

R. MOHR

**Généralisation de la notion de langage à contexte libre.  
Application à l'analyse syntaxique de figures**

*Revue française d'automatique informatique recherche opérationnelle.  
Informatique théorique*, tome 9, n° R2 (1975), p. 55-88

[http://www.numdam.org/item?id=ITA\\_1975\\_\\_9\\_2\\_55\\_0](http://www.numdam.org/item?id=ITA_1975__9_2_55_0)

© AFCET, 1975, tous droits réservés.

L'accès aux archives de la revue « Revue française d'automatique informatique recherche opérationnelle. Informatique théorique » 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/>

## **GENERALISATION DE LA NOTION DE LANGAGE A CONTEXTE LIBRE. APPLICATION A L'ANALYSE SYNTAXIQUE DE FIGURES**

par R. MOHR <sup>(1)</sup>

---

Communiqué par M. NIVAT

*Résumé. — La notion de « langage à contexte libre » se généralise sans difficulté dans une structure algébrique quelconque en « ensemble algébrique ». Le but de cet article est de montrer, à partir d'un exemple, comment sans trop de difficultés, peuvent être généralisés les algorithmes d'analyse syntaxique; une des applications possible est leur utilisation dans des algorithmes de reconnaissances dans des structures plus complexes que celles du monoïde libre.*

Le but de cet article <sup>(2)</sup> est de présenter les ensembles algébriques comme une généralisation de la notion de langage à contexte libre, puis d'illustrer cette généralisation en l'appliquant à une structure permettant de décrire des images. Enfin nous montrerons sur cet exemple comment peuvent être étendues les techniques d'analyse syntaxique.

Le premier paragraphe expose brièvement les rappels concernant les ramifications, le théorème du point fixe et les schémas fonctionnels; les définitions et les résultats y sont exposés sous une forme simple suffisant à l'utilisation qui en est faite.

Le second paragraphe définit les ensembles algébriques sur une structure quelconque et établit le lien entre les ensembles engendrés par une grammaire à contexte libre et les ensembles algébriques; et ensuite sont introduits les arbres syntaxiques correspondant à ces grammaires. Les résultats énoncés dans ces paragraphes sont soit connus, soit des généralisations simples de

---

(1) Maître-assistant à l'Université de Nancy I, U.E.R. de Mathématiques, Nancy.

(2) Il reprend une grande partie de Mohr [13] et nous renvoyons le lecteur à ce dernier travail pour les démonstrations qui ne seront pas exposées ici.

résultats connus. Après avoir introduit une structure sur des figures (chapitre 4), nous nous intéressons à l'analyse syntaxique pour des grammaires sur cette structure; le lecteur familier des techniques d'analyses syntaxiques utilisés en théorie des langages remarquera l'adaptation évidente qui en est faite.

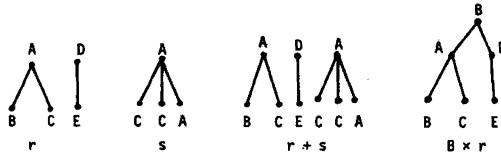
## 1. PRELIMINAIRE

**Notation :**  $\mathbf{N}$  et  $\mathbf{R}$  désignent respectivement l'ensemble des entiers naturels et celui des nombres réels.

Si  $E$  est un ensemble  $\mathfrak{p}(E)$  désigne l'ensemble des parties de  $E$ .

### 1.1. Ramifications. Bilangage [16], [17]

Soit  $V$  un ensemble, on appelle ramification sur  $V$ , toute suite d'arborescence orienté étiqueté par des éléments de  $V$ . On note  $\hat{V}$  l'ensemble de ces ramifications; on munit  $\hat{V}$  d'une loi de composition interne; la concaténation notée  $+$ , et d'une loi de composition externe à opérateur dans  $V$  l'enracinement noté  $\times$ . La ramification vide est notée  $\Lambda$ . Les schémas ci-dessous, montrent des exemples de ramifications :



La proposition suivante nous donne un principe de récurrence sur  $\hat{V}$ .

**Proposition :** Soit  $p$  un prédicat sur  $V$

Si  $p(\Lambda)$  est vrai

Si  $(\forall r, s \in \hat{V})(\forall A \in V) p(r)$  et  $p(s) \Rightarrow p(A \times r + s)$  alors  $\forall r \in \hat{V}, p(r)$  est vrai.

On peut aussi s'en servir pour définir des fonctions par récurrence, par exemple :  $\rho$  : mot des racines :  $\rho : \hat{V} \rightarrow V^*$

$$\rho(\Lambda) = \Lambda$$

$$\rho(A \times r + s) = A\rho(s)$$

dans l'exemple  $\rho(r) = AD$

$F_A$  : famille de père  $A$      $F_A : V \rightarrow p(V^*)$

$$F_A(\Lambda) = \{\emptyset\}$$

$$F_A(B \times r + s) = \begin{cases} \text{si } B \neq A & , \quad F_A(r) \cup F_A(s) \\ \text{si } B = A & , \quad F_A(r) \cup F_A(s) \cup \{\rho(r)\} \end{cases}$$

dans l'exemple  $F_A(r + s) = \{BC, CCA, \Lambda\}$ .

Une grammaire, au sens de [17], est un triplet

$$= (V, \rightarrow, X) \text{ où } V \text{ est un ensemble}$$

$$\rightarrow \text{ une relation entre } V \text{ et } V^*$$

$$X \text{ une partie de } V^*.$$

Une ramification  $r$  est engendrée au sens large par  $\mathfrak{g}$  si elle vérifie :

$$(\forall A \in V)(\forall \alpha \in F_A(r)) \quad A \rightarrow \alpha.$$

Une ramification  $r$  est engendrée au sens strict par  $\mathfrak{g}$  si elle est engendrée au sens large et si de plus  $\rho(r) \in X$ .

## 1.2. Fonctions continues et théorèmes du point fixe

*Définition* : Soit  $(A, \leq)$  un ensemble ordonné. On dit que  $(A, \leq)$  est un treillis complet si tout sous-ensemble  $B$  de  $A$  admet une borne supérieure.

