Séminaire Dubreil.
Algèbre et théorie
des nombres

M. P. Schützenberger

Une théorie algébrique du codage

Séminaire Dubreil. Algèbre et théorie des nombres, tome 9 (1955-1956), exp. n° 15, p. 1-24

<http://www.numdam.org/item?id=SD_1955-1956__9__A10_0>

© Séminaire Dubreil. Algèbre et théorie des nombres
(Secrétariat mathématique, Paris), 1955-1956, tous droits réservés.

L'accès aux archives de la collection « Séminaire Dubreil. Algèbre et théorie des nombres » implique l'accord avec les conditions générales d'utilisation (http://www.numdam.org/conditions). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

Numdam
Article numérisé dans le cadre du programme
Numérisation de documents anciens mathématiques
http://www.numdam.org/

Faculté des Sciences de Paris
-:-:-:-

Séminaire P. DUBREIL et C. PISOT
(ALGÈBRE et THÉORIE DES NOMBRES)
Année 1955/1956
-:-:-:-

27 février 1956

Exposé n° 15

UNE THÉORIE ALGÉBRIQUE DU CODAGE,

par M.P. SCHÜTZENBERGER.

Introduction.

Soient donnés deux ensembles discrets (en général finis dans la pratique) :

Λ_0 : l'ensemble des messages élémentaires

A_0 : l'ensemble des lettres ou "alphabet".

En théorie des communications, le problème du codage consiste à représenter les messages, c'est-à-dire les suites de messages élémentaires par des suites de lettres selon une règle fixée à l'avance (un "code" ou "dictionnaire") de telle sorte que la retraduction soit possible et ceci en fonction d'exigences techniques de nature diverse.

Par exemple, Λ_0 étant l'ensemble des signes typographiques habituels et A_0 un "alphabet" consistant en les trois "lettres" "point" "trait" et "intervalle".

Le code télégraphique morse est le système de convention dans lequel "a" est représenté par "point trait intervalle" , "b" par "trait point point point intervalle" ... etc. On appellera respectivement "codage" et "décodage" les deux opérations inverses qui font passer des suites de messages aux suites de lettres et réciproquement.

Plus généralement on pourrait voir dans tout langage, naturel ou artificiel, un code (plus souvent d'ailleurs : une série de codes superposés : phonémiques, grammaticaux, sémantiques, etc.) traduisant en sons, signes ou gestes des idées, des sentiments, etc. Une définition encore plus générale a été étudiée par J. Riguet [1] , et nous désignerons ici, pour abréger, sous le nom de "codes", certaines structures qui idéalisent les restrictions plus ou moins explicitement ou rigoureusement admises par la quasi-totalité des systèmes utilisés dans la théorie des communications stricto sensu :

1°) Universalité : Toute suite de messages élémentaires correspond à une suite de lettres au moins.

2°) Catégoricité : Cette suite de lettre est unique pour une suite de messages élémentaires donnée.

3°) Univocité du décodage : Réciproquement, si une suite de lettres correspond à une suite de messages élémentaires, cette dernière est unique.

4°) Isomorphie : Si les messages " λ " et " μ " sont codés respectivement par " ℓ " et "m" , le message " $\lambda\mu$ " est codé par " ℓ m".

Le code morse satisfait rigoureusement à ces quatre conditions et aussi, dans une certaine mesure, ce code compliqué qu'est la transcription orthographique française du langage parlé(1).

Formellement il est clair que les conditions énoncées conduisent à la :

Définition :

Λ et A étant respectivement les demi-groupes libres engendrés par Λ_0 et A_0 , une application $\mathcal{C}$ de Λ_0 sur une partie P_0 de A sera un code, si et seulement si l'extension de $\mathcal{C}$ à Λ détermine un isomorphisme de Λ sur le sous-demi-groupe P de A engendré par les suites de lettres constituant P_0 .

On appellera "mot" les suites de lettres constituant P_0 et, pour abréger, "message" aussi bien les éléments de P que ceux de Λ . Un code sera défini par la donnée de Λ_0 , A_0 et $\mathcal{C}$, ou plus fréquemment par celle de A et de P , puisque, A et P étant des demi-groupes, leur donnée implique celle de leurs générateurs.

---

(1) Les contre-exemples suivants illustrent la signification des conditions précédentes :

ad 1°) : le phonème "t*" (le click dental du Bantou, utilisé en France pour stimuler les chevaux) est intranscriptible dans l'orthographe française.

ad 2°) : le phonème "K" est codé K (Képi) ou ch (choléra) ou c (carotte) ou qu (quai).

ad 3°) : la graphie "ent" code plusieurs phonèmes distincts (vent, vient viennent).

ad 4°) : la graphie "oi" qui se lit "ua" et devrait s'écrire "oua".

La définition précédente fait apparaitre la théorie de codage comme une application de la théorie des demi-groupes. Il est particulièrement remarquable que les concepts fondamentaux de cette dernière, introduits par P. Dubreil [3], [4], [5] en 1941, et étudiés depuis par lui-même et son école du point de vue abstrait, aient des interprétations immédiates et importantes sur le plan de la réalisation concrète des machines codeuses ou transcodeuses.

Sous un autre angle, la théorie des événements récurrents de W. Feller [6], dans laquelle on étudie des processus stochastiques sur les suites de lettres à partir d'autres processus définis sur les suites de messages élémentaires, et réciproquement se rattache étroitement à la théorie du codage, comme l'a montré Mandelbrot qui a utilisé cette analogie dès 1951 [7], [8] : les processus récurrents sont des codes unitaires au sens que nous donnerons plus loin à ce terme. En retour, l'extension par Feller lui-même de sa théorie à des alphabets topologiques spéciaux, ouvre la voie à des généralisations intéressantes qui feront l'objet d'un exposé ultérieur.

Enfin, la théorie développée ici est une théorie "sans bruit" dans laquelle on ne se préoccupe pas des problèmes que pose au décodeur une altération d'origine aléatoire des messages à décoder. Le contraste avec les travaux actuels sur cette question (Cf : [10] ou [11] par exemple) est moins grand qu'il ne paraît. En particulier, la notion de "code absorbant" y joue un rôle considérable, et ne sera ici l'objet que de remarques incidentes.

Exemple.

L'exposé sera simplifié si un exemple est traité qui montre la nature du problème :

Soit le code[2] suivant dans l'alphabet $A_0 = \{ a, b, c \}$:

$\mathcal{C}\alpha \longrightarrow a$; $\mathcal{C}\beta = bb$; $\mathcal{C}\gamma = c$; $\mathcal{C}\delta = ab$; $\mathcal{C}\varepsilon = bcb$.

Il n'y a aucune difficulté à réaliser le codage d'une suite de messages élémentaires. Par exemple : $\mathcal{C}(\alpha\,\varepsilon\,\delta\,\varepsilon) = abcbabbcb = m$.

Le décodage par contre n'est pas aussi direct : en procédant de gauche à droite, en effet, le "a" initial est, a priori, aussi bien "α" que la première lettre du mot δ . Cependant cette dernière hypothèse est prouvée fausse, car "c" (troisième lettre du message) devrait être lu "γ" et la séquence restante (babbcb) ne pourrait pas être décodée puisqu'aucun mot ne commence par "ba". Donc selon des notations évidentes :

---

(2) On prouvera plus loin qu'il s'agit bien d'un code.

$$m = \alpha . bcb.abbcb = \alpha\xi abbcb = \alpha\xi m'$$

La même ambiguité se présente à nouveau : si la lettre initiale "a" du message restant m' était α on aurait nécessairement : $m' = \alpha bbcb = \alpha\beta cb = \alpha\beta\gamma b$ et b resterait sans pouvoir être décodé. Donc : $m' = \delta bcb = \delta\xi$ et l'on retrouve bien $\mathcal{C}_m^{-1} = \alpha\xi\delta\xi$. Observons au passage qu'il se peut que l'ambiguité ne soit susceptible d'être levée qu'avant un délai très long :

Le message $q = a(bbc)^n$ (a suivi de n-fois bbc) correspond à $\alpha(\beta\gamma)^n$, et le message $qb = a(bbc)^n b$ correspond à $\delta\xi^n$.

Il est donc nécessaire :

1°) De systématiser les méthodes de décodage (2e section).

2°) De rendre possible leur mécanisation en les simplifiant au maximum par l'emploi de structures moins lourdes que les demi-groupes libres A et P : (3e section).

3°) D'étudier des codes particulièrement maniables (les codes unitaires nets) et de montrer qu'ils forment une "classe admissible" en ce sens qu'ils sont aussi efficaces que n'importe quel autre code du point de vue de la longueur moyenne des mots. (3e et 4e section).

1.- Méthodes de décodage.

Soient A un demi-groupe libre, P un sous-demi-groupe de celui-ci. Le premier problème sera de caractériser algébriquement les P susceptibles de correspondre à un code.

