (2) ne sera jamais impossible ; il pourra être indéterminé si $m_2$ nombre d'équations est inférieur à $k_1$ nombre d'inconnues ou s'il existe trop de combinaisons linéaires entre les équations.

L'opération de décodage se traduira par une transformation de la matrice $|\overline{A}|$ et du vecteur $|\overline{C}|$ en une matrice $|\overline{A'}|$ (à $m_2$ lignes et $k_1$ colonnes) et en un vecteur $|\overline{C'}|$ ($m_2$ digits). Pour résoudre le système (2) nous chercherons à triangulariser la matrice $|\overline{A'}|$ :

- S'il y a plus d'inconnues que d'équations ceci sera impossible ; soit $k_2$ le nombre d'inconnues en trop ; nous prendrons la décision d'affecter aux $k_2$ premieres inconnues, la valeur 0 ou 1 aléatoirement. Nous reconstruirons $|\overline{A'}|$ qui aura alors autant de lignes que de colonnes. Nous reconstruirons aussi $|\overline{C'}|$.

- Si il y a trop de combinaisons linéaires entre les équations, cela se traduira au cours du calcul par une ligne nulle et la colonne de même indice nulle (ou en partie : à partir de la diagonale principale) ; nous prendrons la décision d'affecter à l'inconnue de même indice, aléatoirement la valeur 0 ou 1 . Nous reformerons alors $|\overline{A'}|$ et $|\overline{C'}|$ correspondants et continuerons le calcul.

La matrice $|\overline{A'}|$ étant triangularisée nous calculerons les inconnues. Nous aurons une solution possible du système (2). Nous replacerons ces digits dans $|\overline{I}|$ et comparerons $|\overline{I}|$ décodé au mot $|\overline{U}|$ réellement émis.

4° Sous Programme E N G E.

Le sigle ENGE vient du verbe engendrer.

Suivant les cas nous cherchons à avoir aléatoirement :

- ou N digits avec une égale probabilité d'apparition pour 0 et 1 ;

- ou des effacements dans un message de longueur n ; la probabilité d'effaçage étant P.

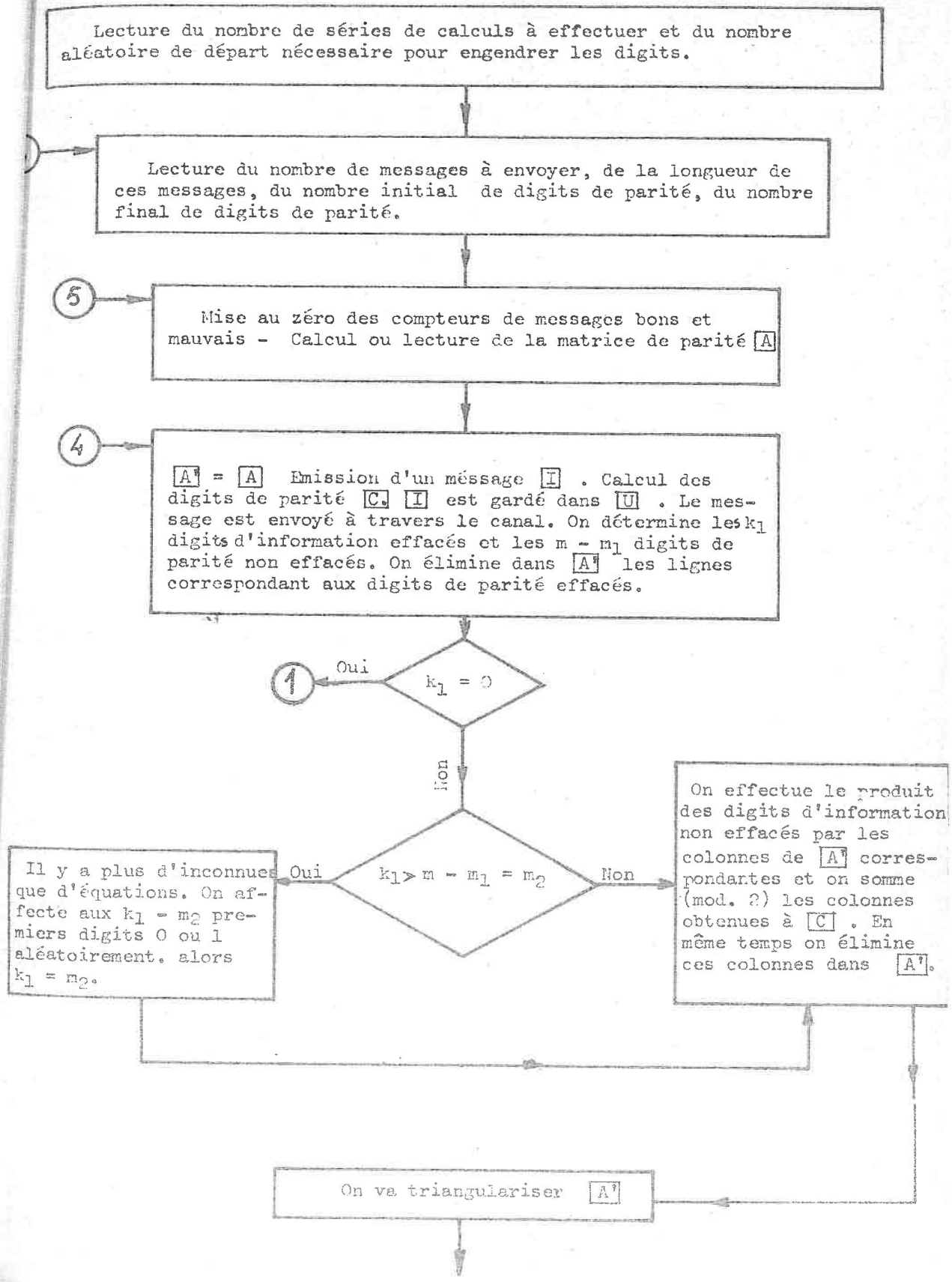
Nous utiliserons le même sous programme ENGE car dans les deux cas, nous commençons par engendrer au hasard des nombres de 8 chiffres significatifs (*) , compris entre 0 et 1 et suivant leur

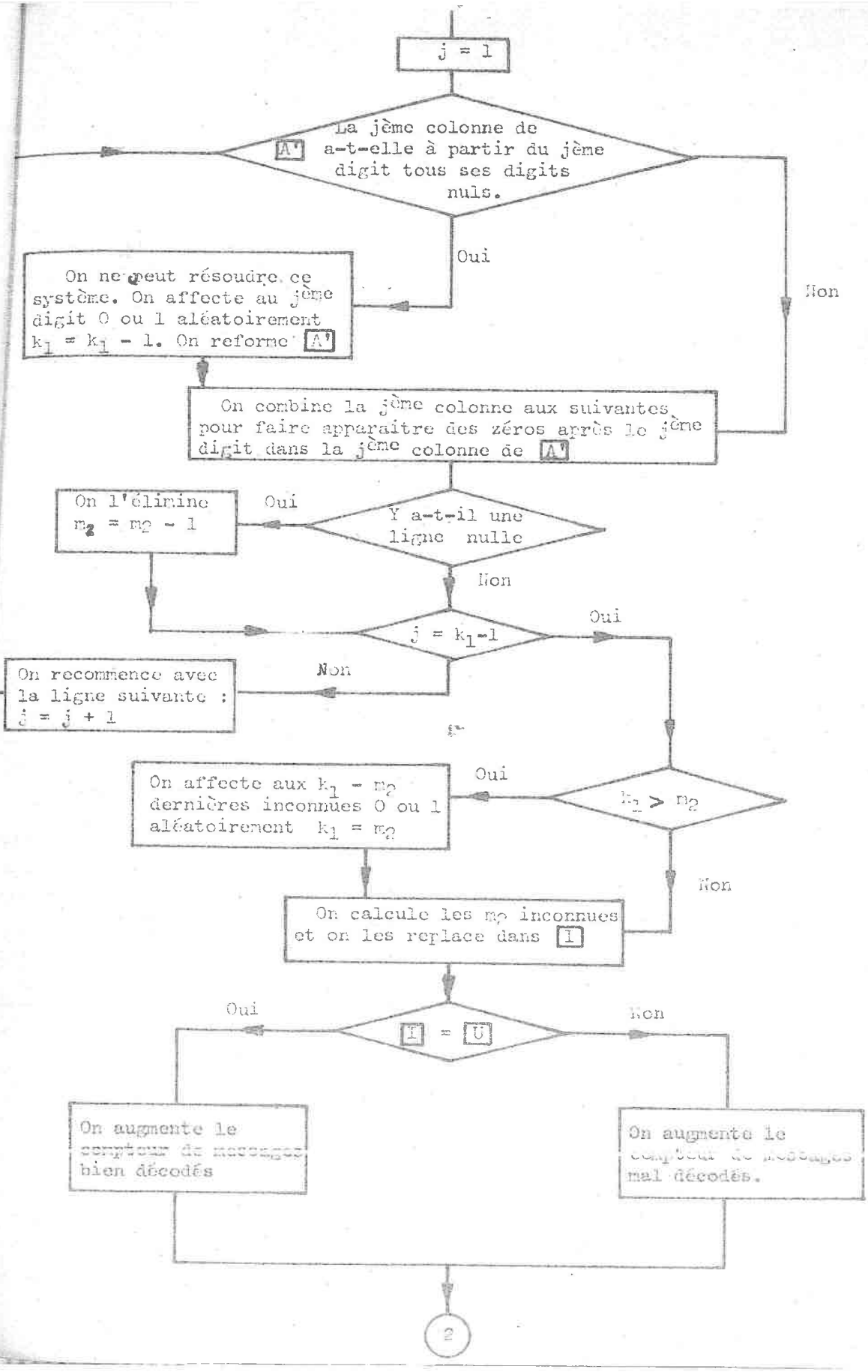
*GIANNESINI Thèse (Oct 62) - "Présentation et mise au point d'un test non paramétrique . Génératrice de nombres pseudo - aléatoires".

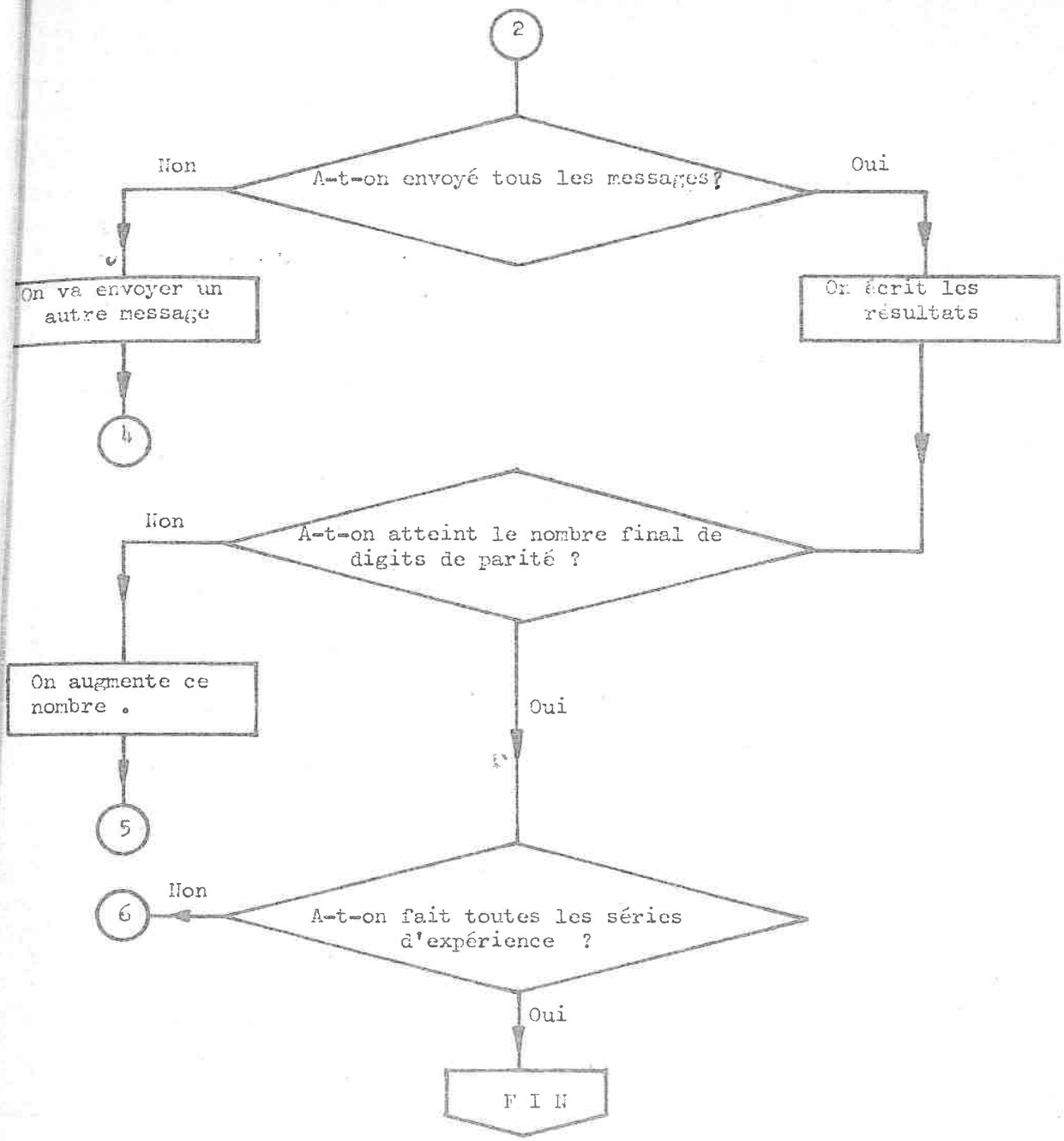
valeur, dans le premier cas par rapport à 0.5 nous obtenons le digit 0 ou 1 , dans le deuxième cas par rapport à P nous avons ou n'avons pas effaçage.

5° Organigramme de Simulation

Il utilise le sous programme défini ci-dessus.
Nous en donnons l'organigramme à la page.suivante


Lecture du nombre de séries de calculs à effectuer et du nombre aléatoire de départ nécessaire pour engendrer les digits.
Lecture du nombre de messages à envoyer, de la longueur de ces messages, du nombre initial de digits de parité, du nombre final de digits de parité.
5
Mise au zéro des compteurs de messages bons et mauvais - Calcul ou lecture de la matrice de parité [A]
4
[A'] = [A] Emission d'un méssage [I] . Calcul des digits de parité [C]. [I] est gardé dans [U] . Le message est envoyé à travers le canal. On détermine les $k_1$ digits d'information effacés et les $m - m_1$ digits de parité non effacés. On élimine dans [A'] les lignes correspondant aux digits de parité effacés.
1
Oui
$k_1 = 0$
Non
Il y a plus d'inconnues que d'équations. On affecte aux $k_1 - m_2$ premiers digits 0 ou 1 aléatoirement. alors $k_1 = m_2$.
Oui
$k_1 > m - m_1 = m_2$
Non
On effectue le produit des digits d'information non effacés par les colonnes de [A'] correspondantes et on somme (mod. 2) les colonnes obtenues à [C] . En même temps on élimine ces colonnes dans [A'].
On va trianguliariser [A']


j = 1
La jème colonne de A' a-t-elle à partir du jème digit tous ses digits nuls.
Oui
Non
On ne peut résoudre ce système. On affecte au jème digit 0 ou 1 aléatoirement $k_1 = k_1 - 1$. On reforme A'
On combine la jème colonne aux suivantes pour faire apparaître des zéros après le jème digit dans la jème colonne de A'
Y a-t-il une ligne nulle
Oui
On l'élimine $m_2 = m_2 - 1$
Non
$j = k_1 - 1$
Oui
Non
On recommence avec la ligne suivante : $j = j + 1$
$k_1 > m_2$
Oui
On affecte aux $k_1 - m_2$ dernières inconnues 0 ou 1 aléatoirement $k_1 = m_2$
Non
On calcule les $m_2$ inconnues et on les replace dans I
I = U
Oui
Non
On augmente le compteur de messages bien décodés
On augmente le compteur de messages mal décodés.
2


2
A-t-on envoyé tous les messages?
Non
Oui
On va envoyer un autre message
On écrit les résultats
4
A-t-on atteint le nombre final de digits de parité ?
Non
Oui
On augmente ce nombre .
5
6
A-t-on fait toutes les séries d'expérience ?
Non
Oui
F I N

## IV Etude des codes aléatoires de parité par blocs.

1° Définition

Définition : On appelle Code aléatoire de parité par bloc un code linéaire tel que la matrice $\boxed{A}$ de parité du code soit construite aléatoirement avec une égale probabilité d'apparition pour 0 et 1.

$$\boxed{A} = \left[ a_{ij} \right] \quad \text{et} \quad a_{il} = \begin{cases} 0 \\ 1 \end{cases} \quad \text{avec} \quad P(0) = P(1) = 0.5$$

Shannon [4] et Elias [5] ont montré que le codage aléatoire est aussi bon que tout autre moyen de codage pour des rythmes voisins de la Capacité, mais que pour des rythmes de transmission très faibles il existe des codes qui sont meilleurs.

Nous voulons par cette simulation vérifier les propriétés des codes aléatoires

Pour chaque valeur de k, n, P nous avons émis un certain nombre de messages et les avons envoyés à travers le canal ; un compteur de messages mal décodés nous donne une évaluation des performances du code, mesurées par Q , pourcentage de messages mal décodés. La méthode de Décodage utilisée est celle définie au chapître précédent.

2° Variations de Q à k , P fixés

$R = \frac{k}{n}$ étant le rythme de transmission (k = longueur de l'information, n longueur totale du message), C = 1 - P (P probabilité d'effaçage) étant la capacité, nous avons fait une première série d'expériences pour voir comment variait Q en fonction de R / C.