Soient  $(A, \leq)$  et  $(B, \leq)$  deux treillis complets et  $f$  une application de  $A$  dans  $B$ ;  $f$  est dite continue si pour toute suite croissante  $(x_n)$  d'éléments de  $A$

$$f(\sup(x_n)) = \sup(f(x_n))$$

On démontre les résultats suivants :

**Proposition :**

- tout treillis complet  $A$  admet une borne inférieure  $\perp_A$
- le composé d'applications continues est une application continue.

**Théorème 1.** (Théorème du point fixe) : Soit  $(A, \leq)$  un treillis complet et  $f$  une application continue de  $A$  dans  $A$ . L'équation  $f(x) = x$  admet une solution minimum  $x_0$  avec

$$x_0 = \lim_n f^n(\perp_A)$$

Dans la suite nous appliquerons ces résultats en prenant pour  $A$  l'ensemble  $p(E)$  des parties d'un ensemble et pour  $\leq$  la relation d'inclusion  $\subset$ , et plus généralement nous prendrons  $A = [p(E)]^n$  et  $\leq = \subset^n$ .

On démontre facilement :

**Proposition :**

- Soit  $E$  un ensemble,  $(\mathfrak{p}(E)^n, \subset^n)$  est un treillis complet.
- Soit  $f$  une application de  $E^n$  dans  $E$ ; prolongeons  $f$  en une application  $f : \mathfrak{p}(E)^n \rightarrow \mathfrak{p}(E)$  en posant

$$f(A_1, \dots, A_n) = \bigcup_{\substack{(x_1, \dots, x_n) \\ \in A_1 \times \dots \times A_n}} \{ f(x_1, \dots, x_n) \}$$

alors  $f$  est continue.

### 1.3. Schémas fonctionnels

Nous utilisons ici le vocabulaire de [5] mais en utilisant comme outils les schémas fonctionnels [10].

Une théorie  $\Pi$  est une famille  $(F_n)_{n \in \mathbb{N}}$  d'ensembles disjoints. Posons  $F = \bigcup_{n \geq 0} F_n$ .

Soit alors  $G = (F, \{ X \}, ::=, X)$  la grammaire à contexte libre définie par

$$(\forall n \geq 0)(\forall a \in F_n) \quad X ::= aX^n.$$

Tout mot engendré par  $G$  est appelé *schéma fonctionnel* sur  $\Pi$  et nous noterons  $\bar{\Pi}$  l'ensemble des schémas fonctionnels sur  $\Pi$ .

On démontre les résultats suivants :

**Proposition 1 :** Soit  $a \in F_0$ ; si  $xy$  et  $z$  sont des schémas fonctionnels sur  $\Pi$  alors  $xzy$  est un schéma fonctionnel sur  $\Pi$ .

Pour tout schéma fonctionnel  $x$ , il existe un entier  $n$  unique, il existe  $a \in F_n$  et  $a_1, \dots, a_n \in \bar{\Pi}$  uniques tels que  $x = aa_1a_2 \dots a_n$ .

### 1.4. Algèbre sur une théorie

Soit  $\Pi = (F_n)_{n \in \mathbb{N}}$ , une théorie.

**Théorème 2 :** Soit un ensemble  $E$ , et, pour chaque  $n$ , une application  $\varphi_n a \rightarrow \hat{a}$  de  $F_n$  dans l'ensemble des applications de  $E^n$  dans  $E$ . Alors, il existe une application et une seule  $\varphi_E : \bar{\Pi} \rightarrow E$  ( $\varphi_E(x) = \bar{x}$ ), telle que, pour tout  $a \in F_n$  et pour tout  $x_1, \dots, x_n \in \bar{F}$

$$\overline{ax_1 \dots x_n} = \hat{a}(\bar{x}_1, \dots, \bar{x}_n).$$

S'il n'y a pas de confusion possible, nous noterons  $\varphi$  au lieu de  $\varphi_E$ . Pour  $a$  appartenant à  $F_0$ , on identifie  $\bar{a}$  et  $a$ .

Exemple : Soit  $E = F^*$ ;

$$\forall a \in F_n \quad \varphi_n(a) : (x_1, \dots, x_n) \rightarrow ax_1 \dots x_n \in F^*.$$

Dans ce cas, l'application  $\varphi_{F^*}$  de  $\bar{\Pi}$  dans  $F^*$  est l'injection canonique de  $\bar{\Pi}$  dans  $F^*$ .

$E$  est une  $\Pi$ -algèbre si on se donne, pour tout  $n$ , une application de  $F_n$  dans l'ensemble des applications de  $E^n$  dans  $E$ . Les éléments de  $F_n$  représentent les opérations  $n$ -aires sur le  $\Pi$ -algèbre  $E$ .

$\bar{\Pi}$  est donc une  $\Pi$ -algèbre particulière que nous appellerons  $\Pi$ -algèbre libre.

Notons que si  $E$  est une  $\Pi$ -algèbre,  $p(E)$  peut être muni canoniquement d'une structure de  $\Pi$ -algèbre en posant

$$\varphi_{n,p(E)}(a) = \hat{a} \quad \text{avec} \quad \hat{a}(A_1, \dots, A_n) = \bigcup_{\substack{(x_1, \dots, x_n) \\ \in A_1 \times \dots \times A_n}} \hat{a}(x_1, \dots, x_n)$$

## 1.5. Morphismes

Les schémas fonctionnels nous offrent une notation permettant de représenter les résultats obtenus en appliquant des applications ( $\in F_n, n > 0$ ) composées entre elles à des objets élémentaires ( $\in F_0$ ). Mais il nous sera nécessaire de représenter des applications composées en tant qu'application et c'est ce que nous allons faire par les morphismes.

Soit  $\Pi = (F_n)_{n \geq 0}$  une théorie; on appelle *morphisme à  $n$  variables* de  $\Pi$  les schémas fonctionnels sur  $(F_0 \cup \{1, \dots, n\}, F_1, \dots, F_n, \dots)$ . Notons  $M_n$  cet ensemble et  $M = \bigcup_{n \geq 0} M_n$ .

Remarquons que  $\bar{\Pi} = M_0$  et que  $M_i \subset M_{i+1}, i \in \mathbb{N}$ .