On posera : $P_0 = P - ((P - \emptyset)^2 \cup \emptyset)$ = l'ensemble des $p \in P - \emptyset$ tels que $p'p'' \in P$ et $p', p'' \in P$ impliquent $p' = \emptyset$ ou $p'' = \emptyset$ (3). Comme A est un demi-groupe libre chaque élément possède une "longueur" (le nombre de générateurs figurant dans son expression) et par récurrence, il est clair que tout $p \in P$ est d'au moins une manière décomposable en un produit $p = p_1 p_2 \cdots p_m$ où tous les p_i appartiennent à P_0.

Proposition 1.1.- Une condition nécessaire et suffisante pour que $P \subset A$ corresponde à un code est que la décomposition en produits d'éléments $p_i \in P_0$ soit unique pour tout $p \in P$.

---

(3) Ici, et par la suite, on désignera par $\emptyset$ la suite vide comme distincte de $\emptyset$: l'ensemble vide. On conviendra en outre que la "suite vide" est un élément de tout demi-groupe considéré dont elle est évidemment un élément neutre.

Attachons en effet à tout p_i un $\lambda_i = \mathcal{C}^{-1} p_i$ par une correspondance biunivoque. L'extension de $\mathcal{C}$ au demi-groupe libre Λ engendré par les λ_i est un homomorphisme de Λ sur P puisque si $\lambda = \lambda_1 \lambda_2 \dots \lambda_m$, $\mathcal{C}\lambda = \mathcal{C}\lambda_1 \mathcal{C}\lambda_2 \dots \mathcal{C}\lambda_m = p_1 p_2 \dots p_m$. Cet homomorphisme est un isomorphisme sur si et seulement si $\lambda \neq \lambda'$ entraine $\mathcal{C}\lambda \neq \mathcal{C}\lambda'$ c'est-à-dire si et seulement si $p_1 p_2 \dots p_m \neq p'_1 p'_2 \dots p'_m$ quels que soient p_i, $p'_i \in P_0$.

Considérons, pour P quelconque, un élément $a = a_1 a_2 \dots a_n$ de A où les a_i sont des lettres $(a_i \in A_0)$.

Définition. On appellera indice critique de a tout indice i tel que $a_1 a_2 \dots a_i = {}^*a_i \in P$ et $a_{i+1} a_{i+2} \dots a_n = a_i^* \in P$. Evidemment l'ensemble J_a des indices critiques de a n'est non vide que si $a = {}^*a_i\, a_i^* \in P$.

Proposition 1.2.- Si J est un indice critique de la sous-séquence de a $b = b_{i,i'} = a_{i+1} a_{i+2} \dots a_{i'}$ où $i, i' \in J_a$ il est aussi un indice critique de a.

En effet si ${}^*b_J = a_{i+1} a_{i+2} \dots a_J \in P$ et $b_J^* = a_{J+i} \dots a_{i'} \in P$ on a aussi ${}^*a_i\, {}^*b_J = {}^*a_J \in P$ et $b_J^* a_{i'}^* = a_J^* \in P$.

Proposition 1.3.- Une condition nécessaire et suffisante pour que $P \subset A$ corresponde à un code est que, pour tout $a \in P$ et toute sous-séquence de a, $b = b_{ii'}$, $i, i' \in J_a$ entraine $b \in P$.

La condition est suffisante car si $J_a = \{1 = i_1 < i_2 < \dots i_m = n\}$ d'une part les sous-séquences $b_{i_k i_{k+1}}$ sont indécomposables d'après 1.2 et sont donc des mots,

d'autre part il ne peut exister aucune autre décomposition de a puisque celle-ci impliquerait l'existence d'un indice critique $i' \notin J_a$.

La condition est nécessaire : supposons que $b_{ii'} \notin P$. Il existe dans J_a J et J' (éventuellement : J = 1 et J' = n) tels que :

1°) $J < i < i' < J'$

2°) b_{Ji}, $b_{Ji'}$, $b_{iJ'}$, $b_{JJ'} \in P$

3°) $|J' - J|$

soit minimum parmi les couples J et J' satisfaisant à 1°) et 2°).

$c = b_{JJ'}$ possède deux décompositions en mots au moins : l'une avec :

$$p_1 p_2 \dots p_{m'_1} = b_{Ji'} \quad \text{et} \quad p_{m_1+1} \dots p_m = b_{i'J'}$$

l'autre avec

$$p'_1 p'_2 \dots p_{m'_1} = b_{Ji} \quad \text{et} \quad p_{m'_1+1} \dots p_{m'} = b_{iJ'} \quad (p_i \in P_0)$$

Ces deux décompositions sont distinctes, car si par exemple on avait $p_1 = p'_1 = a_1 a_2 \dots a_{J''}$ le couple (J'', J') satisferait à 1°) et 2°) et on aurait $|J' - J''| < |J' - J|$ en contradiction avec 3°). Une conséquence de la proposition précédente est le

Théorème 1.4.- Une condition nécessaire et suffisante pour que le sous-demi-groupe P du demi-groupe libre A corresponde à un groupe est que :

$$P^{(-1)}P \cap PP^{(-1)} \subset P \quad (4) \tag{1.4}$$

En effet la condition indiquée signifie simplement que : $p , p' , pq , qp' \in P$ implique $q \in P$ et il suffit d'appliquer la proposition en prenant $a = pqp' \in P$.

Si $P \subset A$, où A n'est pas nécessairement un demi-groupe libre, satisfait 1.4, on dira que P est "libérable" dans A . Un autre corollaire intéressant est obtenu en remarquant que $P \subset A$ correspond à un code si et seulement si P est isomorphe à un demi-groupe libre. Soit $\mathcal{L}_A$ l'ensemble des sous-demi-groupe de A qui satisfont à cette propriété et soit δ le "demi-groupe libre vide" $(P_0 = \emptyset)$ que l'on supposera appartenir à $\mathcal{L}_A$:

Proposition 1.5.- $\mathcal{L}_A$ est un treillis.

Comme $A \in \mathcal{L}_A$, il suffit de montrer que $P \in \mathcal{L}_A$ et $P' \in \mathcal{L}_A$ entraine $P \cap P' = \delta$ où $P \cap P' = P'' \in \mathcal{L}_A$. Or si $p_1 p_2 , p_1 q , q p_2 \in P''$, $q \in P$ et $q \in P'$ puisque P et P' satisfont à la relation 1.4 , donc $q \in P''$ et P'' satisfait aussi 1.4.

Définition. Un code sera dit unitaire à gauche si $P^{(-1)}P \subset P$ (unitaire à droite si $PP^{(-1)} \subset P$)

---

(4) Les conventions d'écritures suivantes seront systématiquement employées : (à droite ou à gauche).

$$X^{[-1]}Y = Y \cdot X = \{ \hat{z} : \forall_X x \quad xz \in Y \}$$

$$X^{(-1)}Y = A - (A - Y) \cdot X = \{ \hat{z} : \exists_X x \quad xz \in Y \}$$

si X est un élément, les symboles $x^{[-1]}$ et $x^{(-1)}$ ont le même sens. On observera que les "parenthèses" satisfont à une série d'identité parallèle à celle qui est bien connue pour les "crochets" . Notamment :

$$X^{(-1)}(Y^{(-1)}Z) = (YX)^{(-1)}Z \ ; \ X^{(-1)}(ZY^{(-1)}) = (X^{(-1)}Z)Y^{(-1)} ;$$

$$X^{(-1)}(XY) \supset Y \text{ etc. Toutefois :}$$

$$(X \cup Y)^{(-1)}Z = X^{(-1)}Z \cup Y^{(-1)}Z \quad \text{et} \quad (X \cup Y)^{[-1]}Z = X^{[-1]}Z \cap X^{[-1]}Z \text{ , etc.}$$

On écrira $X^{(-n)}Z$ pour $X^{(-1)}(X^{(-1)}(X^{(-1)} \dots Z))) = (X^n)^{(-1)}Z$.

On a évidemment :

Proposition 1.6.- Tout sous-demi-groupe unitaire P d'un demi-groupe libre correspond à un code.

La définition précédente est exactement celle de P. Dubreil [3]. Elle conduit à un décodage très simple : soit en effet $a = a_1 \dots a_n \in P$ une suite de lettres. Procédant de gauche à droite, soit $i \neq 1$ le plus petit indice tel que ${}^*a_i \in P$. Par hypothèse, puisque P est unitaire, $a_i^* \in P$ et $i \in J_a$. Donc $b_{1i} \in P_0$ et il suffit de recommencer sur la suite $a' = a_{i+1} \dots a_n \in P$ pour aboutir au décodage de a .