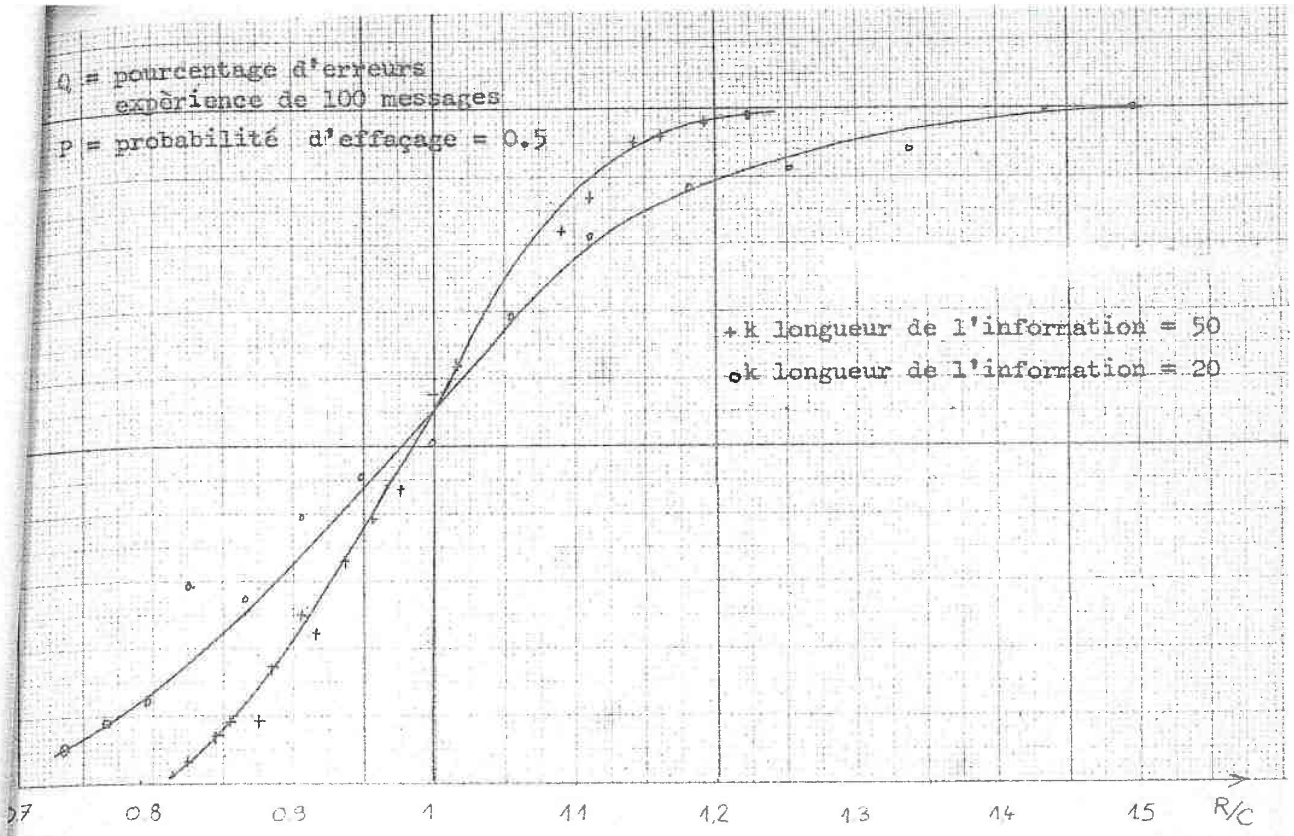



Fig. 3

k = 20

| n - k | R/C | Q |
|---|---|---|
| 6 | 1.54 | 0.99 |
| 8 | 1.43 | 0.98 |
| 10 | 1.334 | 0.92 |
| 12 | 1.25 | 0.89 |
| 14 | 1.18 | 0.86 |
| 16 | 1.11 | 0.79 |
| 18 | 1.056 | 0.67 |
| 20 | 1. | 0.49 |
| 22 | 0.95 | 0.44 |
| 24 | 0?91 | 0.38 |
| 26 | 0.87 | 0.26 |
| 28 | 0.83 | 0.28 |
| 30 | 0.8 | 0.11 |
| 32 | 0.77 | 0.08 |
| 34 | 0.74 | 0.04 |

k = 50

| n - k | R/C | Q |
|---|---|---|
| 26 | 1.28 | 0.99 |
| 30 | 1.25 | 0.99 |
| 32 | 1.22 | 0.97 |
| 34 | 1.19 | 0.96 |
| 36 | 1.16 | 0.94 |
| 38 | 1.14 | 0.93 |
| 40 | 1.11 | 0.85 |
| 42 | 1.09 | 0.80 |
| 44 | 1.06 | 0.79 |
| 46 | 1.04 | 0.70 |
| 48 | 1.02 | 0.60 |
| 50 | 1. | 0.56 |
| 52 | 0.98 | 0.42 |
| 54 | 0.96 | 0.38 |
| 56 | 0.94 | 0.32 |
| 58 | 0?92 | 0.21 |
| 60 | 0.91 | 0.24 |
| 62 | 0.89 | 0.16 |
| 64 | 0.88 | 0.08 |
| 66 | 0.86 | 0.08 |
| 68 | 0.85 | 0.06 |
| 70 | 0.83 | 0.01 |
| 72 | 0.82 | 0.00 |

Tableau 1 (P = 0.5)

k et P étant fixés, nous avons pour chaque valeur de n construit la matrice $[\bar{A}]$ de parité qui a été utilisée pour coder 100 messages émis par la source.

Nous avons pris P = 0.5 et deux valeurs de k : k = 20 , R / C varie de 0.74 à 1.54 ; k = 50 , R/C varie de 0.82 à 1.28 . Nous avons comparé les résultats obtenus (Tableau 1) en traçant les courbes Q en fonction de R/C (Fig.3).

On peut lisser les points obtenus pour chaque valeur de k par une courbe ; dès que $R < C$ , Q décroit assez rapidement vers des valeurs faibles. Nous voyons de plus que la pente de la courbe pour R/C = 1 est plus raide pour k = 50 , que pour k = 20 ; cela ne contredit pas le théorème de Shannon : pour $R < C$ quand n augmente, le pourcentage de messages mal décodés tend vers 0.

Nous remarquons, surtout pour k = 20 que pour $0.8 \leqslant R/C \leqslant 1$ des matrices de code peu performantes peuvent augmenter considérablement Q (c'est le cas de (n - k = 28 ; R/C = 0.83)) qui se trouve donner de moins bonnes performances que (n - k = 26 ; R/C = 0.86 )

Il serait intéressant de savoir comment se comporte l'erreur après décodage ou probabilité de mal interpréter un message reçu.

3° Origines de l'erreur après décodage.

D'après la méthode utilisée nous voyons que l'erreur après décodage a deux origines.

a) Origine due au fait que l'effacement est aléatoire

Les mots sont de longueur n et ont k digits d'information ; la probabilité d'effaçage est P ; on ne pourra décoder s'il y a plus de n - k effaçages dans un mot(ou message). Donc la probabilité minimum de mal décoder est

$$Q_{min} = \sum_{j=n-k+1}^{n} C_n^j \, P^j \, (1 - P)^{n-j}$$

En fait nous avons réduit cette probabilité par la méthode de décodage utilisée : chaque fois qu'il y a plus de $n - k$ effaçages, nous ne rejetons pas le message mais affectons une valeur (0 ou 1 avec une égale probabilité) aux digits qui ont été effacés en trop.

Soit $j$ effaçages et $j > n - k$

nous choisirons donc aléatoirement $j - n + k$ digits ; la probabilité d'avoir les $j - n + k$ digits justes est $\left(\frac{1}{2}\right)^{j-n+k}$ ; alors la probabilité de se tromper pour ces digits est $1 - \left(\frac{1}{2}\right)^{j-n+k}$

donc la probabilité minimum de mal décoder est pour notre méthode de décodage

$$(3) \qquad Q^{*} = \sum_{j=n-k+1}^{n} \left(1 - \left(\frac{1}{2}\right)^{j-n+k}\right) \cdot C_n^j \cdot P^j \cdot (1 - P)^{n-j}$$

b) Origine due à la nature de la matrice A

Supposons que dans le mot de longueur n, nous ayons à la réception , $n - k$ effacements au plus; il y en aura $k_1$ parmi les digits d'information et $m_1$ parmi ceux de parité.

En décodant il faudra résoudre un système linéaire de $k_1$ inconnues à $k_1$ (et peut être plus) équations, sur le corps des entiers module 2. Il faudra donc pour que ce soit possible qu'il y ait $k_1$ équations indépendantes. Cela se traduira pour la matrice $\boxed{A'}$ que l'on cherche à trianguliariser, par une indépendance linéaire de ses $k_1$ colonnes, ou par une indépendance linéaire de $k_1$ de ses lignes.

En conclusion si nous voulons un code qui corrige de une à $t$ erreurs dans l'information (à P , k fixés, on a calculé n de telle sorte que ceci soit possible) il faut construire la matrice $\boxed{A}$ de parité du code de telle sorte qu'on ne puisse former avec ses

lignes (et ses colonnes) de combinaisons linéaires nulles de $\underline{t}$ lignes (et colonnes) sur le corps des entiers modulo 2 c'est-à-dire de telle sorte que toute sous matrice carrée de dimension t de $|\underline{A}|$ soit régulière.

4° Comportement de Q à R , P fixés quand n augmente

Nous avons une borne inférieure $Q^*$ de Q ; il reste à trouver une borne supérieure. Shannon [4] a trouvé une borne supérieure de Q pour les codes aléatoires et Elias [5] a montré que cette borne était la même pour les codes aléatoires de parité par blocs.

Le "codage aléatoire" signifie que $2^{n\,R}$ (R rythme de transmission, n longueur du message) messages sont sélectionnés parmi les $2^n$ séquences possibles ; les séquences sélectionnées sont indépendantes les unes des autres et la sélection se fait au hasard avec une égale probabilité assignée à chaque séquence. $Q_{av}$ moyenne des probabilités Q pour tous ces codes est certainement supérieure à la probabilité $Q_b$ de mal décoder du meilleur d'entre eux. Dans les références citées ci-dessus, Shannon et Elias ont évalué $Q_{av}$.

$$Q_{av} \leqslant P^{n-k} \cdot (1-P)^{k} \cdot C_k^n \left\{ \frac{P\,k}{n-k-nP} + \frac{1}{1-\left(\frac{1-P}{P}\right)\cdot\frac{n-k}{2k}} \right\} = Q_1$$

On a les relations

$$Q^* \leqslant Q_b \leqslant Q_{av} \leqslant Q_1$$

Soit $A^* = \lim_{n\to\infty} \left( \frac{\log_2 Q^*}{n} \right)$ ; $A_{av} = \lim_{n\to\infty} \left( \frac{\log_2 Q_{av}}{n} \right)$ ;

$$A_1 = \lim_{n\to\infty} \left( \frac{\log_2 Q_1}{n} \right)$$

nous avons aussi $A^{*} \leqslant A_b \leqslant A_{av} \leqslant A_1$

et nous allons chercher si $A^{*}$ et $A_1$ tendent vers une limite quand $n \longrightarrow \infty$ , à $R = \frac{k}{n}$ , P fixés.

α Calcul de $A^{*}$

$$Q^{*} = \sum_{j-k+1}^{n} \left(1 - \left(\frac{1}{2}\right)^{j-n+k}\right) \cdot C_n^j \, P^j (1-P)^{n-j} \geqslant \frac{C_n^{n-k+1} \, P^{n-k+1} \cdot (1-P)^{k-1}}{2}$$

$Q^{*} = \sum_{j=n-k+1}^{n}$ (termes positifs) , est en particulier supérieur ou égal au premier terme .

$$Q^{*} \geqslant \frac{P}{2(1-P)} \cdot C_{n-k}^{n} \cdot \frac{k \cdot (1-P)^{k} \, P^{n-k}}{n-k+1}$$

On utilise l'inégalité de Stirling

$$(4) \quad \boxed{C_{n-k}^{n} = C_{n(1-R)}^{n} \approx \frac{R^{-nR} \, (1-R)^{-n(1-R)}}{\sqrt{2\pi n(1-R)\,R}} = \frac{2^{nH(R)}}{\sqrt{2\pi nR(1-R)}}}$$

où $H(R) = H(1-R) = -R \log_2 R - (1-R) \log_2 (1-R)$

quand R varie entre 0 et 1 , H(R) croît de 0 à 1 (pour R = 0.5) et decroît ensuite jusqu'à 0.

$$P^{n-k} (1-P)^{k} = 2^{n \left\{ \log_2 P + R \log_2 \frac{1-P}{P} \right\}}$$

alors $$\frac{\log_2 Q^{*}}{n} \geqslant H(1-R) + \log_2 P + R \log_2 \frac{1-P}{P} - \frac{\log_2 \left\{ 2\pi nR(1-R) \right\}}{2n}$$

$$+ \frac{\log_2 \frac{Pk}{2(1-P)} - \log_2 (n-k+1)}{n}$$

quand $n \longrightarrow \infty$ $\quad \dfrac{\log_2 \dfrac{P\,k}{2(1-P)}}{n} \longrightarrow 0$ ;

$\dfrac{\log_2 \left( 2\,\pi\, R(1-R) \right)}{2\,n} \sim \dfrac{\log_2 n}{2\,n} \to 0$ et $\dfrac{\log_2 (n-k+1)}{n} \sim \dfrac{\log_2 n}{n} \longrightarrow 0$

et $A^* \geq H(1-R) + \log_2 P + R \log_2 \dfrac{1-P}{P} = A^{**}$

β Calcul de $A_1$

Nous utilisons la relation (4)

alors $Q_1 \approx \dfrac{2^{n\left\{H(1-R) + \log_2 P + R \log_2 \frac{1-P}{P}\right\}}}{\sqrt{2\,\pi\, n\,(1-R)\,R}} \left\{ \dfrac{P.k}{n(1-P)-k} + \dfrac{1}{1-\frac{1-P}{P}.\left(\frac{n-k}{2k}\right)} \right\}$

et $A_1 = \lim_{n \longrightarrow \infty} \left( \dfrac{\log_2 Q_1}{n} \right) = H(1-R) + \log_2 P + R \log_2 \dfrac{1-P}{P}$

Nous avions $A^{**} \leq A^* \leq A_b \leq A_{av} \leq A_1$

Or $A_1 = A^{**}$

Donc $A_{av} = A_b = H(1-R) + \log_2 P + R \log_2 \dfrac{1-P}{P} = A$

γ Signe de A

Pour notre canal la capacité est $C = 1 - P < 1$

exprimons dans A , P enfonction de C.

$A = H(1-R) + \log_2 P + R \log_2 \dfrac{1-P}{P} = H(1-R) + \log_2 (1-C) + R \log_2 \dfrac{C}{1-C}$

Soit $A = R \log_2 \dfrac{C}{R} + (1-R) \log_2 \dfrac{1-C}{1-R}$

C est fixé, nous avons $R < C$ si nous voulons un bon codage

Soit $\frac{d A}{d R} = \log_2 \frac{C}{1-C} - \log_2 \frac{R}{1-R}$

$R < C$ alors $1 - C < 1 - R$ et $\frac{1}{1-C} > \frac{1}{1-R}$

donc $\frac{d A}{d R} > 0$ et A <u>croît</u> quand R varie de 0 à C.

Pour $R = C$ A est donc <u>maximum</u> $A(C) = 0$ ;
A est pour $R < C$ <u>négative</u>

$Q_b$ , $Q_{av}$ <u>se comportent comme des exponentielles de - n quand n $\longrightarrow \infty$</u>

Ce théorème fondamental a été démontré par Shannon pour des codes aléatoires :

> à $R < C$ fixé, il existe pour un canal donné des codes de longueur n, tels que l'erreur au décodage Q tende vers 0 comme une exponentielle de - n quand n augmente indéfiniment [ 1 ].

5° <u>Résultats expérimentaux et discussions.</u>

Nous allons, après cette évaluation de l'erreur, étudier la différence $Q - Q^*$ en fonction de R / C, pour k fixé. $Q^*$ étant la probabilité minimum de mal décoder, la différence $Q - Q^*$ mesure donc les performances du code étudié.

a) <u>Variations de Q, k étant constant.</u>

Les tableaux 2, 3, 4 contiennent les résultats obtenus pour $P = 0.6$ ($k = 20, 26, 30$) , $P = 0.5$ ($k = 20, 26, 30$) et $P = 0.4$ ($k = 26, 30$); à chaque valeur de $n - k$ nous avons construit aléatoirement une matrice de parité de code et codé 100 messages différents. Ensuite ces messages furent transmis et décodés.

<u>Remarques</u> Les courbes correspondantes sont en figures 5 , 6 , 7.

$\longrightarrow$ Les courbes obtenues sont en dents de scie ; ceci est dû au fait que 100 messages émis pour chaque valeur de k, n, p ne permet pas de donner une bonne évaluation de l'erreur Q ;

| R/C | Q-Q* | Q* | Q |
|---|---|---|---|
| 1.54 | 0.02 | 0.97 | 0.99 |
| 1.43 | 0.05 | 0.93 | 0.98 |
| 1.334 | 0.05 | 0.87 | 0.92 |
| 1.25 | 0.10 | 0.79 | 0.89 |
| 1.18 | 0.17 | 0.69 | 0.86 |
| 1.11 | 0.22 | 0.57 | 0.79 |
| 1.056 | 0.22 | 0.45 | 0.67 |
| 1. | 0.15 | 0.34 | 0.49 |
| 0.95 | 0.20 | 0.24 | 0.44 |
| 0.91 | 0.22 | 0.16 | 0.38 |
| 0.87 | 0.15 | 0.11 | 0.26 |
| 0.834 | 0.21 | 0.07 | 0.28 |
| 0.8 | 0.07 | 0.04 | 0.11 |
| 0.77 | 0.06 | 0.02 | 0.08 |
| 0.74 | 0.03 | 0.01 | 0.0.4 |

k = 20

| k | R/C | Q-Q* | Q* | Q |
|---|---|---|---|---|
| | 1.22 | 0.09 | 0.84 | 0.93 |
| | 1.18 | 0.09 | 0.76 | 0.85 |
| | 1.13 | 0.15 | 0.67 | 0.82 |
| | 1.08 | 0.19 | 0.56 | 0.75 |
| | 1.04 | 0.20 | 0.45 | 0.65 |
| | 1. | 0.17 | 0.35 | 0.52 |
| | 0.96 | 0.17 | 0.26 | 0.43 |
| | 0.93 | 0.16 | 0.19 | 0.35 |
| | 0.9 | 0.17 | 0.13 | 0.30 |
| | 0.866 | 0.07 | 0.09 | 0.16 |
| | 0.84 | 0.08 | 0.06 | 0.14 |

k = 26

| k | R/C | Q-Q* | Q* | Q |
|---|---|---|---|---|
| 8 | 1.25 | 0.06 | 0.89 | 0.95 |
| 0 | 1.2 | 0.08 | 0.82 | 0.90 |
| 2 | 1.15 | 0.15 | 0.75 | 0.90 |
| 4 | 1.11 | 0.13 | 0.65 | 0.78 |
| 6 | 1.07 | 0.14 | 0.56 | 0.70 |
| 8 | 1.036 | 0.19 | 0.46 | 0.65 |
| 0 | 1. | 0.21 | 0.36 | 0.55 |
| 2 | 0.97 | 0.20 | 0.28 | 0.48 |
| 4 | 0.94 | 0.17 | 0.20 | 0.37 |
| 6 | 0.91 | 0.12 | 0.15 | 0.27 |
| 8 | 0.88 | 0.09 | 0.10 | 0.19 |
| 0 | 0.86 | 0.10 | 0.07 | 0.17 |
| 2 | 0.832 | 0.07 | 0.04 | 0.11 |