Soit  $E$  une  $\Pi$ -algèbre. Soit  $X = \{x_1, \dots, x_n, \dots\}$  une suite d'éléments de  $E$ . L'application  $\varphi_0$  de  $F_0$  dans  $E$  (cf. théorème 2) peut se prolonger en  $\varphi_{0X}$  définie sur  $F_0 \cup \mathbb{N}$

$$\varphi_{0X}(a) \text{ si } a \in F_0 \quad \text{alors } \varphi_0(a) \\ \text{sinon } x$$

D'après le théorème 1,  $\varphi_{0X}$  se prolonge de façon unique en une application  $\varphi_X$  de  $M$  dans  $E$  telle que :

$$\varphi_X(aa_1 \dots a_m) = \hat{a}(\varphi_X(a_1), \dots, \varphi_X(a_m))$$

et donc tout morphisme  $\omega$  de  $M$  définit une application  $\hat{\omega}$  de  $E^{\mathbb{N}}$  dans  $E$

$$X \rightarrow \varphi_X(\omega).$$

En particulier si  $\omega \in M_n$ ,  $\hat{\omega}$  définit une application de  $E^n$  dans  $E$ .

Définissons alors la *composition des morphismes*.

Soit  $\omega \in M_n$  et  $\omega_1, \dots, \omega_n$   $n$  morphismes de  $M_m$ .

Par définition posons  $\omega(\omega_1, \dots, \omega_n) = S(\omega, (1/\omega_1, 2/\omega_2, \dots, n/\omega_n))$  où  $S(\alpha, (a_1/\alpha_1, \dots, a_n/\alpha_n))$  désigne le résultat de la substitution simultanée dans  $\alpha$  de toutes les occurrences de  $a_1$ , par  $\alpha_1, \dots$ , de  $a_n$  par  $\alpha_n$ . La proposition 1 nous assure que  $\omega(\omega_1, \dots, \omega_n)$  est encore un morphisme.

**Théorème 3 :** Soit  $E$  une  $\Pi$ -algèbre,  $\omega \in M_n$  et  $n$  morphismes  $\omega_i \in M_m$  alors, pour tout  $m$  uplet  $(x_1, \dots, x_m)$  de  $E$

$$\widehat{\omega(\omega_1, \dots, \omega_n)}(x_1, \dots, x_m) = \hat{\omega}(\hat{\omega}_1(x_1, \dots, x_m), \dots, \hat{\omega}_n(x_1, \dots, x_m)).$$

*Démonstration :* nous supposons que  $\{x_1, \dots, x_m\} \cap F = \emptyset$ .

Soit  $\Pi' = (F_0 \cup \{x_1, \dots, x_m\}, F_1, \dots, F_k, \dots)$ .

$E$  est aussi une  $\Pi'$ -algèbre : il suffit de prolonger  $\varphi_0$  en  $\varphi'_0$  :

$$\forall a \in F_0 \quad \varphi'_0(a) = \varphi_0(a) \quad ; \quad \text{pour } i = 1, \dots, m \quad \varphi'_0(x_i) = x_i.$$

D'après le théorème 2,  $\varphi_0$  se prolonge de manière unique en une application  $\varphi'$  de  $\overline{\Pi}'$  dans  $E$  telle que

$$\varphi'(aa_1 \dots a_k) = a(\varphi(a_1), \dots, \varphi(a_k)).$$

Or le résultat du théorème 3 est vrai pour la  $\Pi$ -algèbre  $\Pi'$ ; en effet

$$\begin{aligned} \omega(\omega_1, \dots, \omega_n)(x_1, \dots, x_m) &= S(S(\omega, 1/\omega_1, \dots, n/\omega_n), (1/x_1, \dots, m/x_m)) \\ &= S(\omega, (S(\omega_1, (1/x_1, \dots, m/x_m), \dots, S(\omega_n, (1/x_1, \dots, m/x_m)))) \\ &= \omega(\omega_1(x_1, \dots, x_m), \dots, \omega_n(x_1, \dots, x_m)). \end{aligned}$$

Le résultat se projette alors par  $\varphi'$  dans la  $\overline{\Pi}$ -algèbre  $E$ .

**Corollaire :** Soit  $\omega \in M_n$  et  $\omega_1, \dots, \omega_n$   $n$  schémas fonctionnels ( $\in \overline{\Pi}$ )

$$\hat{\omega}(\varphi(\omega_1), \dots, \varphi(\omega_n)) = \varphi(\omega(\omega_1, \dots, \omega_n)).$$

Pour faire des raisonnements par récurrence nous associons à chaque morphisme  $\omega$  un entier  $d(\omega)$  que nous appellerons *degré* de  $\omega$  (le degré défini ici est le même que celui défini dans [5]) le degré est défini par récurrence :

$$\forall a \in F_0 \quad d(a) = 1$$

$$\forall a \in \mathbf{N} \quad d(a) = 0$$

$$(\forall a \in F_n)(\forall a_i \in M) \quad d(a a_1 \dots a_n) = 1 + \sum_{i=1}^n d(a_i).$$

## 2. ENSEMBLES ALGEBRIQUES

La notion de langage contexte-libre se généralise aisément dans une structure algébrique quelconque : soit par exemple  $E$  un ensemble muni de deux opérations binaires  $\varphi$  et  $\psi$ ; si  $A$ ,  $B$  et  $C$  représentent des catégories syntaxiques, on peut par exemple dire

que  $x \in B$  et  $y \in C \Rightarrow \varphi(x, y) \in A$  , qu'un élément  $x_0 \in A$

que  $x \in A$  et  $y \in B \Rightarrow \psi(x, y) \in B$  , qu'un élément  $y_0 \in B$

que  $x \in A$  et  $y \in B \Rightarrow \psi(x, y) \in C$  et  $\varphi(y, y) \in C$

d'où les inéquations :

$$A \supset \{x_0\} \cup \varphi(B, C)$$

$$B \supset \{y_0\} \cup \psi(A, B)$$

$$C \supset \psi(A, B) \cup \varphi(B, B)$$

Si de plus on impose que  $A$ ,  $B$ ,  $C$  sont les plus petits sous-ensembles vérifiant les conditions ci-dessus on aboutit au système

$$A = \{x_0\} \cup \varphi(B, C)$$

$$B = \{y_0\} \cup \psi(A, B)$$

$$C = \psi(A, B) \cup \varphi(B, B)$$

Le système dont les composantes de la solution minimale sont appelées ensembles algébriques. Si on se place sur le monoïde libre  $T^*$ , la loi de composition étant la concaténation, on définit ainsi les langages à contexte libre, comme une composante des solutions minimales des systèmes à point fixe, c'est pourquoi les langages à contexte libre sont aussi appelés langages algébriques.