Exemple : Soit le code "opposé" de celui donné en exemple dans l'introduction :

$$\mathcal{C}\alpha = a \quad ; \quad \mathcal{C}\beta = bb \quad ; \quad \mathcal{C}\gamma = c \quad ; \quad \mathcal{C}\delta = ba \quad ; \quad \mathcal{C}\varepsilon = bcb$$

On vérifie (Cf. plus bas) que P est unitaire et que le décodage de $m = bcbbabcba$ s'effectue directement sans essais et erreurs :

$$m = \varepsilon babcba = \varepsilon\delta bcba = \varepsilon\delta\varepsilon a = \varepsilon\delta\varepsilon\alpha .$$

Le problème consistant à déterminer pratiquement si un ensemble P_0 de mots correspond ou non à un code avait été résolu par Sardinas et Patterson [12] par des considérations purement combinatoires sans faire appel à la notion de demi-groupe. Leurs résultats peuvent être améliorés de la façon suivante :

supposons que $\emptyset \notin P_0$ et que $P_0 \cap P_0^2 = \emptyset$

Proposition 1.7.- Si l'on pose $P_1 = P_0^{(-1)} P_0$ et par récurrence :

$$P_{n+1} = P_n^{(-1)} P_0 \cup P_0^{(-1)} P_n \text{ alors : } P_n = \bigcup_{n'+n''=n} P_0^{-(n')} P_0^{n''} .$$

Par définition P_1 = l'ensemble des $q \in A$ tels qu'il existe $p, p' \in P_0$ avec $pq = p'$. Supposons établi que pour $n \leq n_0$ on ait montré que :

P_n = l'ensemble des q tels qu'il existe $p_1 , p_2 \dots p_{n'} , p'_1 p'_2 \dots p'_{n''} \in P_0$ $(n' + n'' = n)$ avec

$$p_1 p_2 \dots p_{n'} q = p'_1 p'_2 \dots p'_{n''} \qquad (1)$$

Tout élément q' P_{n+1} est défini par :

$$\text{soit } pq' = q \qquad (2)$$

$$\text{soit } qq' = p \qquad (3)$$

avec $p \in P_0$, $q \in P_n$

Dans le premier cas, multiplions (2) à gauche par $p_1\ p_2 \cdots p_{n'}$, il vient:

$$p_1\ p_2 \cdots p_{n'}\ pq' = p_1\ p_2 \cdots p_{n'}\ q \quad \text{soit :} \quad q'\ P^{-(n+1)} P^{n''} .$$

Dans le second cas, multiplions (1) à droite par q' , il vient :

$$p_1\ p_2 \cdots p_{n'}\ q\ q' = p'_1\ p'_2 \cdots p'_{n''}\ q' \quad \text{soit d'après (3) :} \quad q'\ P^{(-n'')} P^{n'+1}.$$

Proposition 1.8.- Une condition nécessaire et suffisante pour que P_0 soit l'ensemble des mots d'un code est que, pour tout $n \geqslant 1$, $P_n \cap P_0 = \emptyset$.

En effet, on aura ou non affaire à un code selon que pour tout $a \in A$

$$a = p_1\ p_2 \cdots p_m = p'_1\ p'_2 \cdots p'_{m'}\ q \qquad (p_i , p'_i \in P_0 \ ; \ p_1 \neq p'_1)$$

entrainera ou non $q \notin P$.

L'intérêt de cette proposition est que, si le code est borné c'est-à-dire si la longueur de tous ses mots est bornée, les $q \in P_n$ sont des diviseurs à gauche ou à droite des séquences composant P_0 , donc sont bornés eux-mêmes et en nombre fini s'il en est de même de la puissance de l'alphabet. Dans ces conditions il existe un $n < \infty$ tel que

soit $P_n = \emptyset$ et par conséquent $P_{n'} = \emptyset$ pour tout $n' \geqslant n$
soit $P_n = P_{n+r}$ et par conséquent $P_{n+J+kr} = P_{n+J}$ pour tout $J \leqslant r$ et tout k .

On a donc · Dans un code borné

Proposition 1.9.- Une condition nécessaire et suffisante pour qu'il existe une valeur fixe $L < \infty$ telle que la connaissance des $m + L$ premières lettres (à gauche) du message permette quel que soit m de décoder les m premières lettres sans ambiguité est qu'il existe un $n < \infty$ tel que $P_n = \emptyset$.

En particulier une condition nécessaire et suffisante pour que le code soit unitaire à gauche est que $P_1 = \emptyset$.

En effet si un tel n n'existe pas, on peut construire des $a \in A$ de la forme 1.9 pour des valeurs aussi grandes que l'on veut de $m + m'$. Inversement si $P_n = \emptyset$ et si $a = p_1 \cdots p_m = p'_1\ p'_2 \cdots p_{m'} q$ on a $p_i = p'_i$ pour tout $i \leqslant m_0$ avec $m - m_0 + m' - m_0$ n .

Les notions suivantes sont utiles pour caractériser certains types de codes :

Définition. Un code sera dit
net (à droite) si $PA^{(-1)} = A$
absorbant (à droite) si $PP^{(-1)} = A$

La notion de code net est encore exactement celle de P. Dubreil [3]: un code est net à droite si quel que soit la séquence de lettres $a = a_1\, a_2 \cdots a_n \notin P$ il existe au moins une séquence $b = a_{n+1}\, a_{n+2} \cdots a_m$ tel que $ab \in P$.

Le code orthographique du français n'est pas net, car il est impossible d'ajouter des lettres à droite de la séquence "Khtg" par exemple qui la complète en un mot. Par contre le code formé par l'ensemble P des phrases sémantiquement correctes est net, car à toute suite de mots ou de signes on peut au besoin rajouter la clause. "Les x-dernières syllabes que j'ai énoncées étaient un exemple de sentence sémantiquement absurde."

La notion de code absorbant est encore plus forte que la notion de code net : elle signifie que quel que soit $a \notin P$ il existe $q \in P$ (et non pas seulement dans A) tel que $aq \in P$. On démontre :

Proposition 1.10.- Si A est fini, quel que soit $P \subset A$, $A^{[-1]}P \neq \emptyset$ est équivalent à $PP^{(-1)} = A$.

En effet $A^{[-1]}P \ni r$ signifie que $ar \in P$ pour tout $a \in A$.

Réciproquement soit $p_1 \in P$ tel $a_1\, p_1 \in P$ pour au moins un $a_1 \in A - P = A_1$ et par récurrence :

$p_i \in P$ tel que $a_i\, p_i \in P$ pour au moins un $a_i \in A_i = A_{i-1}\, p_{i-1} - P$.

Les ensembles $A'_1 = P\, p_1^{-1}$; $A'_2 = P(p_1\, p_2)^{-1}$ soient strictement croissants ; donc pour un certain $i = J$: $A_J = \emptyset$ et $p_1\, p_2 \cdots p_J \in A^{[-1]}P$.
Il est clair que si un code est unitaire net ou absorbant à droite, il ne s'en suit pas qu'il le soit aussi à gauche. En particulier :

Proposition 1.11.- Une condition nécessaire (mais non suffisante) pour qu'un code soit absorbant à droite est qu'il soit unitaire à gauche et net à droite et qu'il ne soit pas unitaire à droite.

En effet si $PP^{(-1)} = A$, P n'est libérable ($P^{(-1)}P \cap PP^{(-1)} \subset P$) que s'il est unitaire à gauche. D'autre part s'il n'était pas net, il existerait w tel que $wx \notin P$ pour tout x .

Remarque. Les énoncés précédents semblent avoir une partie pratique assez faible puisque, par exemple la recherche des indices critiques d'une séquence s'effectue pratiquement en vérifiant que ${}^*a_i$ et a_i^* sont décodables. Pour aller plus loin il nous faudrait trouver des demi-groupes $\bar{A}$ et $\bar{P}$ de préférence finis et une application φ tels que pour tout $a \in A$, $\varphi a \in \bar{P}$

entraine $a \in P$. C'est ce que nous ferons dans la section suivante. Il se trouve cependant que ces notions dépassent largement le cadre des problèmes de codage et il a paru aussi simple de les traiter en toute généralité pour un A quelconque et un complexe $K \subset A$ qui n'en est pas nécessairement un sous-demi-groupe.

2.- Equivalence syntaxiques.

Soient A un demi-groupe contenant un élément neutre, K une partie quelconque de A .

Définition. On dira que a est syntaxiquement plus fort que b dans A , par rapport à K ($a \gtrcurlyeq b$ (A , K)) si pour tout $x, y \in A$:

(II) $\quad xby \in K$ entraine $xay \in K$

Si $a \gtrcurlyeq b(A , K)$ et $b \gtrcurlyeq (A , K)$, a et b seront "syntaxiquement équivalents" ($a \equiv b(A , K)$).