k = 30

Tableau 3 P = 0.5



Q - Q* = estimation des performances du code

P = 0.5 probabilité d'effaçage

expérience de 100 messages

k ( information) = 20 ———

k ( information) = 26 – – – –

k ( information) = 30 — . — . —

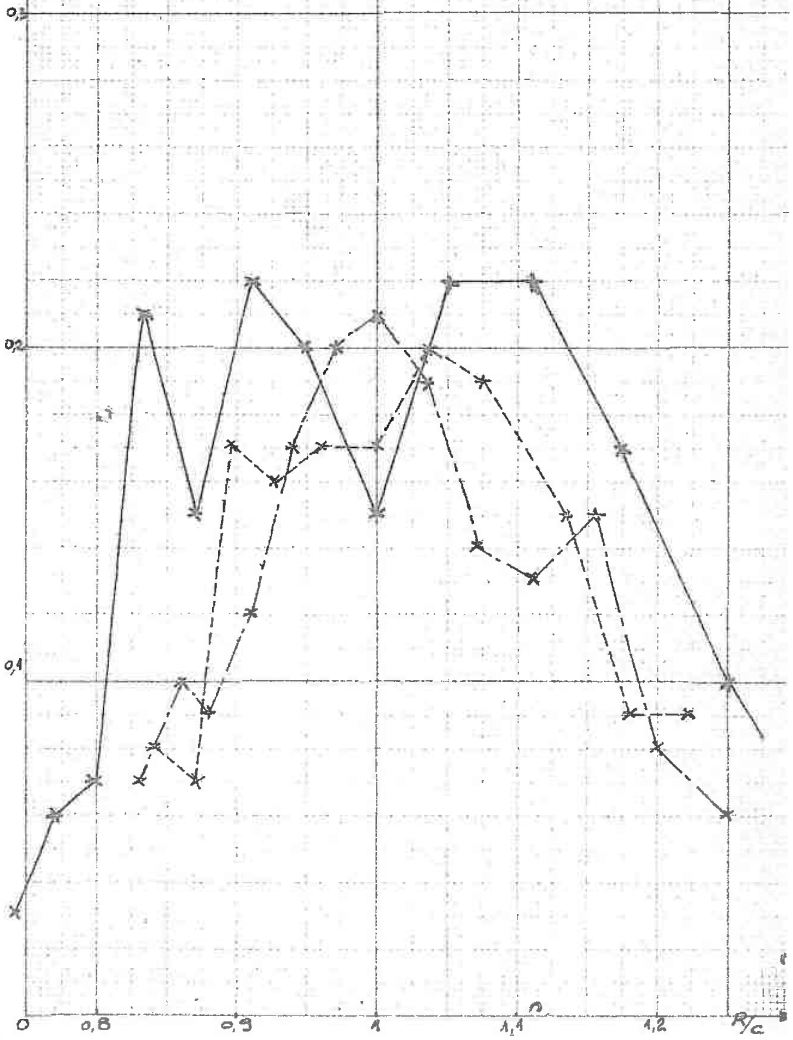



Fig. 6

| R/C | Q-Q* | Q* | Q |
|---|---|---|---|
| 1.25 | 0.07 | 0.84 | 0.97 |
| 1.20 | 0.07 | 0.78 | 0.85 |
| 1.16 | 0.05 | 0.71 | 0.76 |
| 1.12 | 0.13 | 0.64 | 0.77 |
| 1.08 | 0.14 | 0.56 | 0.70 |
| 1.05 | 0.20 | 0.49 | 0.69 |
| 1.02 | 0.24 | 0.40 | 0.64 |
| 0.99 | 0.10 | 0.33 | 0.43 |
| 0.96 | 0.11 | 0.27 | 0.38 |
| 0.93 | 0.11 | 0.21 | 0.32 |
| 0.9 | 0.13 | 0.16 | 0.29 |
| 0.88 | 0.16 | 0.12 | 0.28 |
| 0.86 | 0.05 | 0.09 | 0.14 |
| 0.835 | 0.08 | 0.07 | 0.15 |

k = 26

| k | R/C | Q-Q* | Q* | Q |
|---|---|---|---|---|
| | 1.22 | 0.09 | 0.82 | 0.91 |
| | 1.17 | 0.10 | 0.76 | 0.86 |
| | 1.14 | 0.03 | 0.70 | 0.73 |
| | 1.1 | 0.11 | 0.63 | 0.84 |
| | 1.07 | 0.18 | 0.55 | 0.73 |
| | 1.04 | 0.21 | 0.49 | 0.70 |
| | 1.015 | 0.17 | 0.41 | 0.58 |
| | 0.99 | 0.14 | 0.34 | 0.48 |
| | 0.96 | 0.17 | 0.28 | 0.45 |
| | 0.94 | 0.20 | 0.22 | 0.42 |
| | 0.92 | 0.21 | 0.18 | 0.39 |
| | 0.89 | 0.04 | 0.14 | 0.18 |
| | 0.87 | 0.10 | 0.11 | 0.21 |
| | 0.85 | 0.05 | 0.08 | 0.13 |

k = 30

| - k | R/C | Q-Q* | Q* | Q |
|---|---|---|---|---|
| 8 | 1.79 | 0.01 | 0.99 | 1.00 |
| 10 | 1.66 | 0.00 | 0.98 | 0.98 |
| 12 | 1.56 | 0.03 | 0.96 | 0.99 |
| 14 | 1.48 | 0.06 | 0.93 | 0.99 |
| 16 | 1.34 | 0.08 | 0.90 | 0.98 |
| 18 | 1.32 | 0.08 | 0.84 | 0.92 |
| 20 | 1.25 | 0.10 | 0.78 | 0.88 |
| 22 | 1.19 | 0.11 | 0.70 | 0.81 |
| 24 | 1.14 | 0.19 | 0.61 | 0.80 |
| 26 | 1.08 | 0.18 | 0.52 | 0.80 |
| 28 | 1.04 | 0.20 | 0.43 | 0.63 |
| 30 | 1. | 0.20 | 0.35 | 0.55 |
| 32 | 0.96 | 0.20 | 0.28 | 0.48 |
| 34 | 0.925 | 0.15 | 0.21 | 0.36 |
| 36 | 0.89 | 0.15 | 0.16 | 0.31 |
| 38 | 0.86 | 0.12 | 0.12 | 0.24 |
| 40 | 0.835 | 0.12 | 0.08 | 0.20 |
| 42 | 0.805 | 0.15 | 0.06 | 0.11 |
| 44 | 0.78 | 0.05 | 0.04 | 0.09 |

k = 20

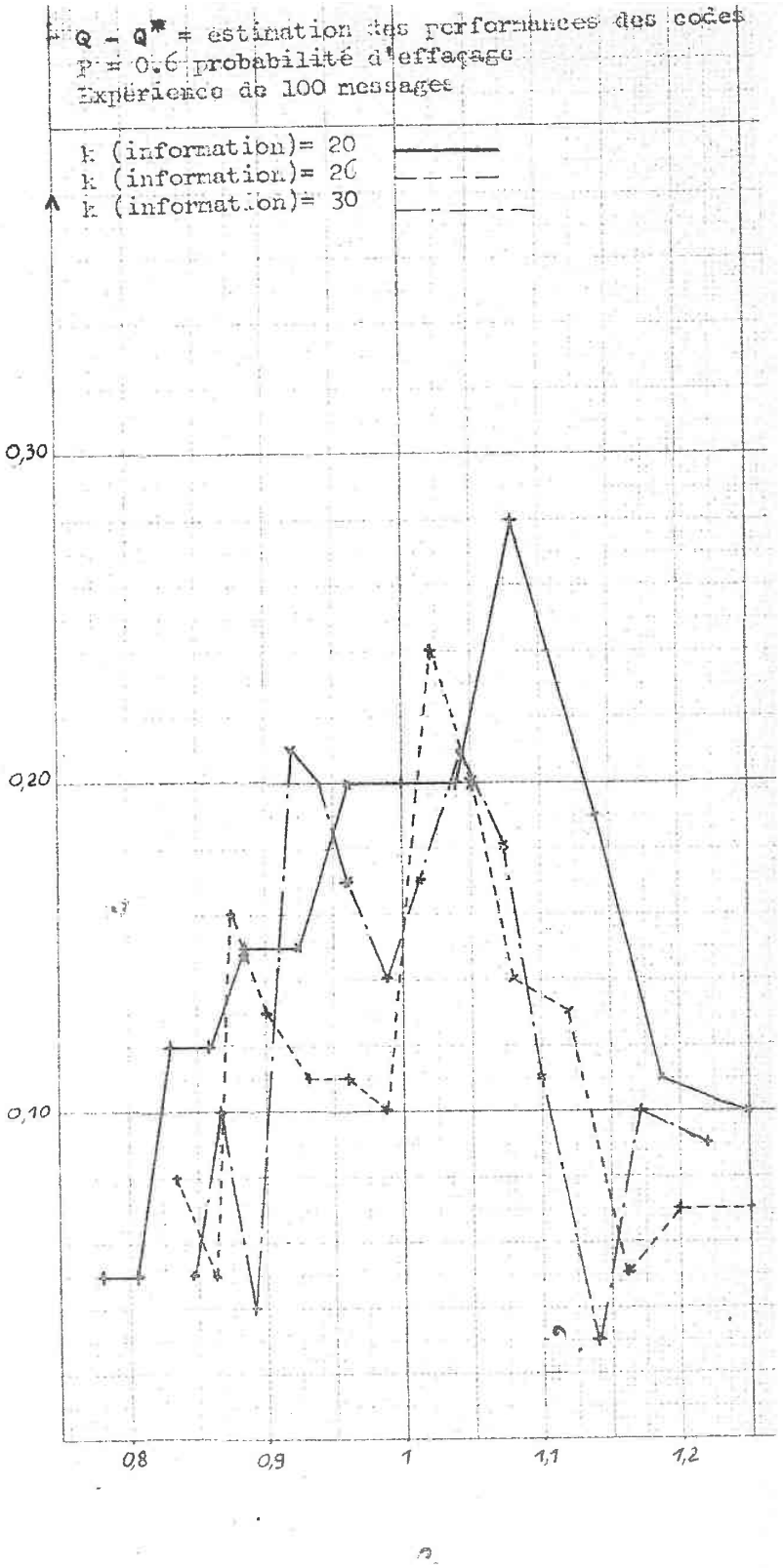



Fig. 5

Tableau 2 - P = 0.6

| R/C | Q - Q* | Q* | Q |
|---|---|---|---|
| 1.45 | 0.02 | 0.98 | 1.00 |
| 1.35 | 0.05 | 0.95 | 1.00 |
| 1.28 | 0.09 | 0.90 | 0.99 |
| 1.2 | 0.11 | 0.82 | 0.93 |
| 1.14 | 0.15 | 0.70 | 0.85 |
| 1.08 | 0.19 | 0.56 | 0.75 |
| 1.03 | 0.16 | 0.42 | 0.58 |
| 0.985 | 0.19 | 0.30 | 0.49 |
| 0.94 | 0.18 | 0.19 | 0.37 |
| 0.9 | 0.20 | 0.12 | 0.32 |
| 0.87 | 0.13 | 0.07 | 0.20 |
| 0.835 | 0.07 | 0.04 | 0.11 |
| 0.8 | 0.11 | 0.02 | 0.13 |
| 0.775 | 0.05 | 0.01 | 0.06 |

k = 26

| | R/C | Q-Q* | Q* | Q |
|---|---|---|---|---|
| | 1.39 | 0.00 | 0.98 | 0.98 |
| | 1.32 | 0.01 | 0.96 | 0.97 |
| | 1.25 | 0.07 | 0.91 | 0.98 |
| 2 | 1.19 | 0.07 | 0.83 | 0.90 |
| 4 | 1.13 | 0.12 | 0.73 | 0.87 |
| 6 | 1.08 | 0.12 | 0.60 | 0.73 |
| 8 | 1.04 | 0.20 | 0.47 | 0.67 |
| 10 | 1. | 0.26 | 0.35 | 0.51 |
| 12 | 0.96 | 0.22 | 0.24 | 0.46 |
| 24 | 0.925 | 0.17 | 0.16 | 0.33 |
| 26 | 0.89 | 0.06 | 0.10 | 0.16 |
| 28 | 0.86 | 0.06 | 0.06 | 0.12 |
| 30 | 0.835 | 0.16 | 0.03 | 0.19 |
| 32 | 0.81 | 0.02 | 0.02 | 0.04 |
| 34 | 0.785 | 0.02 | 0.01 | 0.03 |

k = 30

Tableau 4 P = 0.4

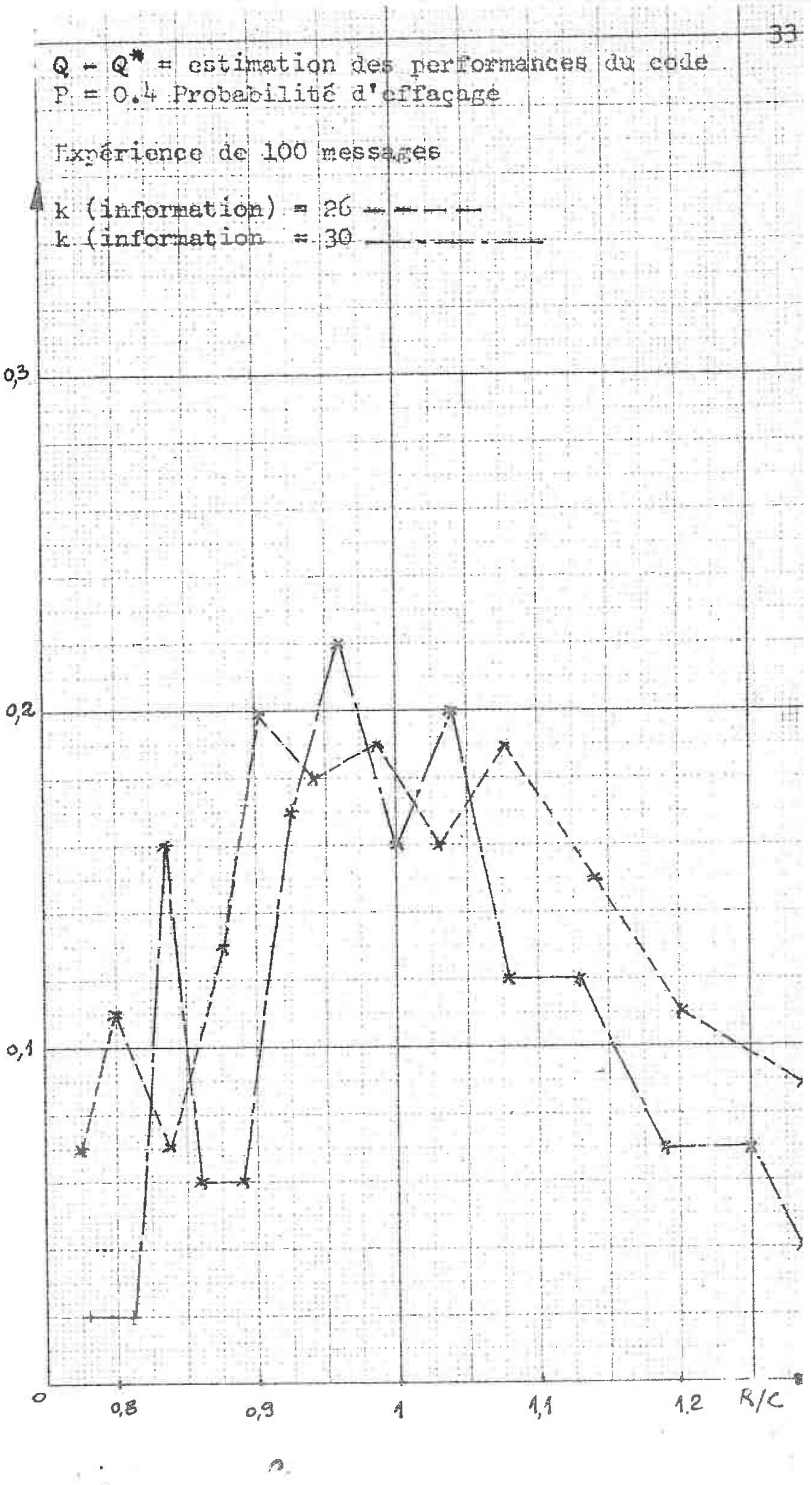



Fig. 7

cela devient très net dès que R devient égal à C ; nous avons alors Q qui devient inférieur à 0.5 et décroît très rapidement ; en même temps l'importance d'un message mal reçu, augmente quand on veut évaluer Q. il nous faudra donc prendre un nombre de messages plus grand.

Pour une valeur de $Q - Q^{*}$ , on peut assimiler une "largeur" à ces courbes et la comparer ; il semble que cette largeur pour $Q - Q^{*}$ faible, décroît lorsque k augmente (donc quand n augmente) .

⇒ Pour toutes ces courbes au voisinage de $R/C \leq 1$, alors que Q décroît rapidement $Q - Q^{*}$ reste important et il paraît judicieux de chercher à réduire ces causes d'erreurs.

b) Ces premières courbes ne nous permettant pas de conclure, nous avons cherché à avoir des résultats plus probants ; nous avons alors fait deux séries d'expériences où nous emettions 1000 messages pour chaque valeur de k , n , P fixés ; ces expériences ont été effectuées pour P = 0.5 et k = 20 , 50 . Comme précédemment nous avons tracé $Q - Q^{*}$ en fonction de R/C ; les résultats sont dans le tableau 5 et les courbes sur la figure 8.

Cette fois la "largeur"des courbes, pour une même valeur de $Q - Q^{*}$ , décroît quand n augmente.

Nous avons aussi tracé sur la figure 8 la probabilité Q (pour $R/C < 1$) ; ceci nous permet de voir que dès que R/C est inférieur à une valeur $(R/C)_0$ (valeur qui dépend de k) Q est inférieur à $10^{-3}$ ; ce seuil $(R/C)_0$ augmente et tend vers 1 quand $n \longrightarrow \infty$ $(k \longrightarrow \infty)$ ; ceci ne contredit pas le théorème de Shannon ; en conclusion nous pouvons utiliser, sans craindre de trop mauvaises preformances, un code aléatoire quand $(R/C) < (R/C)_0$.