Sur le monoïde libre  $T^*$  on peut définir un langage à contexte libre soit par un système à point fixe sur l'ensemble des parties de  $T^*$ , soit à l'aide d'une grammaire à contexte libre, soit à l'aide des bilangages grammaticaux. Les systèmes à point fixe simplifient beaucoup de démonstrations; les grammaires et les dérivations permettent de faire le lien avec les autres types de grammaires; les ramifications et les bilangages sont utiles pour aborder l'analyse syntaxique et l'ambiguïté. Ces trois notions sont introduites ici.

### 2.1. Systèmes algébriques

Soit  $\Pi = (F_n)_{n \geq 0}$  une théorie;

soit  $E$  une  $\Pi$ -algèbre et pour tout  $n$ ,  $n \geq 0$  soit  $\mathcal{F}_n = \{\hat{a} \mid a \in F_n\}$  les réalisations des schémas d'opérations à  $n$  variables.

$$\text{Posons } \mathcal{F} = \bigcup_{n \geq 0} \mathcal{F}_n.$$



On appelle *famille d'applications engendrées par  $\mathcal{F}$*  et on note  $\text{Eng}(\mathcal{F})$  le plus petit ensemble de fonctions stable par composition et contenant les fonctions projections et  $\mathcal{F}$ .

Pour toute application  $f$  de  $E^n$  dans  $E$  nous noterons  $f$  l'application de  $\mathfrak{p}(E)^n$  dans  $\mathfrak{p}(E)$

$$\underline{f}(A_1, \dots, A_n) = \bigcup_{\substack{(x_1, \dots, x_n) \\ \in A_1 \times \dots \times A_n}} \{ f(x_1, \dots, x_n) \}.$$

Un système algébrique sur  $E$  sera un système à point fixe sur  $\mathfrak{p}(E)$  du type : pour  $i = 1, \dots, n$   $X_i = \bigcup_{j=1}^{p_i} f_{ij}(X_1, \dots, X_n)$  où  $f_{ij} \in \text{Eng}(\mathcal{F})$ .

Un tel système admet une solution minimale (cf. 1.1) (pour un théorème d'unicité, voir Lescanne [11]). La solution minimale peut être construite ainsi :

soit  $\Phi$  l'application de  $\mathfrak{p}(E)^n$  dans lui-même définie par :

$$\Phi(A_1, \dots, A_n) = (A'_1, \dots, A'_n) \quad \text{où} \quad A'_i = \bigcup_{j=1}^{p_i} f_{ij}(A_1, \dots, A_n)$$

alors  $(A_1, \dots, A_n) = \bigcup_{n \geq 0} \Phi^n(\emptyset, \dots, \emptyset)$  est la solution minimale du système.

Toute composante de la solution minimale d'un tel système sera appelée *ensemble algébrique* de la  $\Pi$ -algèbre  $E$ .

## 2.2. Grammaire algébrique

Soit  $\Pi = (F_n)$  ( $F = \cup F_n$ ) une théorie et soit  $N$  un ensemble tel que  $F \cap N = \emptyset$ ; nous noterons  $\bar{\Pi}(N)$  les schémas fonctionnels sur  $(F_0 \cup N, F_1, F_2, \dots)$ .

Une *grammaire algébrique*  $\mathcal{G}$  sur  $\Pi$  (ou encore  $\Pi$ -grammaire) est la donnée d'un ensemble fini  $N$ , d'un élément  $X$  de  $N$ , et d'une relation finie  $:: =$  entre  $N$  et  $\bar{\Pi}(N)$ ; on notera  $\mathcal{G} = (N, :: =, X)$  :

On peut alors définir la même dérivation que pour la grammaire à contexte libre  $(N, F, :: =, X)$  :

On dit que  $\alpha$  se réécrit  $\beta$ , et on notera  $\alpha \succ_{\mathcal{G}} \beta$  si et seulement si :

$$(\exists \lambda, \lambda' \in (F \cup N)^* \quad (\exists A \in N) \quad (\exists \varphi \in \bar{\Pi}(N)) \quad (\alpha = \lambda A \lambda' \text{ et } \beta = \lambda \varphi \lambda' \text{ et } A :: = \varphi).$$

**Proposition 2.** Si  $\alpha \in \bar{\Pi}(N)$  et  $\alpha \succ_{\mathcal{G}} \beta$  alors  $\beta \in \bar{\Pi}(N)$ .

C'est une conséquence immédiate de la proposition 1.

Soit alors  $\succ_{\mathcal{G}}^*$  la fermeture transitive et réflexive de  $\succ_{\mathcal{G}}$ . Soit

$$F(\mathcal{G}) = \{ \alpha \in \bar{\Pi} \mid X \succ_{\mathcal{G}}^* \alpha \}.$$

Soit  $E$  une  $\Pi$ -algèbre et soit  $\varphi$  l'application de  $\bar{\Pi}$  dans  $E$  (cf. théorème 2). On appelle *ensemble engendré par  $\mathcal{G}$  sur  $E$*  et on note  $L_E(\mathcal{G})$  l'ensemble  $\varphi(F(\mathcal{G}))$ .

### 2.3. Système associé à une grammaire

Soit  $\Pi$  une théorie et  $\mathcal{G} = (\{ A_1, \dots, A_n \}, ::=, A_1)$  une  $\Pi$ -grammaire avec pour  $i = 1, \dots, n$

$$A_i ::= \omega_{ij} \quad j = 1, \dots, p_i.$$

Chaque  $\omega_{ij}$  de  $\bar{\Pi}(\{ A_1, \dots, A_n \})$  peut s'écrire de façon unique  $\omega_{ij} = \omega'_{ij}(A_1, \dots, A_n)$  avec  $\omega'_{ij} \in M_n$  (morphisme à  $n$  variables).