Proposition 2.1.- $\gtrcurlyeq(A , K)$ est une relation régulière, réflexive, transitive (une relation de préordre (2) compatible avec la structure de demi-groupe). Manifestement $a \gtrcurlyeq a$ (A , K) et $a \gtrcurlyeq b$ (A , K) et $b \gtrcurlyeq c$ (A , K) entrainent $a \gtrcurlyeq c$ (A , K).

D'autre part, si (II) est vraie pour tout $x , y \in A$, (II) est encore vraie pour $x \in Av$ et $y \in vA$. Donc $a \gtrcurlyeq b$ (A , K) entraine $uav \gtrcurlyeq ubv$ (A , K) et en particulier si $a \gtrcurlyeq a'$ (A , K) et $b \gtrcurlyeq b'$ (A , K) on a successivement:

$ab \gtrcurlyeq a'b$ (A , K) et $a'b \gtrcurlyeq a'b'$ (A , K) d'où $aa' \gtrcurlyeq bb'$ (A , K) .

Proposition 2.2.- $\gtrcurlyeq(A , K)$ est la plus forte des relations régulières de préordre pour laquelle K soit supérieurement saturé (5). K est supérieurement saturé pour $\gtrcurlyeq (A , K)$ car $b \in K$ et $a \gtrcurlyeq b$ (A , K) entraine en particulier $a \in K$ en faisant $x = y = \emptyset$ dans (II). Réciproquement, si ρ est régulière et K supérieurement saturé pour ρ, $a \rho b$ implique $(xay) \rho (xby)$ et en particulier $xby \in K$ entraine $xay \in K$ donc $a \rho b$ implique $a \gtrcurlyeq b$ (A,K).

Proposition 2.3.- Si A est un groupe fini $\gtrcurlyeq(A , K)$ se réduit à une relation d'équivalence qui est précisément l'équivalence normale associée au plus grand sous-groupe normal G de A qui soit contenu dans l'un des complexes de K .

(5) ρ étant une relation de préordre ($\geqslant$ réflexive et transitive) on dire que K est supérieurement saturé pour ρ si $a \rho b$ et $b \in K$ entrainent $a \in K$. ρ est régulière si $a \rho b$ entraine $(xay) \rho (xby)$ pour tout x , y . (Cf [3]) .

En effet : si A est un groupe $a \geqslant b\ (A\ ,\ K)$ peut s'écrire : $K\,y^{-1}\,a^{-1}\,b\,y \subset K$ c'est-à-dire encore $a^{-1}\,b \in G$ où G est l'ensemble des c tels que $K\,y^{-1}\,c\,y \subset K$ pour tout y. Mais, d'une part $Kd \subset K$ est équivalent à $Kd = K$, d'autre part $c, c' \in G$ entraine $cc' \in G$. Donc G est le plus grand sous-groupe normal de A tel que $KG = K$ et l'on a : $K = \bigcup_{x \in K} xG$ c'est-à-dire $G \subset k^{-1}K$ pour tout $k \in K$.

Remarque 1.- Si l'on voulait interpréter l'équivalence syntaxique dans le code des "phrases françaises grammaticalement correctes" on trouverait par exemple que "postulat" $\not\equiv$ "axiome" $\equiv$ "hippopotame" car on peut aussi bien dire "l'axiome d'Euclide est à la base de la géométrie élémentaire" que "l'hippopotame d'Euclide ...etc" alors que la phrase "l'postulat d'Euclide ...etc" est incorrecte.

Remarque 2.- Dans ce même cadre linguistique il serait possible et utile de généraliser la définition précédente

Définition.- Le n-tuple $a = (a_1\ ,\ \ldots\ ,\ a_n)$ d'éléments de A est "syntaxiquement plus fort que le n-tuple $b = (b_1\ ,\ b_2 \ldots\ b_n)$ dans A par rapport à K et au n+1-tuple $(g_0\ ,\ g_1 \ldots\ g_n) = g$" si pour tout 2n-tuple $(x_1\ ,\ y_1\ ,\ x_2\ ,\ y_2 \ldots\ x_n\ ,\ y_n) = (x\ ,\ y)$ $g_0\,x_1 b_1 y_1\,g_1\,x_2\,b_2 \ldots\ g_{n-1}\,x_n\,b_n\,y_n\,g_n = g_{(x,y)}(b) \in K$ entraine $g_{(x,y)}(a) \in K$. $(a \geqslant b (A\ ,\ K\ ;\ g))$

Les propriétés 2.1 et 2.2 subsistent et naturellement $a_i \equiv b_i\ (A\ ,\ K)$ pour tout i entraine $a \geqslant b\ (A\ ,\ K\ ;\ g)$ la réciproque n'étant pas forcément vraie.

Nous ne ferons pas usage ici de cette notion générale.

Définition. Soit φ_K l'homomorphisme attaché à $\equiv (A\ ,\ K)$. On posera $\bar{A} = \varphi_K A$; $\bar{K} = \varphi_K K$; si $K = P \supset P^2$ on dira que A et P sont les demi-groupes syntaxiques fondamentaux (GSF) de $(A \supset P)$.

Si $\varphi_K = 1$ (φ_K réduit à l'application identique) on dira que $K\ (= \bar{K})$ est syntaxiquement simple dans $A\ (= \bar{A})$.

Proposition 2.4.- $a \geqslant b\ (A\ ,\ K)$ est équivalent à $\varphi_K a \geqslant \varphi_K b\ (\bar{A}\ ,\ \bar{K})$. Donc $\bar{K}$ est syntaxiquement simple dans $\bar{A}$.

D'une part $xby \in K$ entraine $\varphi_K x\ \varphi_K b\ \varphi_K y \in \varphi_K K$.

D'autre part : $\varphi_K x\ \varphi_K b\ \varphi_K y = \varphi_K(xby) \in \varphi_K K$ entraine $xby \in K$ puisque K est saturé. Donc $\varphi_K a \equiv \varphi_K b$ $(\varphi_K A\ ;\ \varphi_K K)$ est équivalent à $a \equiv b\ (A\ ,\ K)$ donc encore à $\varphi_K a = \varphi_K b$ dans $\varphi_K A$.

Proposition 2.5.- Si $K \supset K^2 = Q$ est un sous-demi-groupe, chacune des propriétés suivantes est simultanément vraie ou non pour $Q \subset A$ et pour $\Psi_Q\, Q \subset \Psi_Q\, A$: Q est libérable, unitaire, net, absorbant.

Soient $B \supset R$ deux demi-groupes quelconques et Ψ un homomorphisme. Une condition nécessaire et suffisante pour que quel que soit $x \in B$, $\Psi(Qx^{-1}) = \Psi Q(\Psi x)^{-1}$, est que Q soit saturé pour l'équivalence d'homomorphisme attachée à Ψ. Car :

$a \in Qx^{-1} \rightleftarrows ax \in Q$ qui entraine : $\Psi a\, \Psi x \in \Psi Q$ soit $\Psi a \in \Psi Q(\Psi x)^{-1}$; et réciproquement : $\Psi a\, \Psi x = \Psi q \in \Psi Q$ entraine $ax \in q$ est la définition même de la saturation.

Or les propriétés indiquées ne font appel qu'à l'opération $Q X^{(-1)} = \bigcup_{x \in X} Q x^{-1}$.

Proposition 2.6.- Soient $A \supset K$ et Ψ un homomorphisme. Une condition nécessaire et suffisante pour que $\Psi_K = \Psi_{\Psi_K} \circ \Psi$ au sens de la composition ($\circ$) des homomorphismes est que K soit saturé pour l'équivalence σ attachée à Ψ (ce que l'on peut noter $\Psi^{-1} \Psi K = K$).

La condition est nécessaire : si $a \in A - K$, $k \in K$ et $\Psi k = \Psi a = k' \in \Psi K$ alors $(\Psi_{\Psi_K} \circ \Psi)\, k = (\Psi_{\Psi_K} \circ \Psi) a$ et $\Psi_K\, a \neq \Psi_K\, k$.

La condition est suffisante : si σ est moins forte que $\equiv (A, K)$ $a \equiv b\ (A, K)$ entraine $\Psi a \equiv \Psi b\ (\Psi A, \Psi K)$.

Définition. Si A est un demi-groupe libre et K un complexe de A nous dirons que $A \supset K$ est une extension libre de $A' \supset K'$ s'il existe un homomorphisme Ψ tel que (1) $A' = \Psi A$, (2) $K' = \Psi K$, (3) $\Psi^{-1} \Psi K = K$.

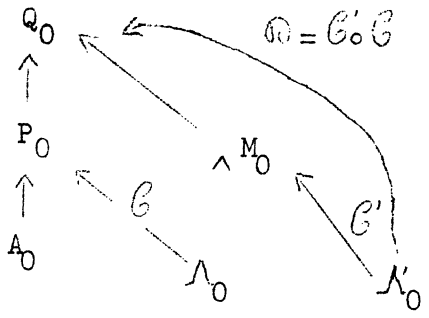
Proposition 2.7.- L'ensemble des couples $(A' \supset P')$ tels que leurs GSF soient isomorphes au couple syntaxiquement simple $(\bar{A} \supset \bar{P})$ est identique à l'ensemble des couples $(\Psi A \supset \Psi P)$ où $(A \supset P)$ est une extension libre de $\bar{A} \supset \bar{P}$ et où $\Psi^{-1} \Psi P = P$.