. 5 1000 messages par expérience
* = estimation des performances des codes
rcentage d'erreur

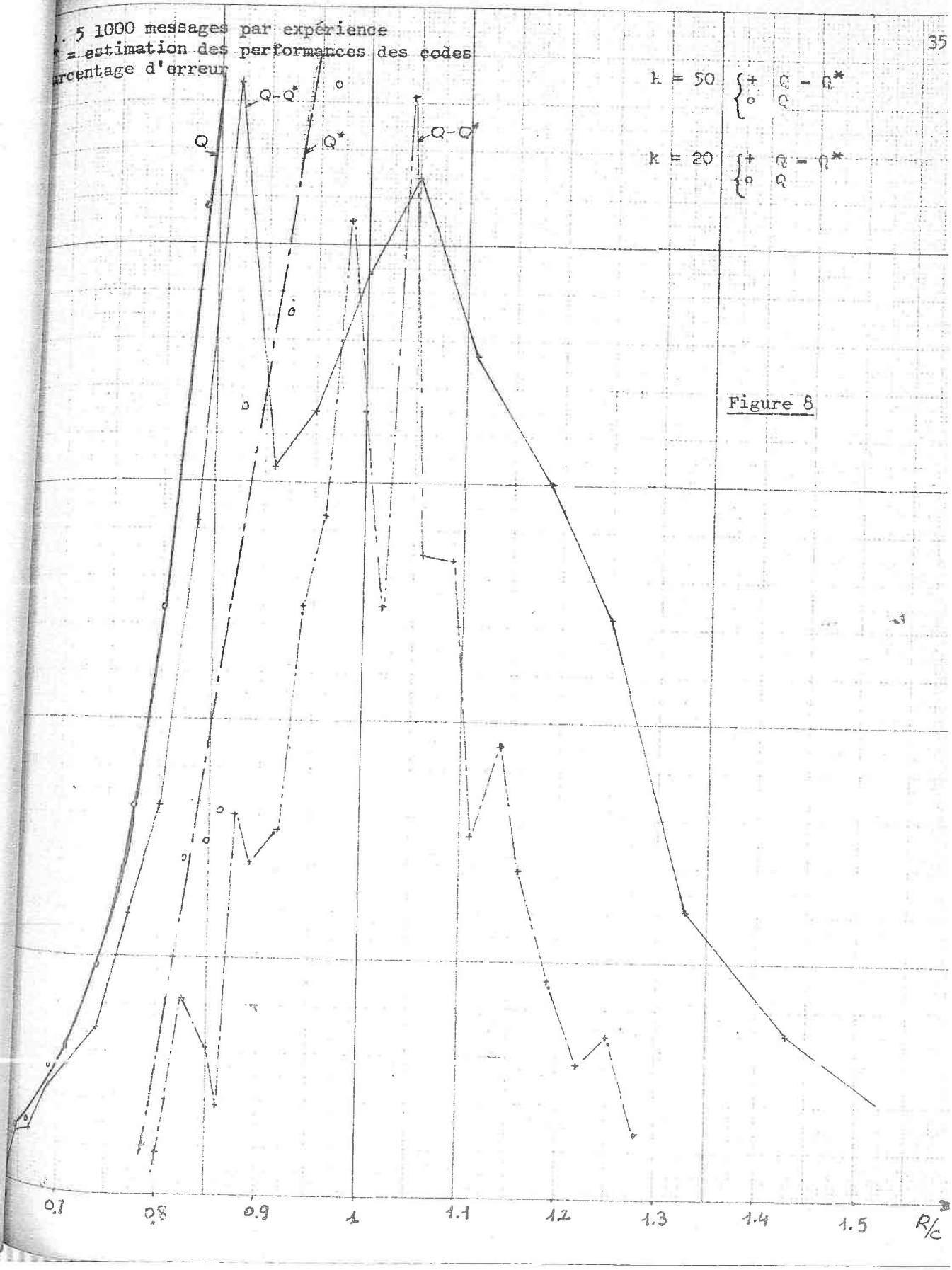



Figure 8

R = 50

| n - k | R/C | Q - Q* | Q* | Q |
|---|---|---|---|---|
| 28 | 1.28 | 0.014 | 0.978 | 0.992 |
| 30 | 1.25 | 0.034 | 0.964 | 0.998 |
| 32 | 1.22 | 0.028 | 0.942 | 0.970 |
| 34 | 1.19 | 0.046 | 0.912 | 0.958 |
| 36 | 1.16 | 0.069 | 0.872 | 0.941 |
| 38 | 1.14 | 0.093 | 0.822 | 0.915 |
| 40 | 1.11 | 0.076 | 0.762 | 0.838 |
| 42 | 1.088 | 0.134 | 0.694 | 0.828 |
| 44 | 1.06 | 0.135 | 0.620 | 0.755 |
| 46 | 1.042 | 0.229 | 0.542 | 0.671 |
| 48 | 1.02 | 0.124 | 0.464 | 0.588 |
| 50 | 1.00 | 0.165 | 0.388 | 0.553 |
| 52 | 0.98 | 0.203 | 0.318 | 0.421 |
| 54 | 0.96 | 0.143 | 0.256 | 0.399 |
| 56 | 0.94 | 0.124 | 0.200 | 0.324 |
| 58 | 0.92 | 0.077 | 0.153 | 0.230 |
| 60 | 0.908 | 0.070 | 0.115 | 0.185 |
| 62 | 0.89 | 0.080 | 0.084 | 0.164 |
| 64 | 0.876 | 0.019 | 0.060 | 0.081 |
| 66 | 0.86 | 0.031 | 0.043 | 0.074 |
| 68 | 0.85 | 0.041 | 0.030 | 0.071 |
| 70 | 0.832 | 0.020 | 0.020 | 0.001 |
| 72 | 0.82 | - | 0.008 | - |

R = 20

| n - k | R/C | Q - Q* | Q* | Q |
|---|---|---|---|---|
| 6 | 1.54 | 0.018 | 0.966 | 0.984 |
| 8 | 1.43 | 0.035 | 0.932 | 0.969 |
| 10 | 1.334 | 0.061 | 0.875 | 0.936 |
| 12 | 1.25 | 0.122 | 0.792 | 0.914 |
| 14 | 1.18 | 0.150 | 0.688 | 0.838 |
| 16 | 1.12 | 0.176 | 0.570 | 0.746 |
| 18 | 1.056 | 0.213 | 0.449 | 0.662 |
| 20 | 1. | 0.194 | 0.337 | 0.531 |
| 22 | 0.952 | 0.165 | 0.241 | 0.406 |
| 24 | 0.910 | 0.154 | 0.164 | 0.318 |
| 26 | 0.87 | 0.234 | 0.107 | 0.341 |
| 28 | 0.824 | 0.141 | 0.067 | 0.208 |
| 30 | 0.8 | 0.082 | 0.041 | 0.123 |
| 32 | 0.77 | 0.058 | 0.024 | 0.082 |
| 34 | 0.74 | 0.035 | 0.013 | 0.048 |
| 36 | 0.712 | 0.027 | 0.007 | 0.030 |
| 38 | 0.69 | 0.022 | 0.004 | 0.026 |
| 40 | 0.666 | 0.012 | 0.002 | 0.014 |
| 42 | 0.646 | 0.011 | 0.001 | 0.012 |
| 44 | 0.624 | 0.000 | 0.000 | 0.000 |

Tableau 5 ( P = 0.5 1000 messages émis)

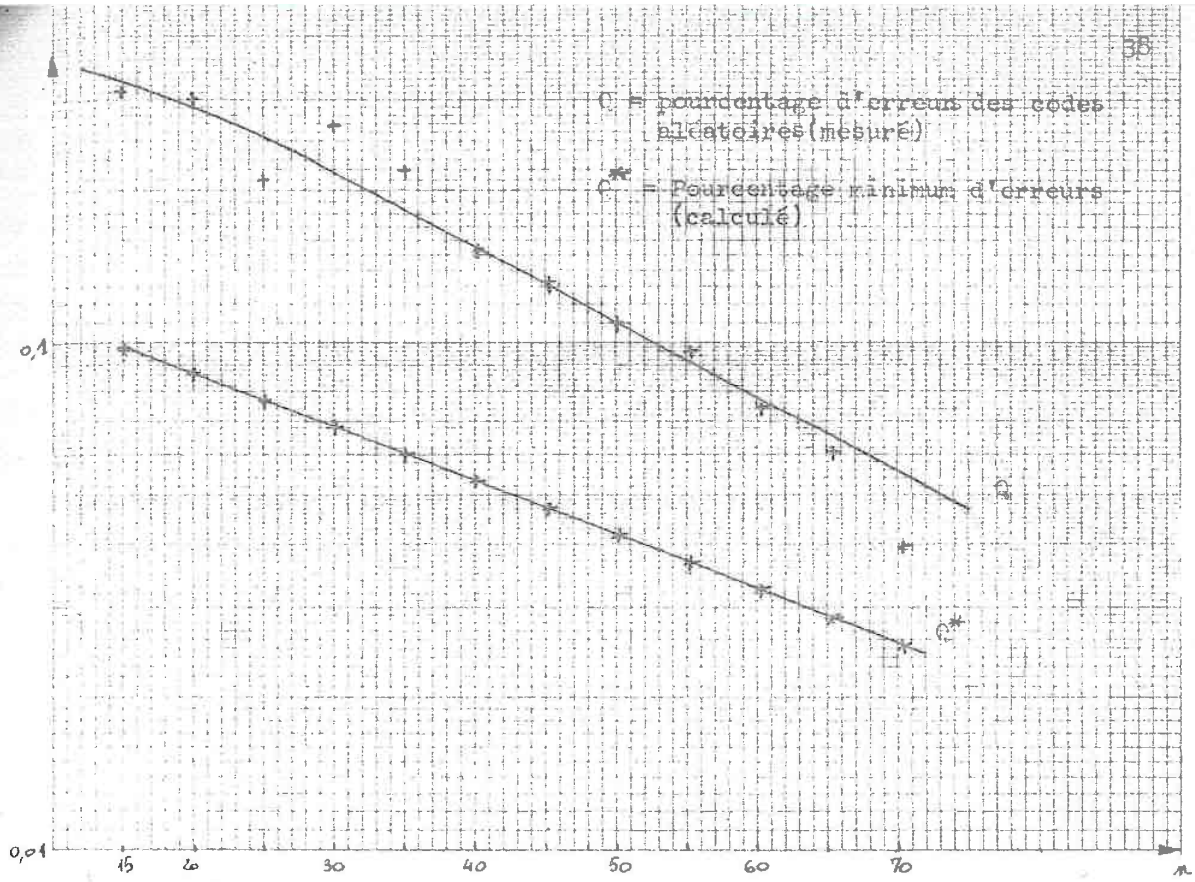
⟹ Nous voyons que ces courbes restent en dents de scie ; cela a deux origines :

. 1000 messages ne permettent pas une bonne évaluation de Q , quand Q tend vers 0.

. à chaque valeur de n et k nous construisons aléatoirement une matrice de code [A] ; nous connaissons les propriétés moyennes des codes aléatoires et savons qu'ils sont en moyenne bons ; mais une fois un code choisi nous n'avons aucune garantie sur ses performances et nous ne savons pas de combien ses propriétés peuvent s'écarter des propriétés moyennes ; par contre nous pouvons dire que la plupart sont bons : en effet soit un ensemble de M quantités positives dont la moyenne est $\varepsilon$ il y en aura au plus $\frac{M}{n}$ qui peuvent être supérieures à $n\varepsilon$ . Ceci nous permet d'affirmer que la plupart des codes aléatoires ont des performances qui ne s'éloignent pas trop des performances moyennes donc sont bons.

Les résultats obtenus précédemment ne sont pas des résultats "moyens" pour les codes aléatoires ; en effet à chaque valeur de n , k , P on étudie les performances d'un <u>seul</u> code choisi au hasard parmi les codes possibles. En changeant de code à chaque message envoyé, nous aurions pu avoir une meilleure évaluation des propriétés moyennes des codes.

Malheureusement tous ces essais prennent énormément de temps machine et nous n'avons pu les recommencer dans de telles conditions, ni en envisager pour $k > 50$. Déjà pour cette valeur, nous avions utilisé une I. B. M. 360-60 et certains points ont demandé plus d'une heure decalcul. Tous les autres essais ont été faits sur I. B. M. 1130 où le temps de calcul a varié d'une demi-heure ($k = 30$ ; n - k faible) jusqu'à plus de 3 heures pour n grand ($n = 70$ ; $k = 30$).




$R = \frac{k}{n} = 0.4$ C = 0.5 Probabilité d'effaçage = 0.5

Fig 9

| k | n | Q | Q* calculé |
|---|---|---|---|
| 6 | 15 | 0.303 | 0.093 |
| 8 | 20 | 0.289 | 0.083 |
| 10 | 25 | 0.199 | 0.074 |
| 12 | 30 | 0.257 | 0.066 |
| 14 | 35 | 0.210 | 0.058 |
| 16 | 40 | 0.148 | 0.052 |
| 18 | 45 | 0.128 | 0.046 |
| 20 | 50 | 0.104 | 0.041 |
| 22 | 55 | 0.094 | 0.036 |
| 24 | 60 | 0.073 | 0.032 |
| 26 | 65 | 0.060 | 0.028 |
| 28 | 70 | 0.039 | 0.025 |

Tableau 6 - (C = 0.5 R = 0.4 P = 0.5)

c) Variations à R = k/n constant < C

Nous avons pris P = 0.5 ; R = 0.4 < C = 0 5 et avons fait varier k de 6 à 28 digits. Les résultats se trouvent dans le tableau 6. Nous avons tracé sur la Figure 9, Q en fonction de n ; les coordonnées sont semi-logarithmiques. A chaque message émis nous avons construit un nouveau code aléatoire.

Remarques

A partir de n = 40 les points pour Q et $Q^*$ semblent se répartir autour d'une droite. Ceci vérifie la loi de décroissance en exponentielle - n de Q. $Q^*$ a été calculé à chaque valeur de n , k , P.

d) Conclusions

Ces divers résultats nous montrent qu'il serait souhaitable de se placer dans l'intervalle $(R/C)_o < R/C < 1$ , avec des valeurs de n grandes et de chercher si l'utilisation de codes performantes peut réduire Q et de quelle quantité.

Nous allons donc essayer de construire des codes performants, ensuite nous comparerons leurs résultats à ceux des codes aléatoires.

## V Construction systématique de Codes corrigeant jusqu'à T effacements.

Nous avons vu précédemment qu'un code corrigera jusqu'à t effacements s'il n'existe pas entre les lignes et les colonnes de $|\overline{A}|$, sa matrice de parité, de combinaison linéaire nulle impliquant de 1 à t vecteurs.

Nous construirons donc la matrice $|\overline{A}|$ de la manière suivante : si la longueur des lignes est plus grande que celle des colonnes, nous prendrons les n - k colonnes de telle sorte qu'il n'y ait pas de combinaisons linéaires nulles impliquant au plus t d'entre elles ; dans le cas contraire nous ferons cette opération sur les lignes.

1° Méthode

Nous cherchons k vecteurs de longueur m = n - k tels que la matrice formée par ces vecteurs, comme vecteurs colonnes, soit celle d'un code corrigeant jusqu'à t effacements : Pour cela il faut que toute sous matrice carréd de dimension t de $|\overline{A}|$ soit régulière ; nous cherchons à exprimer cette condition sous une autre forme :

⟶ supposons qu'il y ait 1 effacement parmi les k digits d'information et t - 1 parmi les m de parité. On a donc un système 1 inconnue à m - t + 1 équations à résoudre ; il faut donc que la colonne de $|\overline{A}|$, correspondant au digit effacé, ait au moins un 1 dans une position qui ne soit pas commune à celles des t - 1 digits de parité effacés ; il faudra donc que cette colonne ait au moins t 1 .

définition

Soit un vecteur $|\overline{V}|$ formé de n digits, on appelle $|V|$ son poids le nombre de digits égaux à 1 existant dans $|\overline{V}|$.

$$|V| = \sum_{j=1}^{n} v_j \qquad (v_j \text{ est un digit de } |\overline{V}|)$$

Les colonnes de $|\overline{A}|$ devront donc avoir toutes un poids supérieur ou égal à t

→ nous supposons maintenant 2 effacements dans les digits d'information et $t-2$ dans la parité ; les digits d'information effacés sont le $i_1^{ème}$ et le $i_2^{ème}$. Nous devrons donc résoudre un système de 2 inconnues à $m - t + 2$ équations. En le résolvant nous formons la combinaison linéaire (ou somme modulo 2) des $i_1^{ème}$ et $i_2^{ème}$ colonnes de A ; il faudra que le vecteur obtenu ait au moins un digit égal à 1 dans une position différente de celle d'un digit de parité effacé ; donc le poids de ce vecteur devra être supérieur ou égal à $t - 1$ .

→ Si nous considérons maintenant 3, 4, ... et t effacements dans l'information, alors qu'il y en a t répartis dans l'information et la parité, nous arrivons aux conditions suivantes que devront remplir les colonnes de $|\overline{A}|$ .

On appelle $a_i$ la $i^{ème}$ colonne de $|\overline{A}|$ et $a_i$ son poids.

$a_i \oplus a_j$ désignera le vecteur somme modulo 2 des vecteurs $a_i$ et $a_j$

$$(5)\quad \begin{cases} |a_j| \geq t & j = 1, \ldots, k \\ |a_{j_1} \oplus a_{j_2}| \geq t-1 & 1 \leq j_1 < j_2 \leq k \\ \ldots \\ |a_{j_1} \oplus a_{j_2} \oplus \ldots \oplus a_{j_t}| \geq 1 & 1 \leq j_1 < j_2 \ldots < j_t \leq k \end{cases}$$

## 2° Construction de tels vecteurs. Fonction " LINTE"

**Définition** : nous dirons que r vecteurs sont t linéairement indépendants, s'il n'existe pas entre eux de combinaisons linéaires nulles impliquant au plus t vecteurs.

Les k vecteurs de longueur m qui forment $|\boxed{A}|$ devront donc être t linéairement indépendants.

### a) Test de "t linéarité" de r vecteurs

Soient r - 1 vecteurs de longueur m = n - k t linéairement indépendants. Ils forment les r - 1 premières colonnes de la matrice $|\boxed{A}|$. Soit un vecteur $|\boxed{Y}|$ de longueur m ; nous allons tester si il est t linéairement indépendant avec les précédents. Si oui il formera la 2ème colonne de $|\boxed{A}|$ et on testera un autre vecteur jusqu'à ce que $|\boxed{A}|$ soit formée de k vecteurs t linéairement indépendants.

Soit $t_1 = \min(t,r)$ ; les conditions (5) s'écriront en réalité pour les r - 1 colonnes de $|\boxed{A}|$, et le vecteur $|\boxed{Y}|$

$$(5)\quad \begin{cases} |Y| \geqslant t \\ |a_j \oplus |\boxed{Y}| | \geqslant t-1 & j = 1, \ldots, t_1 - 1 \\ \cdots \\ |a_{j_1} \oplus a_{j_2} \oplus \ldots \oplus a_{j_{t_1-1}} \oplus |\boxed{Y}| | \geqslant t - t_1 + 1 & 1 \leqslant j_1 < j_2 \cdots < j_{t_1-1} \leqslant t_1 - 1 \end{cases}$$

Nous devrons donc tester toute somme (modulo 2) de j colonnes de $|\boxed{A}|$ ; j variant de 1 à $t_{1-1}$ : soit pour chaque valeur de j tester les $C^j_{r-1}$ sommes possibles

(si $r \geqslant t$ sinon ce sera les $C^j_{t_1-1}$ sommes possibles).

α test de linéarité

Soit Pv le poids de tout vecteur somme de j colonnes de $|\overline{A}|$ et de $|\overline{Y}|$ . Si $|\overline{Y}|$ est t linéairement indépendant avec les r - 1 premiers vecteurs de $|\overline{A}|$ alors $\boxed{Pv \geqslant t - j}$ ; si Pv ne remplit pas cette condition alors $|\overline{Y}|$ ne convient pas.