Soit alors  $E$  une  $\Pi$ -algèbre et soit  $\mathcal{F}_n = \{ \hat{a} \mid a \in F_n \}$  la réalisation dans  $E$  des schémas d'application de  $F_n$ . Soit  $\mathcal{F} = \bigcup_{n \geq 0} \mathcal{F}_n$ ; alors

**Proposition :**  $f \in \text{Eng}(\mathcal{F}) \Leftrightarrow \exists m \exists \omega \in M_m \hat{\omega} = f$ .

La démonstration est rapide;

Soit  $\mathcal{M} = \{ f \mid \exists m \exists \omega \in M_m \hat{\omega} = f \}$ ;  $\mathcal{M}$  contient  $\mathcal{F}$  et est stable par composition (théorème 3);  $\mathcal{M}$  contient aussi les projections :

$$\forall k \in \mathbb{N} \quad k \in M_p \quad (p \geq n) \quad \text{et} \quad \hat{k}(x_1, \dots, x_p) = x_k$$

donc  $\mathcal{M} \subset \text{Eng}(\mathcal{F})$ .

Réciproquement procédons par récurrence sur le nombre  $n$  de fois qu'il faut appliquer l'opérateur de composition pour obtenir un élément  $f$  de  $\text{Eng}(\mathcal{F})$

Si  $n = 0$ ,  $f$  est une projection donc  $f \in \mathcal{M}$

Si  $n > 0$ ,  $f = g(f_1, \dots, f_k)$  par hypothèse de récurrence  $f_i \in \mathcal{M}$  donc  $f \in \mathcal{M}$ .

En conséquence à toute  $\Pi$ -grammaire on peut faire correspondre un système sur la  $\Pi$ -algèbre  $E$  (et réciproquement) en prenant

$$A_i = \bigcup_{j=1}^{p_i} \hat{\omega}'_{ij}(A_1, \dots, A_n)$$

où  $\omega'_{ij}$  est le morphisme de  $M_n$  associé à  $\omega_{ij}$ .

**Théorème 4.** Soit  $\mathcal{G}_i = (\{ A_1, \dots, A_n \}, ::=, A_i)$   $n$ -grammaires et soit pour  $i = 1, \dots, n$   $L_i$  le langage engendré par  $\mathcal{G}_i$  sur la  $\Pi$ -algèbre  $E$ .

Alors  $(L_1, \dots, L_n)$  est la solution minimale du système à point fixe  $S$  associé à  $\mathcal{G}$ .

**Lemme 1.** Le théorème est vrai pour la  $\Pi$ -algèbre  $\overline{\Pi}$ . En effet dans ce cas nous sommes ramenés au monoïde libre où ce résultat est connu.

**Lemme 2.** Soit  $S'$  le système à point fixe associé à  $\mathcal{G}$  sur la  $\Pi$ -algèbre  $\overline{\Pi}$  et soit  $(F_1, \dots, F_n)$  sa solution minimale ; soit  $(K_1 \dots K_n)$  la solution minimale de  $S$ , alors  $(K_1, \dots, K_n) = (\varphi(F_1), \dots, \varphi(F_n))$

$$(K_1, \dots, K_n) = \bigcup_{k \geq 0} (K_1^k, \dots, K_n^k) \quad \text{où} \quad (K_1^k, \dots, K_n^k) = \Phi^k(\emptyset, \dots, \emptyset)$$

où  $\Phi$  est l'application associée au système  $S$ .

$$\text{De même } (F_1, \dots, F_n) = \bigcup_{k=0} (F_1^k, \dots, F_n^k) \text{ avec } (F_1^k, \dots, F_n^k) = \Phi^k(\emptyset, \dots, \emptyset).$$

Procédons par récurrence sur  $k$  :

$$\varphi(F_i^0) = K_i^0 = \emptyset$$

au rang  $k + 1$  :

$$F_i^{k+1} = \bigcup_{j=1}^{p_i} \omega_{ij}(F_1^k, \dots, F_n^k) = \bigcup_{j=1}^{p_i} \bigcup_{\substack{y_1 \dots y_n \\ \in F_1^k \times \dots \times F_n^k}} \widehat{\omega}_{ij}'(y_1, \dots, y_n)$$

en appliquant l'hypothèse de récurrence et le corollaire du théorème 3

$$\begin{aligned} \varphi(F_i^{k+1}) &= \bigcup_{j=1}^{p_i} \bigcup_{\substack{x_1 \dots x_n \\ \in K_1^k \times \dots \times K_n^k}} \widehat{\omega}_{ij}'(x_1, \dots, x_n) \\ &= \bigcup_{j=1}^{p_i} \widehat{\omega}_{ij}'(K_1^k, \dots, K_n^k) = K_i^{k+1} \end{aligned}$$

D'où le lemme 2.

Le théorème résulte alors immédiatement de ces deux lemmes et de la définition même des  $L_i$ .

#### 2.4. Grammaires (ou système) réduites

*Définition :* Soit  $E$  une  $\Pi$ -algèbre. Soient  $\mathcal{G}_1$  et  $\mathcal{G}_2$  deux  $\Pi$ -grammaires. On dit que  $\mathcal{G}_1$  est équivalente à  $\mathcal{G}_2$  sur  $E$  si

$$L(\mathcal{G}_1) = L(\mathcal{G}_2).$$

On dit que  $\mathcal{G}_1$  est équivalente à  $\mathcal{G}_2$  si  $F(\mathcal{G}_1) = F(\mathcal{G}_2)$ . Dans ce cas  $\mathcal{G}_1$  est équivalente à  $\mathcal{G}_2$  sur toute  $\Pi$ -algèbre.

*Réduction inférieure et supérieure :*

On généralise sans difficultés ces notions définies sur les grammaires à contexte libre [1] : Pour toute  $\Pi$ -grammaire  $\mathcal{G} = (N, ::=, X)$  il existe une  $\Pi$ -grammaire  $\mathcal{G}' = (N', ::=', X)$  équivalente telle que :

- a)  $N' \subset N, ::= ' \subset ::=$
- b)  $(\forall A \in N')(\exists \alpha \in \overline{\Pi}) A \succ_{\mathcal{G}'}^* \alpha$  (réduite inférieure)
- c)  $(\forall A \in N')(\exists \lambda, \lambda' \in (F \cup N)^*) X \succ_{\mathcal{G}'}^* \lambda A \lambda'$  (réduite supérieure).