En effet d'après 2.6 on a : d'une part $\Psi_P\, A = \bar{A}$ et $\Psi_P P = \bar{P}$ d'autre part $\Psi_P = \Psi_{\Psi_P} \circ \Psi$.

Remarque 1. 2.7 montre que l'on peut construire pour toute puissance α de l'alphabet assez grande au moins un code admettant comme GSF un couple syntaxiquement simple $\bar{A} \supset \bar{P}$ où $\bar{P}$ est libérable dans A donné :

Si $\bar{g}_1 \ldots \bar{g}_m$ sont des générateurs de $\bar{A}$ et $\bar{h}_J$ d'autres éléments de $\bar{A}$, il suffit de faire correspondre les lettres $a_{1_1} a_{1_2} \ldots a_{1_i}$ à $\bar{g}_1$ les lettres $a_{2_1} a_{2_2} \ldots a_{2_i}$, à $\bar{g}_2 \ldots$, $a_{m_1} a_{m_2} \ldots a_{m_i}$ à $\bar{g}_m$ et $a_{J_1} \ldots a_{J_i}$ à $\bar{h}_J$ en posant $\Psi a_{\ell,k} = \bar{a}_\ell \in \bar{A}$ et de définir P par $\Psi^{-1} \bar{P}$ ce qui assure que $\Psi^{-1}\Psi P = P$. On observera qu'en général pour un tel choix quelconque de générateurs l'ensemble des mots $P_0 = P - ((P - \phi)^2 \cup \phi)$ n'est pas borné même si $\bar{A}$ est fini.

Remarque 2. Un problème important est celui du surcodage : c'est-à-dire celui de l'utilisation des mots d'un code primaire comme "lettres" d'un alphabet secondaire ainsi qu'il est indiqué dans le schéma ci-contre, qui est illustré par les identifications suivantes :



A_0 : l'"alphabet"morse formé des trois "lettres primaires" : point, trait intervalle

Λ_0 : l'ensemble des signes typographiques mis en correspondance par $\mathcal{G}$ avec :

P_0 : l'ensemble des séquences de points traits et intervalles qui représentent des signes typographiques dans le code Morse. P_0 est à la fois un ensemble de mots primaires et un alphabet secondaire.

Λ'_0 : l'ensemble des mots du français écrit mis en correspondance

1°) par $\mathcal{G}'$ avec M_0 : l'ensemble des séquences de signes typographiques qui représentent des mots du français écrit (M_0 est un ensemble de mots primaires pour $\mathcal{G}'$).

2°) par $\mathcal{G} = \mathcal{G}' \circ \mathcal{G}$ avec Q_0 : l'ensemble des suites de points, traits intervalles, qui représentent des mots du français écrit (Q_0 est un ensemble de mots secondaires par rapport à P_0 et primaires par rapport à $\mathcal{G}$).

Il serait utile de caractériser au moins partiellement les GSF de $(A \supset Q)$ au moyen de ceux des "facteurs" $(A \supset P)$ et $(\Lambda \supset M)$. Le problème sera étudié dans un autre travail et les énoncés suivants très simples sont avec 2.6 et 2.7 à la base des démonstrations.

Proposition 2.8.- Si K et K' sont deux complexes de A $a \geqslant b$ (A, K) et $a \geqslant b$ (A, K') entrainent $a \geqslant b$ $(A, K \cap K')$

Immédiat : car $xby \in K \cap K'$ entraine $xby \in K$ et $xby \in K'$ qui entrainent $xay \in K$ et $xay \in K'$ donc $xay \in K \cap K'$.

On observera que même si K et K' étaient des demi-groupes P et P' il serait possible que $P \supset P'$ sans que pour autant $\equiv (A , P)$ soit plus forte que $\equiv (A , P')$ au contraire de ce qui se produit si A est un groupe. En effet, l'intersection de $\equiv (A , P)$ et de $\equiv (A , P')$ si $(P' \subset P)$ est identique à l'intersection de $\equiv (A , P')$ et $\equiv (A , P - P')$ qui ne sont pas en général identiques.

Proposition 2.9.- Si $A' \subset A$ et $K' = A' \cap K$ la restriction de $\geqslant (A , K)$ à A' est plus forte que $\geqslant (A' , K')$. En effet, $xby \in K'$ implique $xay \in K$ par hypothèse et, si $a , b, x , y \in A'$, on a donc $xay \in A \cap K = K'$.

3.- Préfixes.

Définition. On appellera "préfixe à droite" ($\Pi_i^* \in \Pi^*$) (respectivement préfixe à gauche : ${}^*\Pi_J \in {}^*\Pi$) les classes d'équivalence de A pour la relation $\sim^* (A , P)$ (respectivement ${}^*\sim (A , P)$) définie par : $a \sim^* b \ (A , P)$ si et seulement si $a^{-1}P = b^{-1}P$ (respectivement : $a \, {}^*\sim b \ (A , P)$ si et seulement si $Pa^{-1} = Pb^{-1}$) .

Il est classique [3] que :

Proposition 3.1.- La relation $\sim^* (AP)$ est régulière à droite (respectivement : ${}^*\sim (AP)$ est régulière à gauche) et $\equiv (A , P)$ est plus forte que l'intersection de $\sim^* (AP)$ et ${}^*\sim (A , P)$.

En effet $a \equiv b \ (A , P)$ s'écrit aussi bien :

$(ax)^{-1}P = (bx)^{-1}P$ pour tout x que $P(xa)^{-1} = P(xb)^{-1}$ pour tout x .

Il en résulte :

Proposition 3.2.- La représentation A^* des éléments $x \in A$ comme application de l'ensemble Π^* des préfixes à droite (à gauche) dans lui-même est une représentation isomorphe de $\bar{A}$.

En effet puisque $\sim^*$ est régulière à droite $a,b \in \Pi_i^*$ entraine $ax , bx \in \Pi_J^* = \Pi_i^* x$ pour un certain J quels que soient a , b et x , d'autre part $\Pi_i^* x = \Pi_i^* y$ pour tout $\Pi_i^* \in \Pi$ entraine $x \equiv y \ (A , P)$ et réciproquement.

On déduit de cette remarque la :

Proposition 3.3.- $A \supset P$ étant deux demi-groupes quelconques si $\varphi_P P = \bar{P}$ est un groupe $A = \varphi_P A$ s'il est fini, est la réunion d'un groupe et éventuellement d'un zéro.[6]

En effet la représentation 3.2 étant isomorphe $\bar{P}$ est un groupe si et seulement si toutes les applications correspondantes de $\widehat{\Pi}^*$ dans lui-même sont des permutations. Donc si $a \in A$ est tel que $xay \in P$ pour au moins un couple $x, y \in A$, a est aussi une permutation. Sinon $ab \equiv ba \equiv a\ (A, P)$ pour tout b et $\varphi_P a = 0$.

Remarque 1. Il est possible et souvent commode de développer la théorie de l'équivalence syntaxique de la façon suivante : soit $a \in A$. L'ensemble $a^{-1}P$ est celui des séquences $b \in A$ telles que $ab \in P$ et l'ensemble $\widehat{\Pi}^*_a = P\,(a^{-1}P)^{[-1]}$ celui des séquences c telles que $cb \in P$ pour tout $b \in a\,P$.

Evidemment $a \in \widehat{\Pi}^*_a$ et $\widehat{\Pi}^{[-1]}_a P = a^{-1}P$. On peut vérifier que $\widehat{\Pi}^*_a$ ainsi défini est précisément le préfixe à droite qui contient a . De la même manière on trouverait ${}^*\overline{\Pi}_a = (Pa^{-1})^{[-1]}P$, et la relation ρ entre préfixes à droite et à gauche définie par : $\overline{\Pi}^*_i \ \rho\ {}^*\overline{\Pi}_J$ si et seulement si $a \in \overline{\Pi}^*_i$ et $b \in {}^*\overline{\Pi}_J$ entraine $ab \in P$, permet d'établir une correspondance de Galois [2] entre $\widehat{\Pi}^*$ et ${}^*\widehat{\Pi}$. Le treillis complet associé pourrait être appelé le "treillis fondamental" du code $(A \supset P)$ et ses propriétés, une fois encore, ne dépendent que des GSF $(\bar{A} \supset \bar{P})$.

Remarque 2. Il est utile de distinguer certains préfixes remarquables :

$\overline{\Pi}^*_0$: le résidu de P dans A au sens de P. Dubreil [3] : l'ensemble des a tels que ax n'appartienne à P pour aucun x .