β Méthode pour obtenir toutes les combinaisons linéaires de j vecteurs parmi r - 1

→ Pour avoir toutes les sommes de j vecteurs colonnes de $|\overline{A}|$ pris parmi r - 1 nous agirons de la manière suivante : nous appelerons $j_i$ l'indice dans la matrice $|\overline{A}|$ de la $i^{ème}$ colonne intervenant dans la somme de j colonnes : Soit $|\overline{V}|$ cette somme.

$$|\overline{V}| = a_{j_1} \oplus \dots \oplus a_{j_i}$$

Ces indices varieront de la mnière suivante :

$$\begin{cases} j_1 & \text{varie de } 1 \text{ à } r - j \\ j_2 & \text{varie de } j_1 + 1 \text{ à } r - j + 1 \\ \dots & \dots \\ j_j & \text{varie de } j_{j-1} + 1 \text{ à } r - 1 \end{cases}$$

Montrons que nous avons ainsi testé toutes les sommes de j vecteurs parmi r - 1 : soit

$$C^j_{r-1} = \frac{(r-1)!}{j!\,(r-j-1)!}$$

Par la méthode exposée précédemment nous testons :

$$B = \sum_{j_1=1}^{r-j} \; \sum_{j_2-j_1+1}^{r-j+1} \dots \sum_{j_j=j_{j-1}+1}^{r-1} (1) \quad \text{combinaisons différentes}$$

il nous faut montrer que $B = C^{j}_{r-1}$

- Si $j = 1$ $\quad B = \sum_{l=1}^{r-1} (1) = r - 1 = C^{1}_{r-1}$

Si $j = 2$ $\quad B = \sum_{l=1}^{r-1} \sum_{i=l+1}^{r-2} (1) = \sum_{l=1}^{r-1} (r - 1 - i)$

$$B = (r - 1)(r - 2) - \sum_{l=1}^{r-1} i = \frac{(r-1)(r-2)}{2} = C^{2}_{r-1}$$

- Supposons que :

$$\sum_{j_1=1}^{r-j+1} \sum_{j_2=j_1+1}^{r-j+2} \cdots\cdots \sum_{j_{j-1}=j_{j-2}+1}^{r-1} (1) = C^{j-1}_{r-1}$$

Calculons

$$B = \sum_{j_1=1}^{r-j} \sum_{j_2=j_1+1}^{r-j+1} \cdots\cdots \sum_{j_j=j_{j-1}}^{r-1} (1) = \sum_{l=1}^{r-j} B_l$$

où $B_l = \sum_{j_2=l+1}^{r-j+1} \sum_{j_3=j_2+1}^{r-j+2} \cdots\cdots \sum_{j_j-j_{j-1}+1}^{r-1} (1)$

$$B_l = \sum_{j_2=1}^{r-j+1-l} \sum_{j_3=j_2+l+1}^{r-j+2} \cdots\cdots \sum_{j_j=j_{j-1}+1}^{r-1} (1)$$

$$B_l = \sum_{j_2=1}^{r-l-j+1} \sum_{j_3=j_2+1}^{r-l-j+2} \cdots\cdots \sum_{j_j=j_{j-1}+1}^{r-l-1} (1) = C^{j-1}_{r-l-1}$$

Donc $B = \sum_{l=1}^{r-1} B_l = C^{j-1}_{r-2} + C^{j-1}_{r-3} + \ldots + C^{j-1}_{j} + C^{j-1}_{j-1}$

utilisons la formule :

$$C^{P}_{n} = C^{P}_{n-1} + C^{P-1}_{n-1}$$

$$C^{j-1}_{j-1} = 1 = C^{j}_{j}$$

et $B = C^{j-1}_{r-2} + \ldots + \underbrace{C^{j-1}_{j} + C^{j}_{j}}_{C^{j}_{j+1}} = C^{j-1}_{r-2} + \ldots + \underbrace{C^{j-1}_{j+1} + C^{j}_{j+1}}_{C^{j}_{j+2}}$

soit en regroupant les termes 2 par 2

$$B = C^{j-1}_{r-2} + C^{j}_{r-2} = C^{j}_{r-1}$$

<u>Nous avons donc montré que par notre méthode nous testions toutes les $C^{j}_{r-1}$ combinaisons linéaires de colonnes de $|\underline{A}|$ possibles.</u>

⟶ si $j_j = r - 1$ il faudra savoir si $j_{j-1} = r - 2 \ldots$

si tous indices $j_i (i = 1, \ldots, j)$ sont égaux à $r - j + i - 1$ alors nous avons testé toutes les sommes de $j$ vecteurs formés sur les $r - 1$ premières colonnes de $|\underline{A}|$.

Nous associerons à chaque vecteur colonne $a_{j_i}$ un entier $i\ k_i$ égal à 0 si $j_i = r - j + i - 1$ et égal à 1 dans le cas contraire.

⟶ Supposons que la $l^{ème}$ colonne de la somme $|\overline{V}|$ soit telle que $j_l < r - j + l - 1$ $(l < j)$ mais que pour $i = l + 1, \ldots, j$ on ait $j_i = r - j + i - 1$ .

nous sommes dans le cas suivant et nous supposons que $|Y|$ est t linéairement indépendant avec les j vecteurs considérés ; nous avons:

$$\begin{cases} 1 & {}_{i}k_{1} = 1 \\ 1+1 & {}_{i}k_{1+1} = 0 \\ \\ j & {}_{i}k_{j} = 0 \end{cases}$$

$|Y|$ étant t linéairement indépendant avec les j vecteurs formant $|V|$ nous avons à considérer une autre somme de j vecteurs colonnes de $|A|$ : les $1-1$ premiers vecteurs seront les mêmes que précédemment ; nous prendrons pour $1^{ème}$ vecteur celui dont l'indice sera $j_1 + 1$ ; donc le nouvel indice du $1^{ème}$ vecteur sera augmenté d'une unité $(j_1 = j_1 + 1)$ ; pour les vecteurs suivants on aura :

$(i = 1+1, \ldots, j)$ $\quad j_i = j_1 + i - 1$ ; il faudra savoir si

$j_1 = r - j + 1 - 1$ $\qquad$ <u>si oui</u> $\qquad j_1 = 0$

$\qquad\qquad\qquad\qquad$ <u>si non</u> $\qquad j_i = 1 \quad (i = 1+1, \ldots, j)$

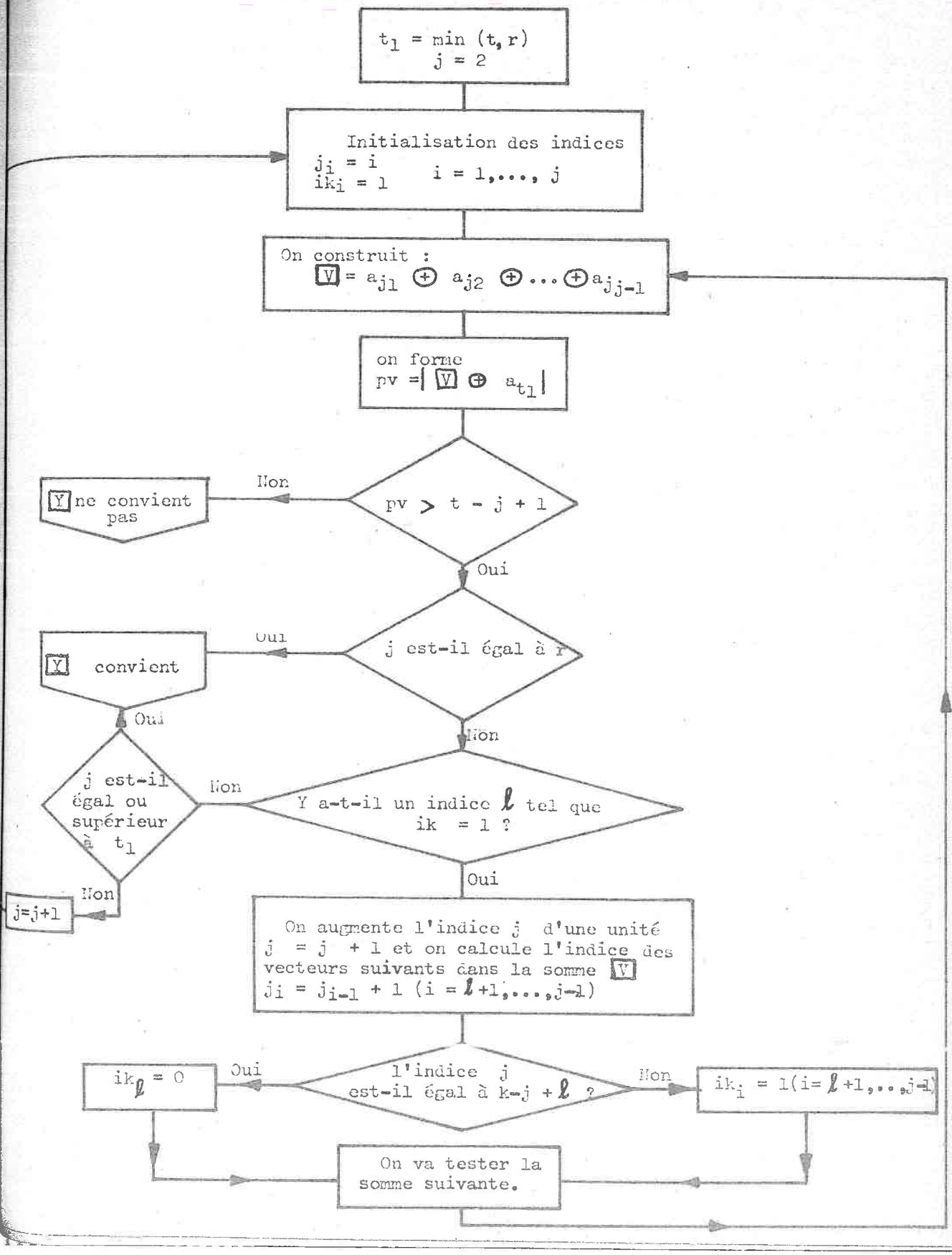
$\longrightarrow$ **ainsi** lorsque ${}_{i}k_{i} = 0$ $(i = 1, \ldots, j)$ nous aurons testé toutes les ommes de j vecteurs formées sur les $r-1$ premiers vecteurs colonnes de $|A|$.

b) <u>Fonction "LINTE"</u>

La méthode exposée précédemment est le corps de la fonction "LINTE" ; LINTE = 1 si le vecteur $|Y|$ convient, 0 dans le cas contraire.

Le sigle LINTE veut regrouper les trois mots : linéaire, T , effacement. L'organigramme est sur la page suivante.

ORGANIGRAMME



c) Construction de [A]

nous agirons de deux façons différentes :

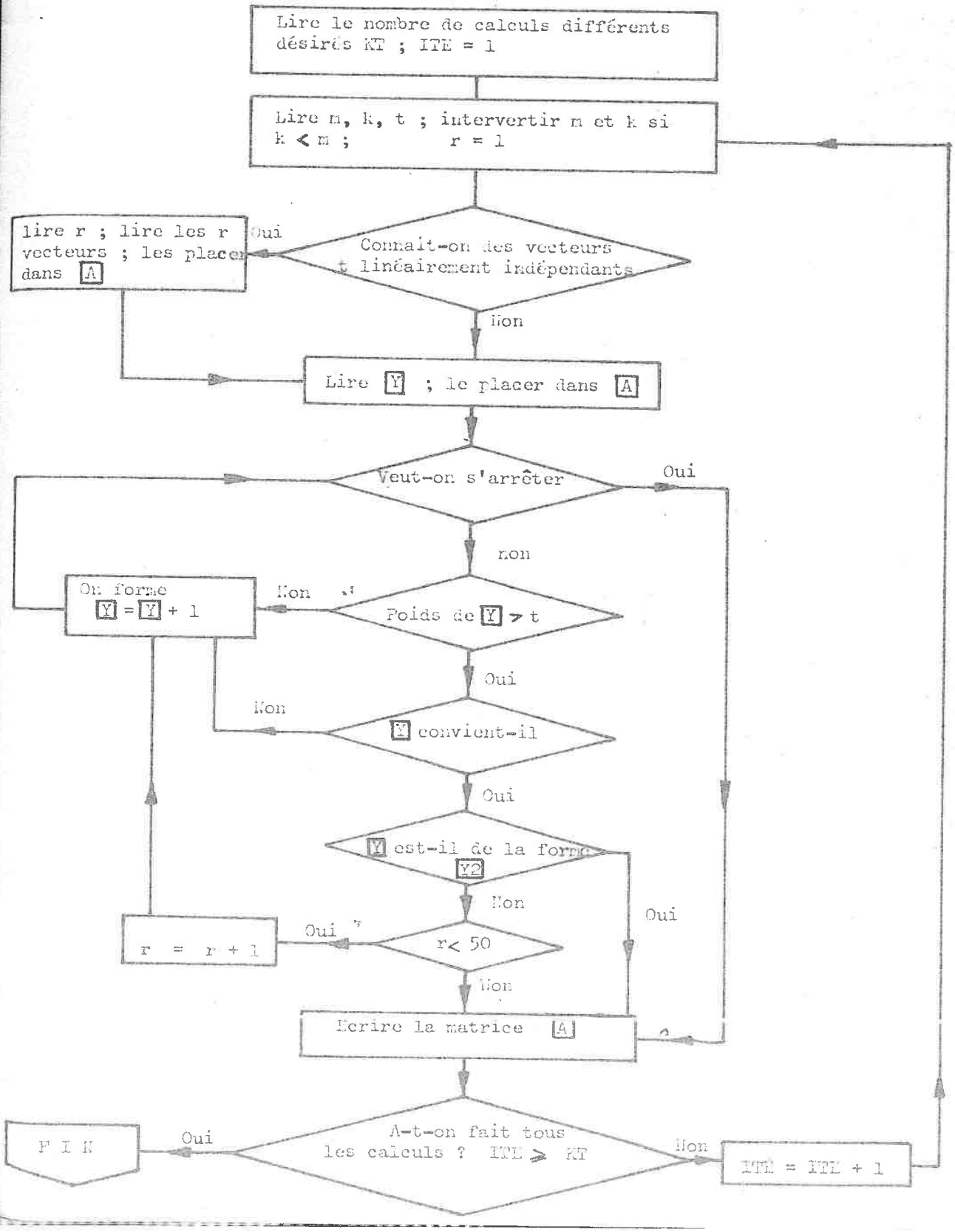
α/ nous ne connaissons pas de vecteurs t linéairement indépendants de longueur $m = n - k$ ; alors nous partons de $[Y1] = [\underbrace{1 \ldots\ldots 1}_{t} 0 \ldots 0]$ que nous affectons à la première colonne de [A] ; nous testerons tous les vecteurs de longueur m ; nous passerons d'un vecteur au suivant en consisidérant [Y] comme un nombre binaire et en lui ajoutant 1 modulo 2 . On s'arrête dès que l'on a obtenu 50 vecteurs (ce nombre est lié aux dimensions de [A] qui sont elles-mêmes limitées par le nombre de mémoires de l'Unité Centrale de l'ordinateur utilisé) ou dès que le vecteur [Y] t linéairement indépendant avec les précédents est de la forme :

$$[Y_2] = [\ldots 1 \underbrace{\ldots 1 \ldots\ldots 1}_{s}] \qquad s \geqslant m-t+2$$

en effet tout vecteur avant [Y1] aura un poids $< t$ et toute somme [Y2] $\oplus$ [Y] ([Y] étant après [Y2]) aura un poids $< t-1$ .

β/ ou nous connaissons r vecteurs ($r < k$ cherché) de longueur m , t linéairement indépendants ; nous affectons aux r premières colonnes de [A] ces valeurs et testons les vecteurs à partir de [Y1] jusqu'à [Y2].

ORGANIGRAMME



# VI

# Comparaison de divers Codes

## I Comparaison entre Codes par Blocs.

Nous avons pris un canal ayant pour probabilité d'effaçage $P = 0.25$ (soit $C = 0.75$) et avons étudié ce qui se passait pour trois rythmes de transmission :

$$\begin{cases} R \leq 0.75 \\ R \simeq 0.6 \\ R = 0.5 \end{cases}$$

Dans les trois cas nous avons fait varier k , longueur de l'information de 4 à 30 (sauf pour $R = 0.5$ où nous nous sommes arrêtés à $k = 24$).

Nous avons fait ces expériences avec des codes aléatoires par bloc, ce qui nous a donné une estimation de l'erreur $Q_a$

Nous avons fait ces expériences avec deux types de codes construits systématiquement :

1/ nous avons essayé de construire des codes performants en utilisant la procédure précédente : n , k et P sont fixés ; il nous fallait calculer t , le nombre d'effacements parmi l'information que devait corriger le code, nous avons donc pris $t = \text{Entier}\left[ (1 - R)\, k \right]$ où R rythme de transmission est égal à $k/n$ ; nous avons cherché s'il existait une matrice $\boxed{A}$ ayant k colonnes de longueur $m = n - k$ , t linéairement indépendantes.

En général, pour $t = \text{Entier}\left[(1 - R)\, k\right]$ (*) une telle matrice n'existait pas ; alors nous avons fait décroître t jusqu'à ce que nous puissions construire $\boxed{A}$ .

(*) Entier $(1 - R)\, k$ désigne la partie entière de $(1 - R)\, k$

La recherche de k vecteurs de longueur m = n - k , t linéairement indépendants devient très longue quand la longueur des vecteurs augmente ; ceci explique pourquoi nous nous sommes arrêtés, dans nos expériences à m = 14.

L'utilisation de tels codes nous donne une évolution de l'erreur au décodage : soit $Q_s$ . De tels codes ont été appelés codes linéaires systématiques.

2/ nous connaissons d'excellents codes dont les matrices de parité ont les propriétés que nous cherchons. Ce sont les Codes de Boose - Chandhuri - Hocquendhem [6] . Malheureusement, ces codes n'existent que pour des valeurs de n , k , t bien définies.

Comme nous savons que les vecteurs colonnes de tels codes ont une certaine indépendance, nous avons repéré parmi ces codes, dont la construction est rapide [6] , ceux qui avaient des colonnes de longueur m(ou des lignes de longueur m, quand nous ne trouvions pas de colonnes de longueur m) et avons choisi k colonnes (ou lignes) parmi celles qui formaient la matrice de parité du code. Ainsi, nous avons construit $|\overline{A}|$ ; t , indépendance linéaire des colonnes de $|\overline{A}|$ était inférieur à Entier $[(1 - R)\, k]$.

Ces codes nous ont donné une autre estimation de l'erreur au décodage : soit $Q_{BCH}$.

Nous avons de plus, calculé dans tous les cas $Q^*$ , probabilité minimum de mal décoder.

Résultats

Nous avons eu trois séries d'expériences.

- $R \simeq 0.75$ les résultats sont répertoriés dans le tableau 6 et les courbes tracées sur la figure 9
- $R \simeq 0.6$ correspond au tableau 7 et à la figure 10
- $R = 0.5$ correspond au tableau 8 et à la figure 11

⟶ – Déjà nous voyons que pour $R \simeq 0.75$ (tableau 6) les différentes probabilités $Q_a$ , $Q_s$ , $Q_{BCH1}$ , $Q^*$ restent à peu près constantes ; elles auraient même tendance à augmenter quand n croît ; ceci s'explique par le fait que $R \simeq C$ Capacité du canal ; or pour $R \geqslant C$ les différentes probabilités tendent vers 1 quand $n \longrightarrow \infty$ (cf théorème de Shannon).

Le coefficient $A = \lim_{n \to \infty} \left( \log_2 \frac{Q}{n} \right)$ est positif pour $R > C$ et pour $R = C$ il est nul. Nous n'avons plus une décroissance en exponentielle de n pour Q (probabilité de mal décoder) quand n tend vers l'infini.