Nous supposons dans la suite que toutes nos  $\Pi$ -grammaires sont réduites inférieurement et supérieurement.

*Réduite de Chomsky*

Dans le cas des  $C$ -grammaires, on dit qu'une grammaire est sous la forme réduite de Chomsky, si chacune de ses règles est d'un des trois types suivants :

$$A ::= BC \quad , \quad A ::= B \quad , \quad A ::= a$$

où  $A, B, C$  sont des non terminaux et  $a$  est un terminal.

Pour toute  $C$ -grammaire, il existe une  $C$ -grammaire équivalente qui est sous forme réduite de Chomsky. Nous allons généraliser ce résultat.

*Définition :* Une  $\Pi$ -grammaire  $\mathcal{G} = (N, ::=, X)$  est dite  $C$ -réduite si chacune de ses règles est de la forme :

$$A ::= a\alpha \quad \alpha \in N^* \quad a \in F_{|\alpha|}$$

ou

$$A ::= B \quad B \in N$$

Une  $\Pi$ -grammaire est donc  $C$ -réduite lorsque le degré du morphisme attaché à chaque second membre de règle est inférieur ou égal à 1.

On démontre par récurrence sur le degré des morphismes le théorème suivant :

**Théorème 5.** Pour toute  $\Pi$ -grammaire  $\mathcal{G}$ , il existe une  $\Pi$ -grammaire  $\mathcal{G}'$  équivalente qui est  $C$ -réduite.

### 3. STRUCTURES ASSOCIEES AUX GRAMMAIRES

Dans le cas des langages à contexte libre, les « marqueurs de phrases » ont leurs nœuds étiquetés par des non terminaux. Les théories algébriques évoquées ici étant plus générales que celles du monoïde, il nous faudra, pour plus de précision, étiqueter les nœuds avec les numéros des règles appliquées dans les dérivations. En effet si  $A ::= + BC$  et  $A ::= \times BC$  il nous conviendra de distinguer ces deux règles.

Nous supposons donc que toutes les règles des grammaires sont numérotées par des entiers  $p, p \in R, R \subset \mathbb{N}$ .

### 3.1. Bilangages des structures

Soit  $\mathcal{G} = (N, ::=, X)$  une  $\Pi$ -grammaire ( $N = \{ A_1, \dots, A_n \}$ ). Ses règles sont numérotées par des entiers  $p, p \in R \subset \mathbb{N}$ .

Pour tout  $A$  de  $N$ , posons :

$$T(A) = \{ p \in R \mid \text{la règle de numéro } p \text{ est } A ::= \alpha, \alpha \in \overline{\Pi} \}$$

( $T(A)$  est l'ensemble des numéros des règles « terminales » du premier membre  $A$ )

$$N(A) = \{ p \in R \mid \text{la règle de numéro } p \text{ est } A ::= \alpha, \alpha \in \overline{\Pi}(N) - \overline{\Pi} \}$$

$$P(A) = N(A) \cup T(A)$$

$P(A)$  est donc l'ensemble des numéros de règles de premier membre  $A$ .

Soit alors, pour tout  $A$  de  $N$ ,  $\mathcal{G}_A = (V, \rightarrow, Y_A)$  la grammaire (au sens de [17], où :

$$V = R$$

$$Y_A = P(A)$$

$\rightarrow$  est définie par :

si  $p$  est le numéro de la règle  $B ::= \alpha_0 A'_1 \alpha_1 \dots A'_k \alpha_k$  ( $\alpha_i \in F^*$ ,  $A'_i \in N$ ) alors  $p \rightarrow p_1, \dots, p_k$  pour tout  $(p_1, \dots, p_k) \in P(A'_1) \times \dots \times P(A'_k)$ . En particulier, si  $p \in T(A)$  alors  $p \rightarrow \Lambda$ .

On appelle *bilangage des structures de  $\mathcal{G}$*  le bilangage grammatical engendré au sens strict par  $\mathcal{G}_X$ .

Nous le noterons  $\mathcal{BL}(\mathcal{G}, X)$ .

Nous nous intéresserons aussi aux ramifications engendrées au sens large par  $\mathcal{G}_X$  et dont le mot des racines a pour longueur 1. Nous noterons  $\mathcal{B}(\mathcal{G})$  ce bilangage.  $\mathcal{B}(\mathcal{G}) = \bigcup_{A \in N} \mathcal{BL}(\mathcal{G}, A)$ .

Toute ramification  $r$  de  $\mathcal{B}(\mathcal{G})$  se décompose de manière unique sous la forme  $r = p \times (r_1 + r_2 + \dots + r_k)$ ; si  $p \in P(A)$  alors  $r \in \mathcal{BL}(\mathcal{G}, A)$ ; de même  $r_i \in \mathcal{B}(\mathcal{G})$  et donc nous avons la proposition suivante :

**Proposition :** Pour toute ramification  $r$  de  $\mathcal{B}(\mathcal{G})$ , il existe  $p \in R$ , il existe  $k \in \mathbb{N}$  et  $r_1, \dots, r_k \in \mathcal{B}(\mathcal{G})$  tels que :

$$r = p \times (r_1 + \dots + r_k).$$

### 3.2. Schémas fonctionnels associés aux structures

A chaque structure  $r$ , nous allons faire correspondre le schéma fonctionnel qui lui est attaché.

Pour toute règle de numéro  $p$ , notons  $\omega'_p$  le morphisme associé au second membre  $\omega_p$  de cette règle (cf. 2.3).

Soit  $\omega_p = \omega_0 A'_1 \alpha_1 \dots A'_k \alpha_k$ ; où  $\alpha_i \in F^*$ ,  $A'_i \in N$ .

Posons :  $\Omega(p \times \Lambda) = \omega'_p$

$$\Omega(p \times (r_1 + r_2 + \dots + r_k)) = \omega'_p(\Omega(r_1), \dots, \Omega(r_k)).$$

Il est immédiat que :

$$\forall r \in \mathcal{B}(\mathcal{G}) \quad \Omega(r) \in \bar{\Pi}.$$

Posons  $X_i = \{ \Omega(r) \mid r \in \mathcal{B}(\mathcal{G}, A_i) \}$   $1 \leq i \leq n$ .