$\overline{\Pi}^*_\infty$: le préfixe "absorbant" : l'ensemble des a tels que $ax \in P$ pour tout x .

$\overline{\Pi}^*_1$: le préfixe unité tel que $a \in \overline{\Pi}_1$ entraine $a \in P$ et $ax \in P$ seulement si $x \in P$.

$\overline{\Pi}^*_{1_{p_i}}$: les préfixes qui sont des sous-classes de P $(a \in \overline{\Pi}_{p_i}$ entraine $a \in P)$.

L'existence ou la non-existence de tels préfixes non vides est trivialement liée au fait que le code est net, absorbant, unitaire etc.

---

[6] un zéro est un élément 0 tel que $x0 = 0x = 0$ pour tout x .

Exemple. Soit le code suivant (unitaire à droite) :

$$G\alpha = a \quad ; \quad G\beta = ab \quad ; \quad G\gamma = bb$$
$$\Pi_0^* \ni b \quad ; \quad \Pi_\infty^* \ni a \quad ; \quad \Pi_1^* \ni bb \quad ; \quad \Pi_{p\rho} \ni ab \ .$$

Proposition 3.4.- Si l'alphabet d'un code est fini, et si la longueur de ses mots est bornée par $L < \infty$ alors son GSF $\bar{A}$ est fini.

Puisque $\bar{A}$ possède une représentation comme groupe d'application de Π (7) dans lui-même, il suffit de montrer que Π est fini. Or si $ax \in P$, ou bien la longueur de x est $\leq L$, ou bien il existe x' diviseurs à gauche de x de longueur inférieure à L tels que $ax' \in P$. Donc $a^{-1}P = b^{-}P$ si et seulement si $a^{-1}P \cap X_L = b^{-}P \cap X_L$ où X_L est l'ensemble fini des séquences de lettres de longueur $\leq L$.

Proposition 3.5.- Dans tout code unitaire l'ensemble des préfixes différents du résidu est une image homomorphe de l'ensemble des diviseurs (à gauche) des mots.

Montrons d'abord que $p \in P$ entraine $(pa)^{-1}P = a^{-1}P$ pour tout a. Or $pax \in P$ entraine $ax \in P$ par hypothèse puisque P est unitaire.

Soit donc $a \in \Pi_i \neq \Pi_0$; il existe y tel que $ay \in P$ donc a est diviseur à gauche d'une suite de mots donc, ou bien a est lui-même diviseur à gauche d'un mot, ou bien il existe $p \in P$, tel que $a = pa'$ et $a' \in \Pi_i$, d'où le résultat par récurrence. On en déduit la remarque utile suivante :

Proposition 3.6.- Il existe une correspondance biunivoque entre les codes unitaires et les structures d'arbres enracinés (8), les codes nets, correspondant

---

(7) Pour simplifier et puisque dans la pratique un ordre temporel est toujours prescrit pour la suite des lettres, nous conviendrons dorénavant sauf mention expresse du contraire, que préfixe signifie, préfixe à droite ; unitaire : unitaire à gauche ; net : net à droite. Cette convention est la plus simple quand l'ordre temporel est identifié avec l'ordre gauche $\longrightarrow$ droite.

(8) Soit U un ensemble ordonné par ρ. Pour tout u soit $\rho[u]$ = l'ensemble des $u' \in U$ tels que $u' \rho u$. U est un arbre enraciné si 1°) $\bigcap_{u \in U} \rho[u] = u_1 \neq \emptyset$ (u_1 est la racine) 2°) pour tout $\rho[u]$ est un ensemble totalement ordonné. Si $u \in \rho[v]$ entraine $v = u$ u est une extrémité, sinon u est un noeud. Si pour tout u qui n'est pas une extrémité il existe un nombre fixe $K < \infty$ de v tels que 1°) $u \rho v$ et 2°) $u \rho w \rho v$ entraine $w = u$ ou $w = v$, l'arbre est dit complet.

aux arbres complets.

Il suffit d'identifier la racine à la séquence vide, les noeuds aux diviseurs à gauche des mots, et les extrémités aux mots eux-mêmes. On observera que deux diviseurs appartiennent au même préfixe si et seulement si les sous-arbres correspondant dont ils sont les racines sont identiques.

Exemple : soit le code suivant :

$\mathcal{C}\alpha$ = aaa ; $\mathcal{C}\beta$ = aab ; $\mathcal{C}\gamma$ = ab ; $\mathcal{C}\delta$ = baa ; $\mathcal{C}\varepsilon$ = bab ; $\mathcal{C}\xi$ = bb

il lui correspond l'arbre ci-contre (figure 1). Les diviseurs "a" et "b" appartiennent au même préfixe.

figure 1

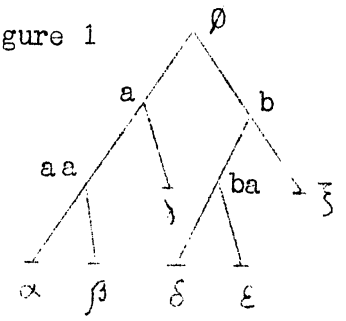
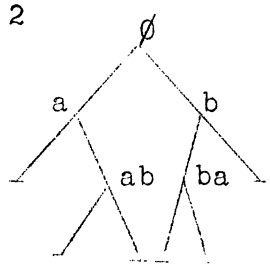


figure 2



On prendra garde que des arbres <u>équivalents à des permutations près de noeuds</u> correspondent en général à des codes dont les GSF sont <u>différents</u>. (l'arbre de la figure 2 par exemple est équivalent par permutation à celui de la figure 1 mais son code à un autre GSF) .

Une conséquence immédiate de la représentation des codes unitaires par des arbres enracinés est :

Proposition 3.7.- Dans un code unitaire une condition nécessaire et suffisante pour que tous les mots aient une longueur bornée est que, si a et ax appartiennent au même préfixe, il existe au moins un diviseur à gauche x' de x tel que $ax' \in P$.

En effet les sous-arbres a et ax ne sont jamais identiques, si $ax' \notin P$, pour tous les diviseurs à gauches de x'.

<u>Proposition 3.8</u>.- Une condition nécessaire et suffisante pour qu'un code unitaire dont la longueur des mots est bornée par $L < \infty$ possède une suite <u>non vide</u> de lettres ℓ telle que $\varphi_P \ell$ soit un élément neutre bilatère de $\varphi_P A$ est qu'il existe une lettre $a \in A_0$ telle que les séquences a , a^2 = aa . a^3 = aaa , ... , $\mathcal{C} \equiv a^m \in P$ forment un système de représentants de ses préfixes, à l'exception éventuelle du résidu π_0 .

Par hypothèse ℓ est une application de Π sur lui-même, si donc ℓ contient la lettre a dans son expression, a est une permutation des préfixes (Cf. 3.3) et puisque la longueur des mots est bornée il existe un m minimum fini tel que $a^m \in P$ et $a^m \equiv \varepsilon$ (A , P) . On vient de voir que $a^{m'} \sim a^{m''}$ $m', m'' \leq m$ entraine $m' = m''$. Soit $b \in \Pi'$. Pour au moins un $n' < \infty$, $ba^{n'} \in P$ mais aussi pour un $m' = m - n'$ (à un multiple de m près) $a^{m'}a^{n'} \in P$ donc Π' est identique au préfixe contenant $a^{m'}$ puisque la multiplication à droite par $a^{n'}$ est une permutation. On en déduit :

Proposition 3.9.- Si la longueur des mots d'un code est bornée et si la puissance de son alphabet est finie, une condition nécessaire et suffisante pour que son GSF $\bar{A}$ soit un groupe est que $P_0 = X\ell$ (= l'ensemble de toutes les séquences de longueur ℓ : code uniforme de longueur ℓ). Dans ce cas $\bar{A}$ est le groupe cyclique d'ordre ℓ et P se réduit à son élément neutre.

De 3.7, il résulte que toutes les lettres de l'alphabet doivent correspondre à des permutations circulaires de même ordre ℓ sur les préfixes . Montrons que $a^n b \sim a^{n+1}$ quels que soient les lettres a et b , (il sera convenu que les indices seront des entiers positifs réduits modulo ℓ .

Supposons que, pour un n' et un m', $m' \leq n'$, on ait $a^{n'}b \sim a^{m'}$ et considérons les séquences non bornées $c_x = a^{n'}(ba^{m'-n'})^x$ (n' fois suivis de x fois la sous-séquence b suivi de $m' - n'$ a).