⟶ Pour $R \simeq 0.6$ (tableau 7) les différentes probabilités décroissent quand n augmente R n'étant pas tout à fait constant et égal à 0.6 nous remarquons que $Q^*$ décroît par paliers, nous n'avons pu avoir pour $R = 0.75$ et $R = 0.6$ une valeur constante du rythme ; en effet $R = \frac{k}{n}$ où k et n sont entiers ; R et k étant fixés nous avions rarement k/R entier.

⟶ les expériences effectuées à $R = 0.5$ (tableau 8) permettent de garder R rigoureusement constant. Pour cette valeur de R , les différentes probabilités décroissent rapidement quand n augmente ; à partir de $k = 24$ , 1000 messages émis étaient correctement décodés pour le code aléatoire utilisé.

Pour les deux premières séries d'expériences ( $R \simeq 0.75$ et $R \simeq 0.6$) R n'étant pas constant nous avons tracé les différences $Q_a - Q^*$, $Q_s - Q^*$, $Q_{BCH} - Q^*$ en fonction de n . Ces différences nous donnent une évaluation des performances des codes utilisés.

Pour $R = 0.5$ nous avons tracé $Q^*$, $Q_s$ , $Q_a$ , $Q_{BCH}$ en fonction de n et sur papier semi – logarithmique, dans ce seul cas nous pouvions vérifier la décroissance exponentielle des différentes probabilités.

| k | n | $Q^*$ | $Q_a$ | $Q_s$ | $Q_{BCH}$ | R |
|---|---|---|---|---|---|---|
| 4 | 6 | 0.095 | 0.143 | | | 0.667 |
| 6 | 8 | 0.193 | | | | 0.75 |
| 8 | 11 | 0.177 | 0.463 | 0.374 | | 0.73 |
| 10 | 14 | 0.163 | 0.402 | 0.361 | 0.347 | 0.72 |
| 12 | 16 | 0.244 | 0.557 | 0.455 | 0.464 | 0.75 |
| 14 | 19 | 0.222 | 0.519 | 0.435 | 0.435 | 0.74 |
| 16 | 22 | 0.202 | 0.479 | 0.390 | 0.372 | 0.73 |
| 18 | 24 | 0.274 | 0.491 | 0.479 | 0.495 | 0.75 |
| 20 | 27 | 0.250 | 0.495 | 0.446 | 0.471 | 0.74 |
| 22 | 30 | 0.230 | 0.462 | 0.430 | | 0.73 |
| 24 | 32 | 0.295 | 0.488 | 0.487 | 0.471 | 0.75 |
| 26 | 35 | 0.271 | 0.509 | 0.485 | 0. | 0.745 |
| 28 | 38 | 0.250 | 0.486 | 0.458 | 0.455 | 0.74 |
| 30 | 40 | 0.310 | 0.490 | 0.515 | 0.482 | 0.75 |

Tableau 6. R 0.75 (C = 0.75)

(1000 méssages pour chaque valeur de n, k, R)

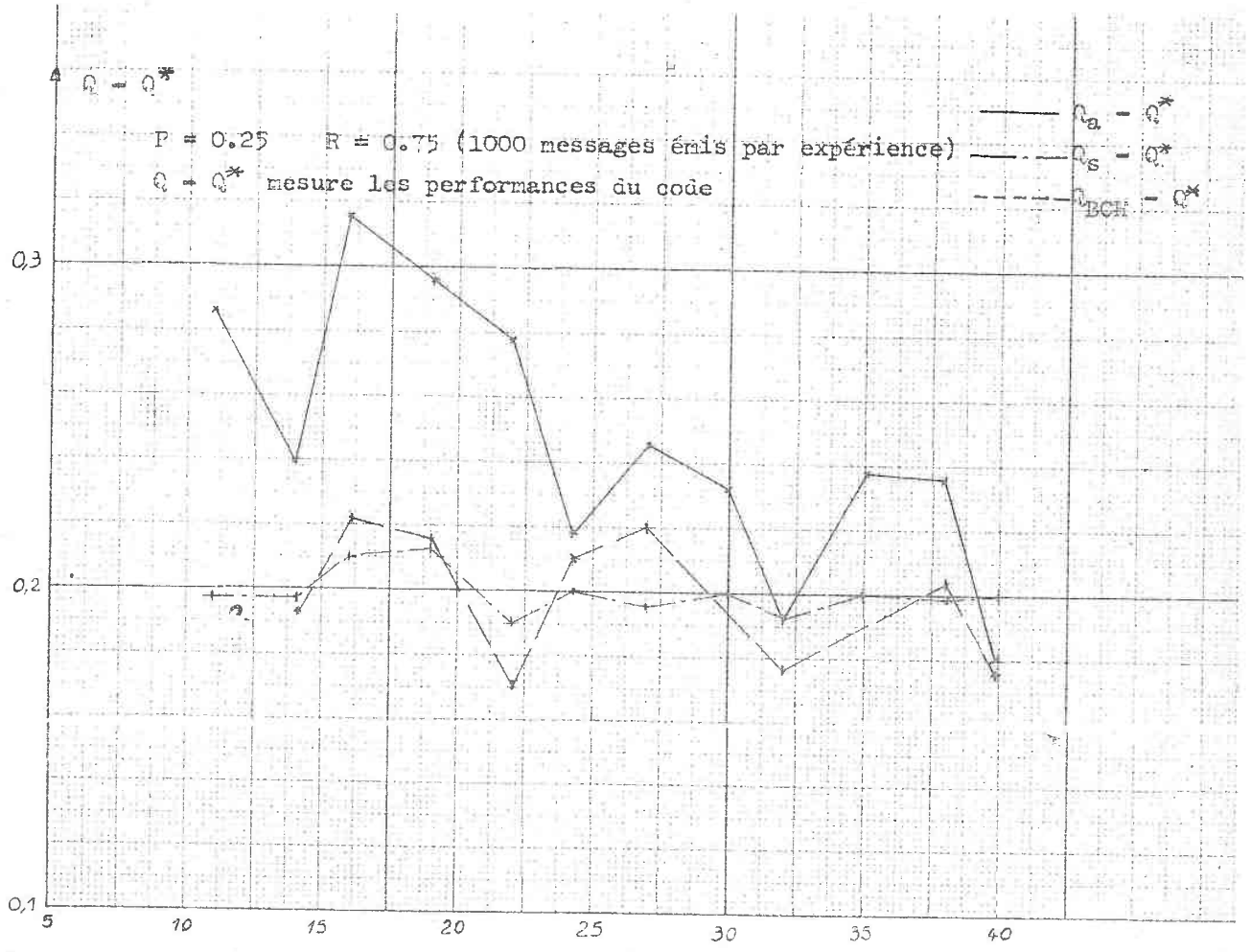



Fig. 2

| k | n | Q* | Qs | Qa | QBCH | R |
|---|---|---|---|---|---|---|
| 4 | 7 | 0.039 | 0.179 | 0.224 | 0.185 | 0.57 |
| 6 | 10 | 0.044 | 0.150 | 0.226 | 0.148 | 0.6 |
| 8 | 13 | 0.047 | 0.158 | 0.201 | 0.166 | 0.61 |
| 10 | 17 | 0.024 | 0.090 | 0.132 | 0.105 | 0.59 |
| 12 | 20 | 0.024 | 0.091 | 0.161 | 0.091 | 0.6 |
| 14 | 24 | 0.013 | 0.067 | 0.087 | 0.057 | 0.58 |
| 16 | 27 | 0.013 | 0.060 | 0.087 | 0.068 | 0.59 |
| 18 | 30 | 0.013 | 0.060 | 0.082 | 0.063 | 0.6 |
| 20 | 34 | 0.007 | 0.013 | 0.045 | | 0.59 |
| 22 | 37 | 0.007 | | 0.056 | | 0.595 |
| 24 | 40 | 0.007 | | 0.049 | 0.045 | 0.6 |
| 26 | 44 | 0.004 | | 0.029 | 0.015 | 0.59 |
| 28 | 47 | 0.004 | | 0.013 | | 0.595 |
| 30 | 50 | 0.004 | | 0.064 | | 0.6 |

Tableau 7 R 0.6 (C = 0.75)

(1000 messages émis pour chaque valeur de n,k,R)

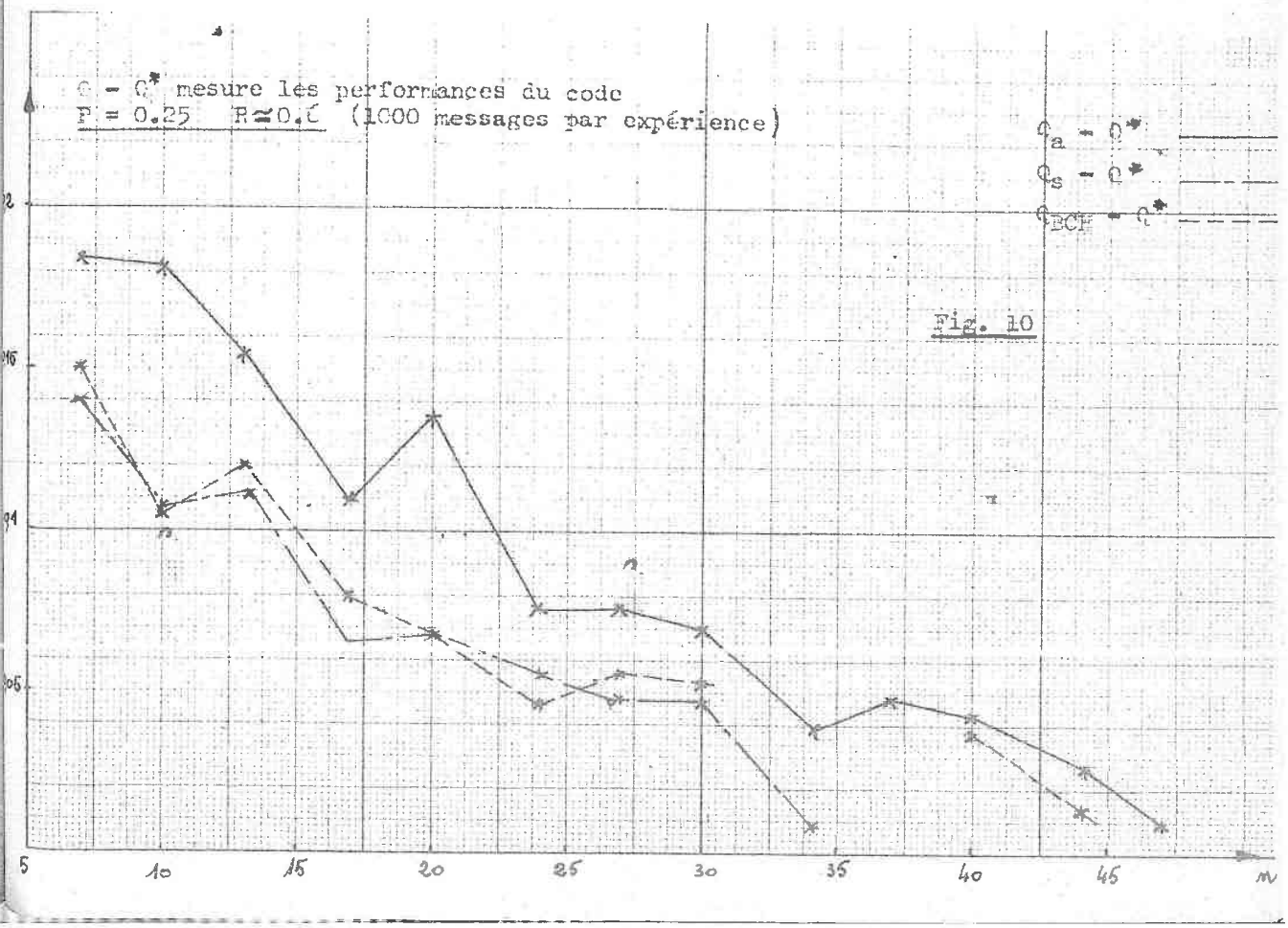



Fig. 10

| k | n | $Q^*$ | $Q_a$ | $Q_S$ | $Q_{BCH}$ |
|---|---|---|---|---|---|
| 4 | 8 | 0.015 | 0.182 | 0.077 | 0.083 |
| 6 | 12 | 0.008 | 0.090 | 0.056 | 0.045 |
| 8 | 16 | 0.004 | 0.076 | 0.028 | 0.065 |
| 10 | 20 | 0.002 | 0.030 | 0.018 | 0.019 |
| 12 | 24 | 0.001 | 0.020 | 0.012 | 0.009 |
| 14 | 28 | 0.0005 | 0.007 | 0.010 | 0.009 |
| 16 | 32 | } | 0.009 | | 0.008 |
| 18 | 36 | } $< 4.10^{-4}$ | 0.005 | | 0.000 |
| 20 | 40 | } | 0.005 | | |
| 22 | 44 | } | 0.001 | | |
| 24 | 48 | } | 0.000 | | 0.000 |

Tableau 8 -
R = 0.5 (C = 0.75)
(1000 messages émis pour chaque valeur de k,n,R)

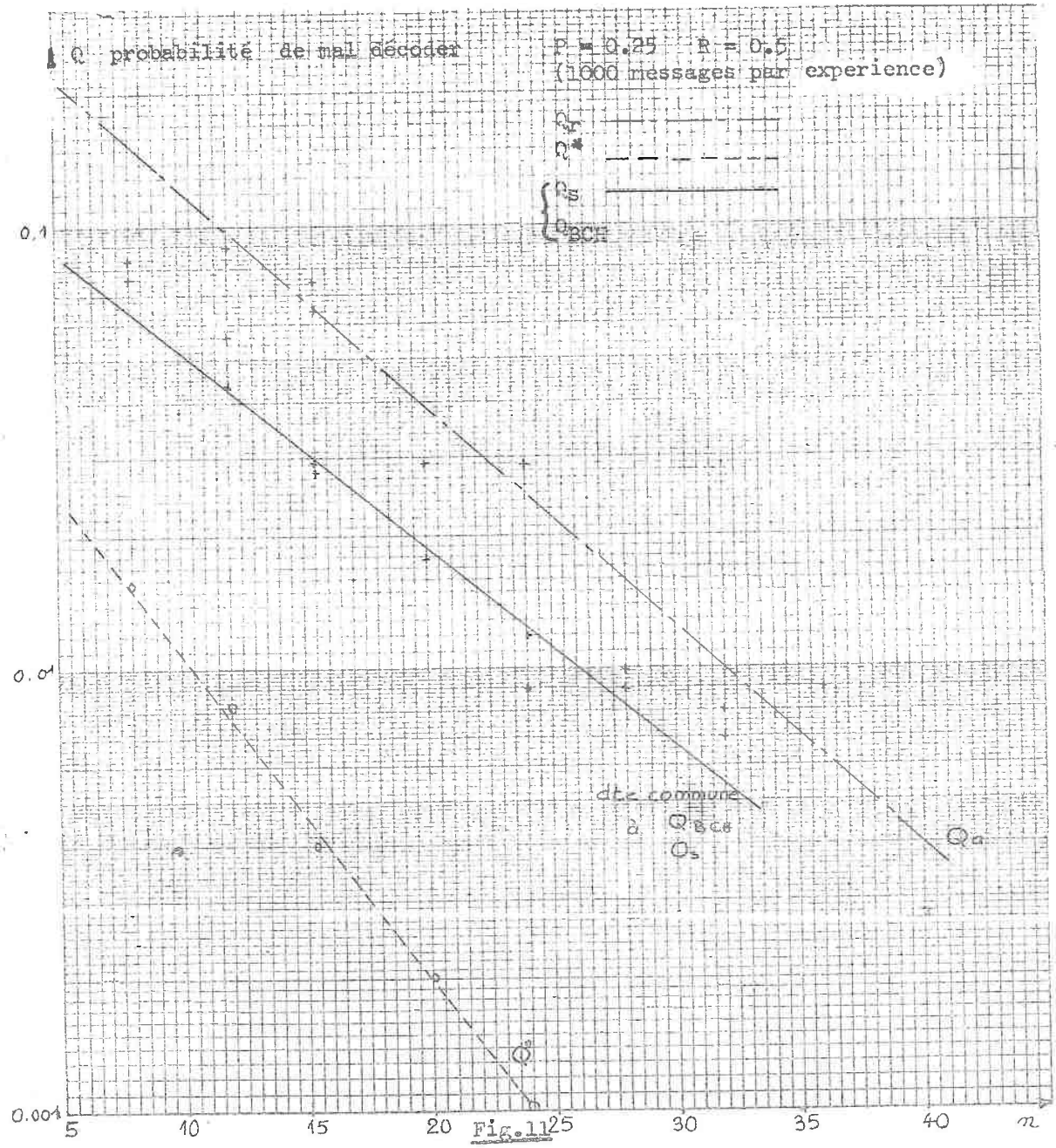



Fig. 11

Etudes des courbes - Conclusions

- R = 0.5 les différentes probabilités $Q_a$ , $Q^*$, $Q_s$ et $Q_{BCH}$ se répartissent autour de droites. Les valeurs de $Q_a$ étant rapidement faibles ($< 10^{-2}$) nous voyons que les points obtenus s'éloignent un peu de la droite moyenne (1000 messages émis ne permettent pas dans ce cas une bonne évaluation de $Q_a$). Les trois réseaux de courbes ( Fig 9 , 10 , 11) nous montrent que dans tous les cas les codes linéaires systématiques ( des deux types) sont meilleurs que les codes aléatoires quand n est faible. Quand n devient grand (derniers points des différentes courbes) nous avons à peu près les mêmes performances pour les trois types de codes.

Si nous comparons les codes linéaires systématiques construits par la procédure du chapitre V , aux codes construits sur des codes B C H , nous voyons que leurs performances sont équivalentes ; mais les codes systématiques semblent donner des réqultats plus réguliers d'un point à un autre (Fig 10).

Ces réseaux nous montrent que dès que n devient de l'ordre 40 , nous n'avons pas intérêt à construire un code linéaire systématique. Les codes aléatoires permettent une assez bonne précision ; si nous voulons que celle-ci soit meilleure il suffira de diminuer la vitesse. De plus, une étude des propriétés asymptotiques [7] des codes B C H a montré que ceux-ci devenaient très mauvais lorsque n augmente ; alors les codes aléatoires sont meilleurs.

II Comparaison avec le "Code Dictionnaire"

Nous avons voulu comparer les trois types de codes définis précédemment au code suivant appelé "Code dictionnaire"

Tous les mots de code possibles sont répertoriés dans un cahier (ou dictionnaire) ; un de ces mots est émis : le décodage est la recherche du mot qui est le plus "proche" du mot reçu les mots de code étant tous distants les uns des autres de t positions.

Pour pouvoir effectuer une comparaison valable, il fallait se placer aux mêmes valeurs de n , R,P . Il a donc fallu dans une première étude savoir combien de mots de codes appartenaient au maximum à un "code dictionnaire" de longueur n , de distance mini - mum t.

1° Nombre de mots de Code maximum d'un "code dictionnaire"

Soit G un code de mots de longueur n , comptrenant k digits d'information.

Tout mot de code est une combianison linéaire de k mots de code indépendants. Soit $\boxed{G}$ la matrice ayant pour lignes ces k mots de code ; $\boxed{G}$ est dite matrice de génération du code et a k lignes et m = n - k colonnes.