On démontre alors :

**Proposition :** Les  $X_i$  forment la solution minimale du système à point fixe sur  $\bar{\Pi}$  associé à  $\mathcal{G}$ .

*Définition :* Soit  $E$  une  $\Pi$ -algèbre. Pour tout  $x$  de  $L(\mathcal{G})$ , on appelle *structure associée à  $x$*  toute ramification  $r$  de  $\mathcal{B}(\mathcal{G}, X)$  telle que  $\varphi \circ \Omega(r) = x$ .

#### 4. UN OUTIL ALGÈBRE POUR LA DESCRIPTION D'IMAGES

Le problème de décrire les langages se pose pour deux problèmes bien distincts : la génération et la reconnaissance.

Pour la génération, deux options fondamentales différentes sont prises :

– l'une dynamique : à chaque image est associé un sous-programme qui la génère; la structure de l'image est celle des appels imbriqués des sous-programmes (par exemple Metavisu [3]);

– l'autre statique : à chaque image est associée une structure en mémoire et la génération se fait par un appel d'entrée-sortie. La seconde méthode permet en particulier de manipuler des objets en trois dimensions (voir Euclid [2]; les objets sont définis à une similitude près et les opérations précisent comment ils sont assemblés).

En reconnaissant des formes on a aussi essayé de décrire des figures par assemblage de figures élémentaires, et essentiellement de morceaux de courbes. Les opérations proposées sont soit des assemblages par raccordement de tangentes (Leydley [12], Eden [4]), soit la figure est considérée comme un graphe et c'est le graphe qui est décrit par des lois de composition (Shaw [18], [19]). Enfin certains auteurs (Kirsch [9], Ota [14]...) ont décrit des figures comme des assemblages de points et ont développé des grammaires de mosaïques. On trouvera dans Fu [7] des références plus complètes sur ce sujet.

Nous allons d'abord présenter un outil pour décrire en particulier des figures, et nous montrerons dans quelle mesure il nous permet de décrire ou de construire des figures.

#### 4.1. Figures. Représentations

##### *Définitions. Notations*

Un espace de figures  $(E, F, G)$  est la donnée de deux ensembles  $E$  et  $F$  et d'un groupe  $G$  opérant sur  $E$  et  $F$ , opérant fidèlement et transitivement <sup>(1)</sup> sur  $F$ .

Tout élément  $g$  de  $G$  définit une application de  $E \cup F$  dans  $E \cup F$  :

$$g : x \rightarrow gx$$

nous noterons  $g(x)$  l'élément  $gx$  et plus généralement si

$$x = (x_1, \dots, x_n) \quad g(x) = (g(x_1), \dots, g(x_n)).$$

Soit un espace de figures  $(E, F, G)$ , une *représentation*  $r$  sur  $(E, F, G)$  est un triplet  $(K, o, e)$  tel que :

$$\begin{aligned} K &\subset E \\ o &\in F \\ e &\in F \end{aligned}$$

$K$  est appelé le *support* de  $r$ ,  $o$  l'*origine* de  $r$  et  $e$  l'*extrémité* de  $r$ .  $o$  et  $e$  sont les *pôles* de  $r$ .

Si  $r$  est une représentation,  $o(r)$ ,  $e(r)$ ,  $\text{supp}(r)$  désignent respectivement l'origine, l'extrémité et le support de  $r$ .

Soit  $R$  l'ensemble des représentations sur  $(E, F, G)$ .

$G$  définit sur  $R$  la relation d'équivalence :

$$r_1 \sim r_2 \Leftrightarrow \exists g \in G \quad g(r_1) = r_2.$$

Une figure est une classe d'équivalence de  $\mathcal{F} = \mathcal{R} / \sim \cdot r \in \mathcal{R}$ , nous noterons  $r$  la figure, classe d'équivalence de  $r$ . Le support d'une figure  $f$  est la classe d'équivalence du support de la représentation  $r$  associée; nous le noterons  $\text{supp}(f)$ .

##### *Exemples d'espaces de figures*

1.  $E = \mathbf{R}^2$ ,  $F = \mathbf{R}^2$ ,  $G$  est le groupe des translations (fig. 3). Parmi les figures, nous trouverons en particulier les vecteurs libres.

2.  $E = \mathbf{R}^3$ ,  $F = \mathbf{R}^3 - \{(0, 0, 0)\}$ ,  $G$  est le groupe des similitudes centrées en  $(0, 0, 0)$ .

---

(1)  $G$  opère fidèlement et transitivement sur  $F$  si et seulement si :

$$\forall x \in F, \quad \forall y \in F, \quad \exists g \in G, \quad g \text{ unique, tel que, } \quad g(x) = y.$$

3.  $E = \mathbf{R}^2$ ,  $F = \mathbf{R}^2 \times \mathbf{R}/2\Pi$ ,  $G$  est le groupe des déplacements du plan.  $G$  opère sur  $F$  de la manière suivante :

Si  $g$  est le composé d'une rotation d'angle  $\theta$  et d'une translation alors :  $g((x, y, \alpha)) = (g(x, y), \alpha + \theta(2\Pi))$ .

Un pôle pourra donc être schématisé par un point du plan origine d'un vecteur unitaire dont l'angle indiquera la valeur de  $\alpha$  (fig. 4). Nous verrons l'intérêt de cet exemple, qui avec l'exemple 1, nous servira d'illustration dans la suite.

4. Comme  $G$  opère fidèlement et transitivement sur lui-même,  $(\emptyset, G, G)$  est un espace de figures.

### Opérations sur les figures

Nous allons nous servir des pôles pour composer les représentations entre elles. Nous définirons trois opérateurs binaires :  $+$ ,  $\circ$ ,  $\times$ ; et une opération unaire :  $-$ .

Soit  $f = \overline{(K, o, e)}$  une figure; on définit  $\underline{-f}$  comme étant la classe de  $(K, e, o)$  (échange de l'origine et de l'extrémité).

Définissons  $f_1 + f_2$  ( $f_1, f_2 \in F$ ). Soient  $r_1 = (K_1, o_1, e_1)$  une représentation de  $f_1$  et de  $r_2 = (K_2, o_2, e_2)$  l'unique représentation de  $f_2$  telle que  $e_1 = o_2$  (l'unique puisque  $G$  opère fidèlement et transitivement sur  $F$ ).  $f_1 + f_2$  est alors définie comme la classe de  $(K_1 \cup K_2, o_1, e_2)$ .  $+$  correspond intuitivement à l'addition vectorielle.