On vérifie facilement qu'aucun de leurs diviseurs à gauche n'appartient à P (ils sont tous $\sim$ à un $a^{n''}$ où $n'' \leq n'$). Donc $c_x a^{\ell - n'} \in P$ est un mot contrairement à l'hypothèse selon laquelle la longueur des mots est bornée. Donc pour tout n' $a^{n'}b \sim a^{m'}$ où 1°) m' parcourt avec n' l'ensemble de tous les nombres $\leq \ell$ (car b est une permutation !) 2°) $m' > n'$ si $n' \neq \ell$. Ceci n'est possible que si $m' = n' + 1$ (modulo ℓ) ou encore que si $a \equiv b$ (A , P) puisque $\Pi_i a \sim \Pi_i b$ pour tout préfixe. Donc $\bar{A}$ est un groupe fini à un seul générateur, etc.

Remarque.

Le résultat précédent pose le problème de savoir s'il existe des couples syntaxiquement simples $\bar{A} \supset \bar{P}$ où $\bar{P}$ serait unitaire et net à droite et à gauche sans que $\bar{A}$ soit un groupe et tels que pour au moins un choix de générateurs la longueur des mots soit bornée. La proposition 3.10 est une réponse

très partielle à cette question. Nous montrons d'abord

Proposition 3.9.- La condition nécessaire et suffisante pour que l'élément u du demi-groupe libre A satisfasse pour au moins un couple v , $w \neq \emptyset$ à : $a = uv = wu$ où la longueur ℓ de a est inférieure ou égale au double de la longueur ℓ' de u est que u soit de la forme $x(yx)^n$.

Raisonnons par récurrence : les longueurs de v et w étant strictement inférieures à $\ell/2$ on a :

$$a = wrv \quad \text{pour un certain } r \text{ et } \quad u = wr = rv .$$

Si la longueur de r est plus petite ou égale $\ell'/2$ on a encore : $u = rsr$ avec $w = rs$ et $v = sr$. Sinon on est ramené au problème initial avec u au lieu de a et r au lieu de u . Comme les longueurs des séquences ne peuvent que décroitre strictement on a bien le résultat.

Proposition 3.10.- Pour un alphabet de $K < \infty$ lettres il existe au moins un code du type qui vient d'être décrit et dont le nombre de mots est égal à $1 + K^{\ell} - 2K^{\ell''} + K^{2\ell''}$ pour tout $\ell \geqslant 3$ et $\ell'' < \ell/2$.

Nous considérons d'abord le code uniforme $\mathcal{A}$ dont P_0 est identique à l'ensemble X_ℓ de toutes les séquences de ℓ lettres. Il satisfait aux conditions voulues sauf que son GSF est un groupe.

Soit u une séquence fixe de longueur $\ell' < \ell$ et soient les ensembles suivants :

P'_0 : les mots (de $X_\ell = P_0$) dont u est un diviseur à gauche ($P'_0 = u(u^{-1}P_0)$)

P''_0 : les mots dont u est un diviseur à droite mais non à gauche
($P''_0 = (P_0 u^{-1})u - P'_0 \cap (P_0 u^{-1})u$)

X : l'ensemble des séquences de longueur $\ell'' = \ell - \ell'$

Q'_0 : l'ensemble $P''_0 X_{\ell''}$ des séquences de la forme $q = pv$ avec $p \in P''_0$ et $v \in X_{\ell''}$.

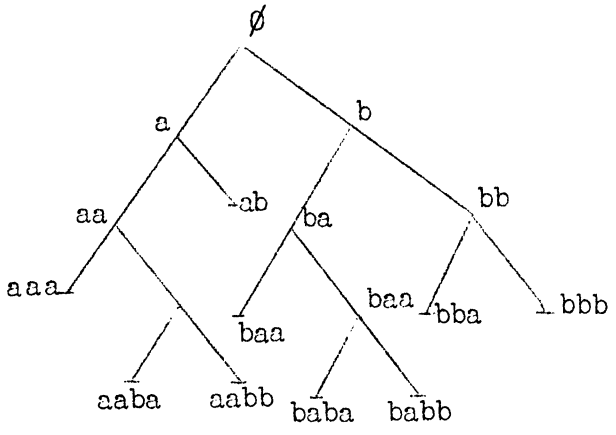
Considérons le code $\mathcal{B}$ dont l'ensemble Q_0 des mots est la réunion de u , Q'_0 et de l'ensemble $Q''_0 = P_0 - (P'_0 \cup P''_0)$. Par construction $\mathcal{B}$ est unitaire à gauche et net à droite. D'autre part si q , $q' \in Q_0$ q ne peut évidemment pas être un diviseur à droite de q' dans les cas suivants :

1°) $q' = u$; 2°) $q \in Q'_0$; 3°) $q \in Q''_0$ et $q' \in Q''_0$; 4°) $q = u$ et $q' \in Q''_0$

Il reste à discuter 5°) le cas de $q \in Q_0''$ et $q' \in Q_0'$ mais les mots de Q_0'' sont de la forme $xuv = xp$ avec $v \in X_{\ell''}$ et $p \in P_0'$ et $p \in P_0'$ ne peut admettre aucun diviseur propre à droite de longueur ℓ . 6°) $\mathcal{B}$ sera donc unitaire à droite si et seulement si $Q_0' u^{-1} = \emptyset$ ou encore si $uvu^{-1} = \emptyset$ pour tout $v \in X_{\ell''}$. Ceci n'est possible comme on l'a vu en 3.9 que si $\ell' > \ell/2$ et si u n'est pas de la forme exceptionnelle $x(yx)^n$. Le calcul du nombre des mots n'offre aucune difficulté.

Exemple.

Le cas le plus simple ($K = 2$; $\ell = 3$) est décrit par l'arbre suivant. Il contient 9 mots et son GSF (privé de l'élément neutre bilatère) a 24 éléments et ne possède pas d'idéaux propres. Cette dernière particularité n'est pas une nécessité pour les GSF des codes de ce type.



(u = ab)

4.- Méthodes de dénombrement.

On supposera désormais toujours que l'alphabet a une puissance $K < \infty$. On dira, pour abréger, qu'un code est borné si la longueur de ses mots est plus petite que $L < \infty$.

Nous commencerons par compléter et systématiser divers résultats plus ou moins explicitement connus et utilisés par les auteurs qui ont étudié ces questions et notamment B. Mandelbrot [9] .

Notations $g(t) = 1 - \sum_{i=1}^{\infty} n_i t^i$: (avec n_i = nombre des mots de longueur i)

La fonction de structure du code $G(t) = 1 + \sum_{i=1}^{\infty} N_i t^i$ (avec N_i = nombre des éléments de P de longueur i) : la fonction génératrice du code.

$H(t) = |Dt - 1|$ la fonction caractéristique du code avec D : la somme des matrices correspondant aux lettres de A_0 dans la représentation régulière de $\bar{A}$.

$h_d(t)$ et $h_g(t)$: les déterminants correspondants dans les représentations de $\bar{A}$ comme demi-groupe d'application dans eux-mêmes de Π^* et de $^*\Pi$.

Proposition 4.1.- $g(t)$ est un diviseur commun et $H(t)$ un multiple commun de $h_g(t)$ et $h_d(t)$. On a $G(t) = \frac{1}{g(t)}$. Introduisons encore $N_i(\bar{a})$: nombre de séquences a de longueur i telles que $\varphi_P a = \bar{a} \in P$. Les $N_i(\bar{a})$ sont liés entre eux par un système d'équations aux différences finies :

$$N_{i+1}(\bar{a}) = \sum_{\bar{b} \in \bar{A}} \delta_{\bar{a},\bar{b}} \, N_i(\bar{a})$$

où $\delta_{\bar{a},\bar{b}} = 0$ ou $1, 2, \ldots, K$ selon qu'il existe $0, 1, 2, \ldots, K$ lettres a_i dans A_0 telles que $\varphi(a\, a_i) = \bar{b} \in \bar{A}$. La matrice des $\delta_{\bar{a},\bar{b}}$ est précisément D et si les ρ_J sont les racines du déterminant $|Dt - 1|$ on sait que $N_i(\bar{a}) = \sum_J \beta_{\bar{a},J} \, \rho_J^{-i}$ pour un certain système de constantes $\beta_{\bar{a},J}$. D'autre part, les $N_i(\Pi^*)$ ou les N_i qui sont obtenus comme sommes (par rapport à $\bar{a}$) de certains $N_i(\bar{a})$ peuvent être calculés directement à partir de représentation de $\bar{A}$ sur Π^* ou sur $^*\Pi$ ce qui établit les relations de divisibilité indiquées. Enfin on a directement $N = \sum_i N_{\ell-i} \, n_i$ ce qui donne $G(t) = 1/g(t)$ par un raisonnement classique. (Cf. 6)

Proposition 4.2.- La fonction de structure d'un code borné unitaire (à gauche) et net (à droite) (code UND) est de la forme :

$$g(t) = (1 - Kt)\, \bar{g}(t) \quad \text{où} : \bar{g}(t) = 1 + \sum \bar{n}_i \, t^i \quad \text{avec} \quad 0 \leq \bar{n}_{i+1} \leq K n_i .$$

K^{-1} en est donc la plus petite racine positive. Réciproquement tout polynome $g(t)$ de la forme précédente peut être considéré comme la fonction de structure d'au moins un code UND.