Le code G est un espace vectoriel de dimension k dont les éléments sont des vecteurs de longueur n formés sur le corps des entiers modulo 2. Il y a $2^k$ mots de code (ou vecteurs) différents appartenant à G. G est un code linéaire.

a) Somme des poids des mots du code G

Soit $\boxed{M}$ la matrice dont les vecteurs colonnes sont tous les mots de longueur k possibles ; $\boxed{M}$ est une matrice à k lignes et $2^k$ colonnes.

Soit $\boxed{G}$ la matrice de génération du code ; $\boxed{G}$ a k lignes et n colonnes.

Soit $\boxed{K}$ la matrice à $2^k$ lignes et n colonnes, dont chaque ligne est un mot de code possible.

$\boxed{K} = \boxed{M}^t \boxed{G}$ ($\boxed{M}^t$ désigne la matrice transposée de $\boxed{M}$)

Soit $\boxed{K}_i$ une colonne de $\boxed{K}$ $(i = 1,\ldots,n)$

$\boxed{K}_i = \boxed{M}^t \boxed{G}_i$ où $\boxed{G}_i$ est la $i^{ème}$ colonne de $\boxed{G}$

donc chaque élément de $\boxed{K}_i$ est une des $2^k$ combinaisons linéaires (modulo2) des k éléments de $\boxed{G}_i$

Supposons que $\boxed{G}_i$ ait j élément égaux à 0 ; pour former les éléments de $\boxed{K}_i$ n'interviendront que les $2^{k-j}$ combinaisons linéaires (modulo 2) différentes des k - j éléments non nuls de $\boxed{G}_i$. Elles feraient apparaître $2^{k-j-1}$ 1 et $2^{k-j-1}$ 0 si elles n'intervenaient chacune qu'une seule fois.

Soit une de ces combinaisons ; elle correspond à une ligne de $\boxed{M}^t$ qui a des digits bien déterminés dans les k - j positions correspondant aux 1 de $\boxed{G}_i$ et des valeurs quelconques dans les j autres positions ; une telle combinaison apparaîtra $2^j$ fois dans le calcul des éléments de $\boxed{K}_i$ ; $\boxed{K}_i$ aura donc $2^{k-1}$ digits égaux à 1 et $2^{k-1}$ égaux à 0.

La somme des poids des mots du code sera la somme de tous les éléments de $\boxed{K}$ :
Soit $n.\ 2^{k-1}$

Le code G a $2^k$ mots différents et $2^k-1$ sont à poids non nul. Soit d le poids minimum alors :

$$(7) \qquad \boxed{d \leqslant \frac{n.\ 2^{k-1}}{2^k - 1}}$$

b) Nombre de mots appartenant à G de poids minimum d

Supposons que G soit un code linéaire dont les mots sont de longueur n et de poids minimum d ; G est un code à k digits d'information.

Soit $B(n , d)$ le nombre de mots de code de G. Soit H le sous-ensemble de G tel que le dernier symbole des mots appartenant à H soit 0 ; H est un sous espace vectoriel de G : en effet si $v_1$ , $v_2$ , mots de code, appartiennent à H alors $v_1 \oplus v_2$ appartient à H ainsi que $\lambda v_1$ (où $\lambda$ est 0 ou 1).

Les éléments de H sont ceux pour lesquels l'élément de la dernière colonne de [K] est 0 ; il y a donc $2^{k-1}$ éléments de G qui appartiennent à H , soit la moitié. De ce qui précède, nous voyons que H est un code linéaire de longueur n - 1 ( la dernière composante n'intervenant pas, on peut la supprimer) et de poids minimum d ; il peut se trouver d'autres éléments qui sont nuls pour tous les mots de H ; on ne changera pas d , poids minimum, en changeant ces éléments.

alors $B(n - 1 , d) \geqslant \frac{B(n , d)}{2}$

soit la relation (8) $\boxed{2 . B(n - 1,d) \geqslant B(n,d)}$

La relation (7) peut s'écrire

$d \geqslant d . 2^k - n\, 2^{k-1}$

soit $2d \geqslant 2^k (2d - n)$

le code G a k digits d'information ; donc $B(n,d) = 2^k$

et $2d \geqslant B(n , d) . (2d - n)$

ce qui peut s'écrire : $\boxed{B (n , d) \leqslant \frac{2d}{2d - n} \quad \text{si } 2 d - n > 0}$ (9)

C'est la borne de Plotkin [6]

Soit $i = 2d - 1$ ; $2d - i = 1$ et on peut appliquer (9)

nous avons $\boxed{B(i , d) \leqslant 2d}$ (10)

Utilisons $n - 1$ fois de suite (8)

Soit $$B(n , d) \leqslant 2^{n-i} B (i , d)$$

ou $$\boxed{B(n , d) \leqslant 2d. 2^{n-2d+1}} \qquad (11)$$

La relation (11) donne le nombre maximum de mots de code appartenant à un code linéaire G de longueur n et de poids minimum d.

2° Codage utilisé - Comparaison avec les codes par bloc.

Nous avons un "code dictionnaire" qui a la propriété suivante : tous les mots de code sont distants de t + 1 positions et ont pour longueur n.

Le premier mot étant le mot (0 ... 0), tous les autres mots ont un poids au moins égal à t + 1 . Soit $G_1$ un tel code et soit G le code linéaire de longueur n et de poids minimum t + 1 ; $G_1$ est inclus dans G.

Pour comparer $G_1$ aux codes par blocs étudiés précédemment il fallait se placer en des points ayant même valeur de k , P et R. Soit $R_1$ le rythme de transmission lié au code $G_1$ , P étant la probabilité d'effaçage.

a/ Methode de décodage.

Nous avons émis un des mots possibles du code $G_1$ et l'avons envoyé à travers le canal. A la réception nous avons compté le nombre d'effacements ; si ce nombre était inférieur à t, nous avons pris comme mot émis celui qui était identique au mot reçu dans les positions non effacées ; dans le cas contraire nous avons comparé le mot reçu à tous les mots de $G_1$ et avons noté tous les mots qui lui étaient semblables dans les positions non effacées ; nous en avons pris un au hasard parmi eux, comme étant le message envoyé ; c'est ce que nous avions aussi fait dans le décodage par bloc.

b/ Rythme maximum de transmission de $G_1$

Nous avons cherché quel était le rythme $R^*$ du code G linéaire qui contient le code $G_1$

G a $2^{nR^*}$ mots de code

donc $2^{nR^*} = B(n, d) \leqslant 2d \cdot 2^{n-2d+1}$

$$2^{nR^*} \leqslant 2^{n-2d+2+\log_2 d}$$

Soit

$$\boxed{R^* \leqslant \frac{n - 2d + 2 + \log_2 d}{n}} \qquad (12)$$

où n est la longueur des mots de code et d le poids minimum égal à t + 1.

c/ Expériences

Nous avons fait les essais en choisissant t de telle sorte que $R^* \geqslant 0.5$ et avons à chaque fois noté le nombre de mots du code $G_1$ pour avoir la valeur exacte de $R_1$ rythme de transmission lié au code $G_1$.

Ces essais ont été fait pour trois valeurs de n( n = 8 , 12 , 16) et P = 0.25 ; pour des longueurs plus grandes des messages de tels essais ne furent pas envisageables ; en effet pour stocker les mots de code $G_1$ , il a fallu utiliser les mémoires périphériques et le temps de calcul s'en est trouvé considérablement augmenté.

Pour n = 20 (t = 6) 8 heures de calcul sur I B M 1130 ne permettent pas de construire entièrment $G_1$ !

Les résultats se trouvent dans le tableau 9.

Le tableau 10 nous donne les performances correspondantes des codes par bloc.

| n | t | $Q_1$ | B(n,d) | $R_1$ | $R^*_{calculé}$ |
|---|---|---|---|---|---|
| 8 | 3 | 0.026 | 16 | 0.5 | 0.5 |
| 12 | 4 | 0.006 | 16 | 0.333 | 0.51 |
| 16 | 5 | 0.009 | 128 | 0.437 | 0.537 |
| 12 | 3 | 0.061 | 128 | 0.583 | 0.8 |

Tableau 9 . Code dictionnaire

P = 0.25 R 0.5

| n | $R_1$ | R | $Q_1$ | $Q_a$ | $Q_s$ | $Q_{BCH}$ | $Q^*$ |
|---|---|---|---|---|---|---|---|
| 8 | 0.5 | 0.5 | 0.02[illegible] | 0.182 | 0.077 | 0.083 | 0.015 |
| 12 | 0.333 | 0.5 | 0.006 | 0.090 | 0.056 | 0.045 | 0.008 |
| 16 | 0.437 | 0.5 | 0.009 | 0.076 | 0.028 | 0.065 | 0.004 |
| 22 (13) | 0.583 | 0.61 | 0.061 | 0.201 | 0.158 | 0.166 | 0.047 |

Tableau 10 : Comparaison entre les différents codes

P = 0.25

Pour n = 12 nous avons fait deux expériences : une avec t = 4 qui nous a donné une valeur de $R_1$ très inférieure à 0.5 et une avec t = 3 qui a donné une valeur de $R_1$ supérieure à 0.5 et voisine de 0.6 ; nous pouvons dans ce cas comparer les performances du "code dictionnaire" (n = 12 , t = 3) à celle des codes par bloc de rythme de transmission 0.6 ; malheureusement aucun essai n'a été effectué pour ces codes avec n = 12 ; nous avons des résultats pour n = 13 ; ils nous serviront de point de comparaison (le théorème de Shannon dit que ces résultats sont meilleurs que ceux obtenus pour n = 12 avec des mêmes valeurs de Capacité et de rythme de transmission). Soit $Q_1$ le pourcentage de messages mal décodés donné par les "codes dictionnaires".

Nous voyons que les valeurs obtenues pour le "code dictionnaire" sont bien meilleures que celles obtenues à l'aide des codes par blocs. Il serait bon d'avoir une comparaison entre les différents temps de calcul avant de conclure.

3° Temps de Calcul.

a) nombre d'opérations élémentaires (additions et multiplications

Nous avons utilisé pour la simulation des "codes dictionnaires" des mémoires périphériques, mais pour la simulation des codes par blocs nous n'avons travaillé qu'avec les mémoires centrales ; comparer les temps de calculs de ces deux méthodes n'aurait donc guère de sens.

Nous appelons opération élémentaire l'addition ou la multiplication ; leur temps de calcul sera à peu près équivalent car toutes nos opérations se font modulo 2.

Nous devons dans une première phase construire, ou le dictionnaire où sont répertoriés tous les mots de code, ou la matrice $|\underline{A}|$ de parité : les calculs nécessaires à cette opération sont un investissement fait une fois pour toutes. Dans une seconde phases, nous devons décoder tout mot reçu. Nous comparerons le nombre d'opérations élémentaires pour l'une et l'autre phase. Cette comparaison se fera pour R (rythme de transmission du canal) et P (probabilité d'effaçage) fixés , n (longueur totale des messages) variant.

Construction des codes.

. Code aléatoire

Soit $NC_a$ le nombre d'opérations élémentaires nécessaires à la construction de la matrice $|\underline{A}|$ de parité.

Soit $k = nR$ = nombre de digits d'information.

$|\underline{A}|$ est une matrice à k colonnes et n - k lignes, il faudra construire aléatoirement k (n - k) digits.

Pour chaque digit émis aléatoirement (cf sous programme ENGE) il faut au plus 4 opérations élémentaires.

Soit $\boxed{NC_a = 4.\ k\ (n - k)}$

$$\boxed{NC_a = 4\ n^2\ R\ (1 - R)}$$

. Code dictionnaire

Il y a $2^n$ mots distincts. Soit $NC_d$ le nombre d'opérations élémentaires nécessaires pour déterminer les $2^{nR}$ mots distincts de code.

Chaque mot a pour longueur n et tous les mots sont distincts de t + 1 positions donc ont un poids $\geqslant$ t + 1 (sauf pour le mot nul). Pour calculer le poids de chacun de ces $2^n$ vecteurs, on aura n additions à faire donc en tout $n.2^n$ opérations élémentaires.

Les mots de code seront pris parmi les mots de poids $\geqslant t+1$ ; ces mots sont au nombre de $2^{n-t-1}\ C_n^{t+1}$ ; chacun de ces mots sera comparé au moins à un mot déjà répertorié ; la comparaison de deux mots nécessite 2 n opérations ( on forme le vecteur somme et on calcule son poids).

alors $NCd > n\ 2^n + 2n.\ 2^{n-t-1}\ C_n^{t+1}$

$$NCd > n.2^n + 2n\ 2^{n-t-1}$$

d'après les études précédentes nous savons que

$t \leqslant n - k$ (k étant le nombre de digits d'information)

alors $NCd > n.\ 2^n + n\ 2^k$

Soit $$\boxed{NC_d > n\ (2^n + 2^{nR}) = NC_1}$$

Dès que n croît $NC_1$ devient rapidement supérieur à $NC_a$ : $NC_1$ est une fonction exponentielle de n ; $NC_a$ est une fonction puissance de n.

## Décodage

. Code aléatoire

Soit $ND_a$ le nombre d'opérations élémentaires moyen nécessaire pour pouvoir décoder un mot reçu.

Il faudra en moyenne résoudre un système linéaire (modulo 2) de k P inconnues à $(n - k)(1-P)$ équations.

Posons $m = k\ P = n.\ R.\ P$ = nombre d'inconnues.

$l = (n-k)(1-P) = n(1-R)(1-P)$ = nombre d'équations

Il y a deux phases pour la résolution :

- trianguläriser une matrice à l lignes et m colonnes : cette opération se fait sur le corps des entiers modulo 2 et chaque digit a une probabilité $\frac{1}{2}$ d'être le digit nul.

- résoudre le système alors obtenu.

Soit $B_T$ le nombre d'opérations élémentaires nécessaire à la triangularisation et $B_r$ celui nécessaire à la résolution.

Au cours de la triangularisation, supposons que nous en soyons à la $j^{ème}$ ligne (j=1,..., l-1) ; la $j^{ème}$ ligne aura son $j^{ème}$ digit égal à 1 (dans le cas contraire il y aura en permutation avec une des lignes suivantes) ; on doit combiner cette ligne aux l-j lignes suivantes ; il y a combinaison si le $j^{ème}$ digit de ces lignes est égal à 1 et chaque combinaison (somme de deux vecteurs de longueur m+1-j) nécessitera m+1-j sommes. Il y aura en moyenne $\frac{l-j}{2}$ combinaisons à effectuer. Ces considérations permettent de calculer $B_T$ .

$$B_T = \frac{1}{2} \sum_{j=1}^{l-1} (l-j)(m-j+1)$$

en tenant compte de ce que :

$$\sum_{j=1}^{n} j^2 = \frac{n(n+1)(2n+1)}{6} \quad ; \quad \sum_{j=1}^{n} j = \frac{n(n+1)}{2}$$

Il vient :

$$2 B_T = \sum_{j=1}^{l-1} j^2 - \sum_{j=1}^{l-1} j(m+1+l) + l\,(l-1)(m+1)$$

$$2 B_T = \frac{l(l-1)(2l-1)}{6} - \frac{l(l-1)(m+1+1)}{2} + l(l-1)(m+1)$$

$$2 B_T = \frac{l(l-1)}{6} (3m - l + 2)$$

Résolvons le système ainsi obtenu : les j - 1 premieres racines étant connues, il faut pour calculer la $j^{ème}$ , effectuer j -1 multiplications et j - 1 additions (le calcul de la 1ère racine ne nécessite aucune opération : c'est une affectation)

$$B_r = 2 \sum_{j=2}^{l} (j - 1) = l\ (l - 1)$$

et $ND_a = B_T + B_r = \frac{l(l - 1)}{12} (3m - l + 14) < \frac{l^2}{12} (3m - l + 14)$

remplaçons l , m par leurs valeurs en fonction de n , R , P

$$\boxed{ND_a < \frac{n^2}{12} (1 - R)^2 (1 - P)^2 (3n\ R\ P - n\ (1 - R)\ (1 - P) + 14)}$$

. Code dictionnaire

Comme pour le code aléatoire nous allons supposer que nous ne pouvons faire les intersections logiques de deux mots sur les machines : à chaque fois nous construisons le vecteur somme et calculons son poids.

Soit $ND_d$ le nombre moyen d'opérations élémentaires nécessaires pour décoder un mot. Le mot reçu a pour longueur n ; il y a en moyenne n(1 - P) digits non effacés ; il faut comparer ce mot à tous les mots répertoriés dans le dictionnaire : on somme les deux vecteurs à comparer ; dès qu'un digit de la somme est 1 on arrête la comparaison et on teste le mot suivant du dictionnaire. Chaque comparaison nécessitera en moyenne $\frac{n(1 - P)}{2}$ opérations. Il faudra en moyenne comparer chaque mot reçu à $2^{nR - 1}$ mots de code ;

$$ND_d = \frac{n(1 - P)}{2} 2^{nR - 1}$$

$$\boxed{ND_d = n(1 - P)\ 2^{nR - 2}}$$

b) Comparaisons et limites d'utilisations du code dictionnaire

Il est intéressant de voir à partir de quelles valeurs de n, $NC_d \gg NC_a$ et $ND_d \gg ND_a$.

Les nombres $NC_d$ et $NC_a$ correspondent au temps de calcul nécessaire à la construction deu code utilisé ; c'est un investissement, il peut être très élevé ; ce qui importe c'est le temps de calcul nécessaire au décodage.

Nous supposons que nous avons à notre disposition une machine possédant un grand nombre de mémoires centrales : la mise en place d'un code dictionnaire se fera en n'utilisant aucune mémoire périphérique (il faut n(1 - R) R mémoires pour le code aléatoire et n $2^{NR}$ pour le code dictionnaire). Nous avons vu précédemment que pour des messages de même longueur n, transmis au Rythme R à travers le canal de capacité C , les codes dictionnaires étaient plus performants que les codes aléatoires. Les formes de $ND_d$ et $ND_a$ nous disent que $ND_d$ croît plus vite que $ND_a$ quand n augmente ; nous allons nous fixer comme limite d'utilisation d'un code dictionnaire la valeur $n_1$ telle que pour $n < n_1$ $\frac{ND_d}{ND_a} \leq 100$

Les codes dictionnaires étant plus performants que les codes aléatoires, on peut admettre un temps de décodage plus grand. Au dessus de cette valeur, malgré leurs performances, nous ne pourrons nous permettre d'utiliser les codes dictionnaires, leurs temps de calcul nous l'interdisant.

Dans le tableau 12 , nous avons calculé $NC_a$ , $NC_d$ , $ND_a$ $ND_d$ pour P = 0.25 (Probabilité d'effaçage) et pour R = 0.5 , 0.6 , 0.75 (rythme de transmission) , n croissant de 2 à $n_1$ .