Soit $\bar{n}_\ell$ le nombre des séquences de longueur ℓ qui sont diviseur propre à gauche des mots d'un code unitaire (= qui correspondent aux "noeuds" de l'arbre de codage). On a : $\bar{n}_0 = 1$ et $\bar{n}_{\ell+1} + n_{\ell+1} \leq K\bar{n}_\ell$ le signe $\leq$ n'étant partout remplacé par le signe = que si l'arbre est complet, c'est-à-dire si le code est net. Multipliant ces égalités par t^{+1} et sommant, on obtient bien : $g(t) = (1 - Kt)\, \bar{g}(t)$.

Réciproquement, un polynome de la forme $g(t) = 1 - \sum^{\ell} n_i t^i$ $(n_i \geqslant 0)$ admet au moins un diviseur de la forme $(1 - K't)$ et l'on a : $g(t) = (1 - K't)(1 + \bar{n}_1 t + \bar{n}_2 t^2 \ldots)$ avec $\bar{n}_{\ell+1} \leq K'\bar{n}_i$ pour tout i. Donc si $g(t)$ est un polynome $0 \leq \bar{n}_{i+1} \leq K'\bar{n}_i$.

D'autre part K' ne saurait être $> K$ car sinon pour ℓ assez grand $N_\ell = \sum \beta_J \rho_J^{-\ell} \geqslant \beta_1 K'^{\ell}$ serait plus grand que K^ℓ, ce qui est impossible puisqu'il s'agit d'un code et que ceci signifierait que à au moins l'une des K^ℓ suites de ℓ lettres correspondent plusieurs séquences de mots. Enfin si $\bar{n}_i$ satisfait à $0 \leq n_{i+1} \leq K'\bar{n}_i$ pour tout i il est facile de construire un arbre dont $\bar{n}_i$ et n_i sont respectivement les nombres de noeuds et d'extrémités à distance i de la racine [9]. On observera :

Proposition 4.3.- Dans un code UND le coefficient β_1 de K^i dans l'expression $N_i = \sum \beta_J \rho_J^{-i}$ est précisément l'inverse de la longueur moyenne $\bar{L}$ des mots quand la fréquence des mots de longueur ℓ est proportionnelle à $K^{-\ell}$.

En effet : $G(t) = \dfrac{1}{g(t)} = \dfrac{(\bar{g}(K^{-1}))^{-1}}{1 - Kt} + \dfrac{B}{\bar{g}(t)}$ où B est une certaine constante.

D'autre part, on a , par hypothèse :

$L = \sum n_i \, i \, K^{-i}$ c'est-à-dire que $-L$ est la valeur de $\dfrac{\partial g(t)}{\partial t} K^{-1}$ pour $t = K^{-1}$ soit encore précisément $\bar{g}(K^{-1})$.

Remarque.

Attachons à tout $a_i \in A_0$ une probabilité $\omega_i \geqslant 0$ $(\sum \omega_i = 1)$ et considérons au lieu de D la matrice $\sum \omega_i D_i$ en appelant D_i la matrice associée à a_i dans la représentation régulière de $\bar{A}$. Les mêmes considérations sont encore valables et les fonctions $N_i(\bar{a})$ deviennent des probabilités dans un processus stochastique (sur A) ou les lettres successives sont tirées au sort indépendamment et avec les probabilités constantes ω_i. Le cas traité ici correspond à un changement d'échelle sur t (qui devrait être remplacé par $t_1 = tK^{-1}$) à celui où tous les ω_i sont égaux et on vérifiera que les propriétés 4.1 , 4.2 , 4.3 sont des théorèmes bien connus pour les chaines de Markoff. Une théorie peut aussi être développée dans laquelle les matrices D_i n'ont pas nécessairement leurs éléments égaux à zéro ou à un, mais il peut alors

---

[9] Ceci complète une démonstration de B. Mandelbrot [8] qui laissait ouverte la possibilité que β_1 (le coefficient du terme correspondant à la plus petite racine) soit plus grand pour certains codes non unitaires que pour tout code unitaire.

se produire que la relation d'équivalence $\equiv (A, P)$ doive être remplacée par une relation plus faible pour tenir compte de la dépendance en chaine des lettres.

Nous n'avons besoin ici que des méthodes énumératives pour établir les deux résultats suivants qui relèvent strictement de la théorie algébrique des demi-groupes et qu'il semblerait difficile d'obtenir autrement.

Proposition 4.4.- Tout code borné d'alphabet fini net à droite est unitaire à gauche.

Soit $L < \infty$ la longueur maximum des mots du code. Puisque le code est net, il correspond, à chacune des K^ℓ séquences de longueurs ℓ, au moins une séquence de longueur $\leq \ell + L$ qui appartient à P est dont elle est un diviseur à gauche. Donc : $N_\ell + N_{\ell+1} + \ldots N_{\ell+L} \geq K^\ell$ pour tout ℓ assez grand. Donc 4.1, au moins une des racines de $g(t)$ est $\leq K^{-1}$ et comme il s'agit d'un code cette racine de module minimum est précisément K^{-1}. Donc (4.2), $g(t)$ est identique à la fonction de structure d'un certain code UND.

Considérons maintenant P'_0, le sous-ensemble des mots du code qui n'admettent aucun autre mot comme diviseur à gauche. Puisque le code est net toute séquence

soit admet un $p \in P'_0$ comme diviseur à gauche

soit est un diviseur à gauche d'un $p \in P'_0$.

Donc l'arbre de codage correspondant au code UND est complet et sa fonction de structure $g'(t)$ admet la racine K^{-1}. Il est impossible que $P'_0 \neq P_0$ car ceci impliquerait que $g(t) = g'(t) - \sum n''_i t^i$ (où les n''_i correspondent aux mots de $P_0 - P'_0$) ce qui ne se peut puisque $g(K^{-1}) = g'(K^{-1}) = 0$. Donc $P'_0 = P_0$ et le code est bien UND.

Proposition 4.5.- Si un code est UND

ou bien il est aussi net à gauche (et par conséquent unitaire à droite)

ou bien le code opposé est tel qu'il existe des séquences non bornées dont le premier mot ne peut pas être décodé sans que soit connue la totalité du message.

On a vu (proposition 1.9) que s'il existait une borne L à de telles séquences on aurait $P_n = \emptyset$ pour un $n < \infty$ ou encore (en posant P_0^m = l'ensemble des séquences formées de m mots non vides) que pour un certain n' $P_0^{n'}$ ne contiendrait aucune paire d'éléments dont l'un soit diviseur à gauche de l'autre.

Or il est clair que si P_0 est l'ensemble des mots d'un code net il en est de même de P_0^n et réciproquement car la fonction de structure $g_n(t)$ de P_0^n est donnée par : $1 - g_n(t) = (1 - g(t))^n$ quel que soit P_0 et n'admet la racine K^{-1} que si $g(t)$ l'admet elle-même. Si donc $\{P_0^m\}$ était unitaire, il serait net puisque la fonction génératrice d'un code est la même que celle du code opposé. Donc $\{P_0\}$ lui-même serait à la fois unitaire et net des deux côtés.

BIBLIOGRAPHIE

[1] - J. RIGUET : Sur la représentation des syntaxes. (à paraitre dans : Zeitschrift für math. Log., 1955).

[2] - J. RIGUET : Relations binaires (Bull. Soc. math. France, t. 76, 1948, p. 114-155).

P. DUBREIL : Trois mémoires intitulés "Contribution à la théorie des demi-groupes. (I, II, III).

[3] - I : Mem. Acad. Sci. Paris, t. 63, 1941, p. 1-52.

[4] - II : Rendiconti di Matematica e delle sue applicazioni, Università di Roma, serie V, vol. X , 1951, p. 183-200.

[5] - III : Bull. Soc. math. France, t. 81 , 1953 , p. 289-306.

[6] - W. FELLER : Introduction to Probability theory. (Wiley, New York 1950) chapitre 12.

[7] - B. MANDELBROT : Thèse, Paris 1952.

[8] - B. MANDELBROT : On recurrent noise limiting codes (Proc. Symp. Inf. Networks, Brooklin, 1954, p. 206-221).

[9] - B. MANDELBROT : Mémoire dans : Proc. Symp. Inf. Theory, 1955 (à paraitre chez Butterworth,London).

[10] - P. ELIAS : Idem

[11] - D. SLEPIAN : Idem

[12] - A.A. SARDINAS et G.W. PATTERSON : A necessary and sufficient condition for unique decomposition ... of coded messages. (Univ. Pennsylvania, Res. Div. reports 50.27 , march 1950).