Dans le cas où la dimension de notre machine nécessite l'emploi de mémoires périphériques pour l'utilisation du code dictionnaire, nous choisirons le code aléatoire.

| N | $NC_a$ | $NC_d$ | $ND_a$ | $ND_d$ | |
|---|---|---|---|---|---|
| 2 | 4 | 24 | 1 | 1 | |
| 4 | 16 | 160 | 2 | 3 | |
| 6 | 36 | 864 | 6 | 9 | |
| 8 | 64 | 4352 | 11 | 24 | |
| 10 | 100 | 21120 | 16 | 60 | |
| 12 | 144 | 99840 | 24 | 144 | |
| 14 | 196 | 462336 | 32 | 336 | |
| 16 | 256 | 2105344 | 42 | 768 | |
| 18 | 324 | 9455616 | 53 | 1728 | |
| 20 | 400 | 41984000 | 66 | 3840 | |
| 22 | 484 | 184[illegible]480 | 79 | 8448 | R = 0.5 |
| | | | | | |
| 2 | 4 | 25 | 1 | 1 | |
| 4 | 15 | 170 | 2 | 4 | |
| 6 | 35 | 914 | 4 | 14 | |
| 8 | 61 | 4542 | 7 | 42 | |
| 10 | 96 | 21760 | 12 | 120 | |
| 12 | 138 | 101833 | 17 | 331 | |
| 14 | 188 | 468210 | 24 | 886 | |
| 16 | 246 | 2121980 | 31 | 2328 | |
| 18 | 311 | 9501368 | 41 | 6017 | R = 0.6 |
| | | | | | |
| 2 | 3 | 27 | 1 | 1 | |
| 4 | 12 | 192 | 1 | 6 | |
| 6 | 27 | 1040 | 2 | 25 | |
| 8 | 48 | 5120 | 3 | 96 | |
| 10 | 75 | 24100 | 5 | 340 | |
| 12 | 108 | 110592 | 8 | 1152 | R = 0.75 |

Tableau 11 ( C = 0.75 ; P = 0.25 )

4° Conclusions

a nous voyons que dès que $n_1$ longueur des messages émis, est grand, les performances des codes aléatoires sont excellentes ; de plus au point de vue temps de calcul, ils sont préférables aux codes dictionnaires qui ont des performances meilleures mais des temps de calcul beaucoup trop longs.

b Dans les diverses méthodes de décodage utilisées nous avons "relancé" le décodage : chaque fois qu'il y avait trop de digits à déterminer, nous en avons "deviné" un certain nombre ; il serait intéressant de chercher ce que vaut l'information apportée par cette détermination arbitraire d'un ou plusieurs digits.

Soient m digits effacés sur un mot de longueur n ; il est possible de corriger efficacement t digits. Nous supposons $m > t$ ; il faudra donc "deviner" $m - t$ digits. On peut représenter l'opération de relance définie ci-dessus comme un canal à $2^m$ entrées et $2^m$ sorties : chaque symbole d'entrée, ou de sortie, correspond à un mot possible de longueur m.

Pour pouvoir décoder correctement il nous faut "deviner" $m - t$ digits ; supposons tout d'abord $m - t = 1$ , on aura à deviner 1 digit (on le place en 1ère position) : s'il est juste on aura à la sortie du canal le mot correctement reçu, s'il est faux il y aura $2^{m-1}$ mots possibles. Cette opération correspond au canal représenté par la Figure 13. On a placé en première position le message émis.

Les seules liaisons possibles sont celles de la figure 13 et les probabilités conditionnelles sont :

$$P(x \longrightarrow y) = \begin{cases} 1/2 & \text{si le digit est bien deviné} \\ (1/2)^m & \text{sinon.} \end{cases}$$

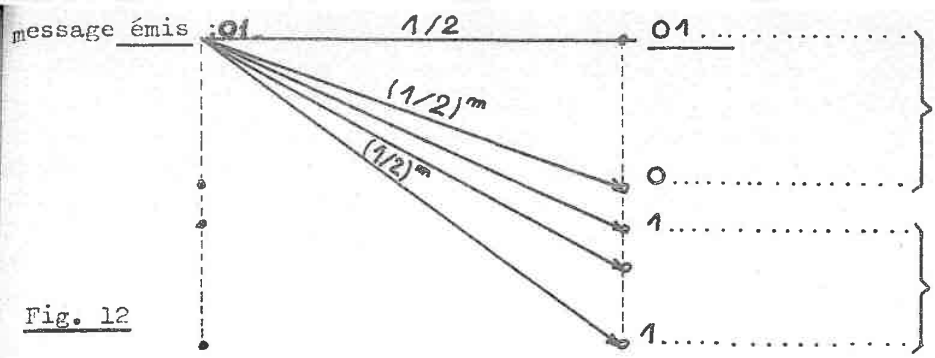



Fig. 12

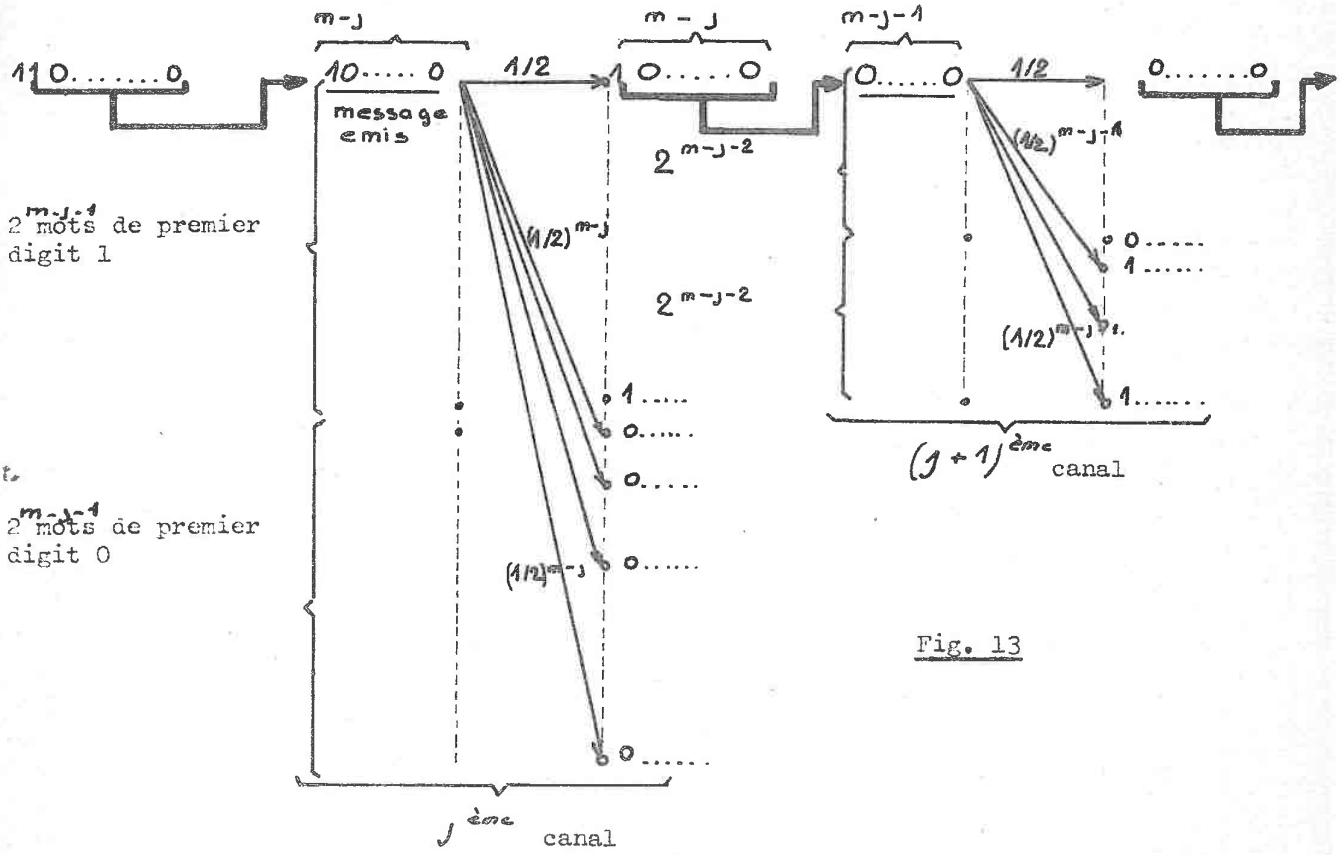



Fig. 13

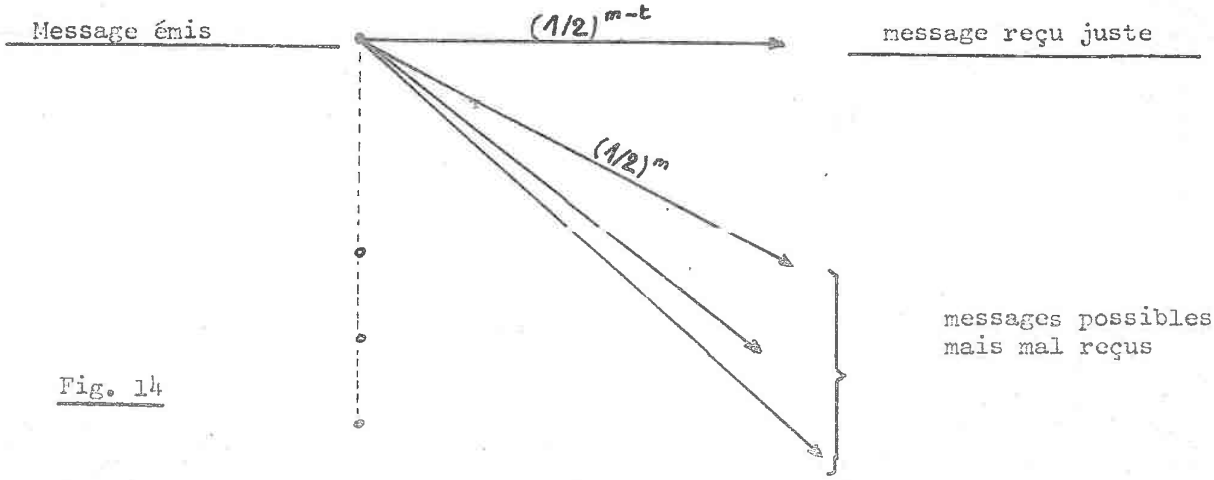



Fig. 14

Si l'on doit deviner m - t digits on peut représenter le canal comme une succession de m - t canaux : c'est le schéma de la figure 14, et on peut schématiser le canal qui doit deviner m- t digits par la Figure 15.

Soit N = nombre de messages possibles mal reçus.

$$\begin{cases} \text{à la détermination du } 1^{er} \text{ digit il y a } 2^{m-1} \\ \text{à la détermination du 2e digit il y a } 2^{m-2} \text{ mots possibles} \\ \text{à la détermination du m-t digit il y a } 2^{t} \end{cases}$$

$$N = \sum_{j=1}^{m-t} 2^{m-j} = 2^{m-1} \sum_{j=1}^{m-t} \left(\frac{1}{2}\right)^{j-1}$$

$$N = \frac{2^{m-1}\left(1 - 2^{t-m}\right)}{1 - \frac{1}{2}} = 2^{m} - 2^{t}$$

On cherche l'Information I transmise par le canal

$$I = H(X) + H(Y) - H(X, Y)$$

Les mots ont à l'émission et à la réception les mêmes probabilités d'apparition : $2^{-m}$ pour chaque mot.

$$H(X) = H(Y) = -\sum_{j=1}^{2^m} 2^{-m} \log_2 2^{-m} = m$$

$$P(x, y) = \begin{cases} 2^{-m}\left(2^{t-m}\right) & \text{si le message est bien reçu.} \\ 2^{-m}\ 2^{-m} & \text{si le message est mal reçu.} \end{cases}$$

$$H(X,Y) = -\sum_{j=1}^{2^m} \left(2^{t-2m} \log_2 2^{t-2m} + \frac{2^m - 2^t}{2^{2m}} \log_2 2^{-2m}\right)$$

$$H(X,Y) = 2^m \left( 2^{t-2m} (t - 2m) + 2m\, 2^{-2m} (2^m - 2^t)\right)$$

$$H(X,Y) = 2m - t\, 2^{t-m}$$

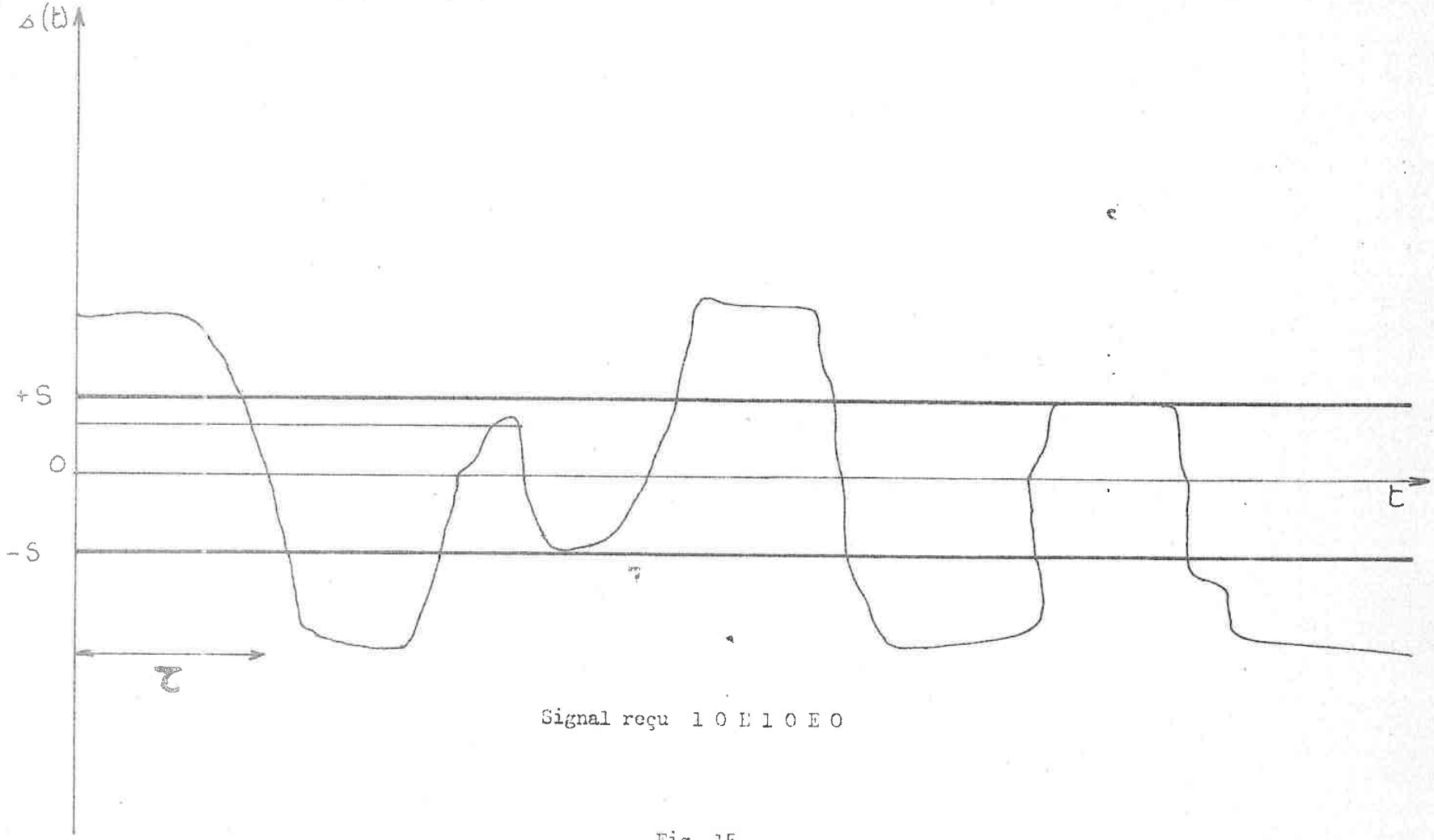
et $\boxed{I = t\, 2^{t-m}}$

Les m - t digits que l'on est forcé de deviner, n'apportent, même si on a deviné juste, aucune information. Par contre les t restants sont de l'information communiquée par la résolution des équations complètes (décodage à faire ensuite), mais la probabilités d'avoir deviné juste ces m - t digits nécessaires pour effectuer le décodage est $2^{t-m}$ d'où

$I = t\, 2^{t-m}$ que l'on retrouve.

Ce calcul pose le problème suivant : peut-on coder ce nouveau canal que nous appellerons super-canal, pour lui appliquer le grand Théorème de Shannon ; chaque message d'entrée serait en fait un groupe de messages de longueur n qui peuvent être décodés (assez d'équations) ou pas (pas assez d'équations) ; le canal ainsi obtenu serait un canal binaire et nous envisagerons dans des études ultérieures de l'étudier.

c Le canal binaire à effaçage est la représentation théorique d'un canal physique qui a deux niveaux + A , - A , correspondant aux digits 0 et 1 ; la durée d'un digit est $\tau$ ; on a effaçage lorsque l'amplitude du signal reçu est inférieure à un seuil A (Figure 16)




Signal reçu 1 0 E 1 0 E 0

Fig. 15

Si nous n'avions pas imposé de seuil, la valeur du signal sur l'intervalle de temps $\tau$ donne la valeur du digit (0 ou 1). Dans le cas de la figure 16 on aurait reçu 1001010 ; et la représentation théorique du canal physique aurait été celle du "canal binaire symétrique" ou B.S.C (Figure 17)

Une autre voie de recherche serait la comparaison de ces deux canaux, B E C et B S C, lorsque l'on travaille à énergie constante ; on chercherait, suivant la valeur du seuil S pour le B E C , et le test qui permet d'affecter une valeur au digit pour le B S C , quelle représentation donne la capacité maximum.

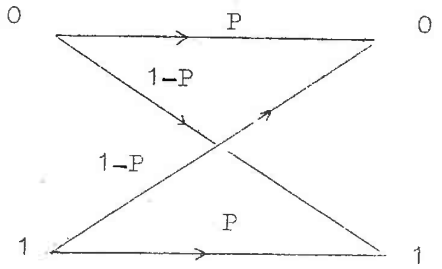



Fig. 17

## VII Bibliographie


[1] ASH . R B 1965 "Information Theory" John Wiley & Sons, Inc... , NEW YORK

[2] GOLDMAN. S 1953 "Information Theory" - Prentice Hall New York.

[3] WOLFOWITZ. J. "Coding Theorems of Information Theory" Springer - Verlag, O H G Berlin ; Math Rev, 26 : 1227

[4] SHANNON. C. E. 1948 "A Mathematical Theory of Communications" Bell System Tech. J 27 : 379 - 423 ; 623 - 656

[5] ELIAS. P. 1956 "Coding for two noisy channels" Information Theory - Colin Cherry (Ed) Academic Press - New York (p. 61 - 74)

[6] PETERSON. W. W. 1961 "Error Correcting Codes" The M I T Press Cambridge Mass ; Math Rev 22 : 12003

[7] BERLEKAMP. E. R. 1968 "Algebraic Coding Theory" Mac Graw Hill Inc. New York.

[8] REZA. F. M. 1961 "An Introduction to Information Theory" Mac Graw Hill Inc New York.

[9] PIERCE. J. R. 1961 "Symboles, signaux, Bruits" Harper and Row, publishers Incorporated - New York.

NOM DE L'ETUDIANT : Melle MATHE Françoise

Nature de la thèse : Spécialité en Mathématiques Appliquées

Vu, Approuvé

et permis d'imprimer

Nancy, le 12 Mai 1969

LE DOYEN

J, AUBRY