Si on choisit la représentation  $(K_2, o_2, e_2)$  telle que  $o_2 = o_1$ , alors  $f_1 \circ f_2$  est la classe de  $(K_1 \cup K_2, o_1, e_2)$  (mise en commun des origines). Enfin si on choisit  $r_2$  telle que  $e_1 = e_2$  alors  $f_1 \times f_2$  est la classe de  $(K_1 \cup K_2, o_1, e_2)$  (mise en commun des extrémités).

La figure 3 montre ce que sont ces opérations dans le cas de l'exemple 1, la figure 4 dans le cas de l'exemple 3. En 3,  $\circ$  schématise l'origine et  $\times$  l'extrémité. En 4, les pôles appartiennent à  $\mathbf{R}^2 \times \mathbf{R}/2\Pi$  et l'origine  $(x, y, \alpha)$  est schématisée par  $\circ$  muni d'un vecteur unitaire dont l'orientation indique la valeur de  $\alpha$ , et l'extrémité est schématisée par  $\times$  aussi muni d'un vecteur unitaire.

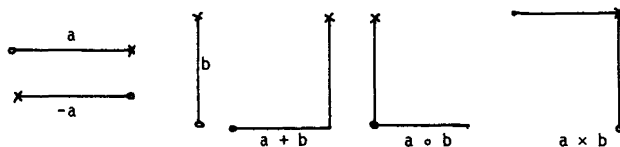


Figure 3



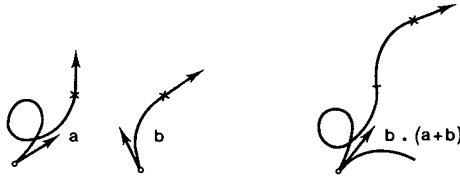


Figure 4

Nous avons bien défini des lois de compositions internes sur  $\mathcal{F}$  : en effet si nous reprenons la construction de  $f_1 \perp f_2$  ( $\perp \in \{+, \circ, \times\}$ ) avec  $r'_1$  et  $r'_2$  :  $r'_1 = g(r_1)$  et donc  $r'_2 = g(r_2)$  d'où  $(K'_1 \cup K'_2, o'_1, e'_2) = g(K_1 \cup K_2, o_1, e_2)$  et donc

$$f_1 \perp f_2 = \overline{(K'_1 \cup K'_2, o'_1, e'_2)} = \overline{(K_1 \cup K_2, o_1, e_2)}$$

Donnons quelques propriétés algébriques :

– les trois lois  $+$ ,  $\circ$ ,  $\times$  sont associatives :

$$f_1 \perp (f_2 \perp f_3) = (f_1 \perp f_2) \perp f_3 \quad \perp \in \{+, \circ, \times\}$$

–  $\circ$  et  $\times$  sont « idem-potentes » :

$$f \times f = f \circ f = f$$

–  $+$  possède un élément neutre, la classe de  $(\emptyset, a, a)$  que nous noterons  $\lambda$  :

$$f + \lambda = \lambda + f = f$$

remarquons que :

$$f_1 \circ f_2 = (f_1 \circ \lambda) + f_2$$

$$f_1 \times f_2 = f_1 + (f_2 \times \lambda)$$

–  $\times$  s'exprime en fonction de  $\circ$  et  $-$  (et réciproquement  $\circ$  s'exprime en fonction de  $\times$  et  $-$ ) :

$$f_1 \times f_2 = -((-f_2) \circ (-f_1)).$$

## 4.2. Figures obtenues à partir d'un alphabet de base

Soit  $\mathcal{F}$  l'ensemble des figures d'un espace de figures  $(E, F, G)$  et  $\mathcal{R}$  celui des représentations ; soit  $A$  un sous-ensemble de  $\mathcal{F}$  appelé alphabet ; soit  $\Sigma$  un ensemble de lois de composition interne sur  $\mathcal{F}$ . On appelle ensemble de figures engendré par  $A$  et on note  $\mathcal{F}(A, \Sigma)$  le plus petit sous-ensemble de  $\mathcal{F}$  contenant  $A$  et stable par les lois de  $\Sigma$ .

On notera  $\mathcal{R}(A, \Sigma)$  l'ensemble des représentations de figures de  $\mathcal{F}(A, \Sigma)$ .

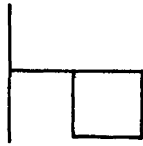


Figure 5

En choisissant pour  $\Sigma$  l'ensemble des lois de composition précédemment introduites, on peut générer toutes les figures connexes comprenant de manière quelconque des figures de  $A$ , les points d'articulation étant les pôles.

Par exemple si  $A$  est l'ensemble  $\{ a, b \}$  de la figure 3, et si on se place dans le cadre de l'exemple 1, on espère obtenir tous les « quadrillages connexes normés ». On obtient par exemple le dessin de la figure 5 avec :

$$b + ((b \circ a) + (a \times ((b \circ a) + b)))$$

ou

$$b + (b \circ a) + a + (-b) + (-a) + b.$$

Formalisons la notion de « toutes les figures connexes obtenues en ajustant sur leurs pôles les figures de  $A$  ».

Définissons sur  $\mathcal{R}$  la relation symétrique :

$$r_1 \Gamma r_2 \Leftrightarrow r_1 \text{ et } r_2 \text{ ont un pôle en commun.}$$

*Définition :* On appelle  $A$ -représentation tout sous-graphe fini connexe de  $(\mathcal{R}, \Gamma)$  dont les éléments sont des représentations de  $A$ .

Le support d'une  $A$ -représentation est la réunion des supports de ses éléments. L'ensemble de ses pôles est celui des pôles de ses éléments.

**Théorème 6.** Soit  $\Sigma = \{ +, \circ, \times \}$ . Pour toute  $A$ -représentation  $R$  de support  $K$ , pour tout  $x$  origine (resp. extrémité) d'un élément de  $K$ , il existe  $y$  tel que :

$$(K, x, y) \in \mathcal{R}(A, \Sigma) \quad (\text{resp. } (K, y, x) \in \mathcal{R}(A, \Sigma)).$$

Mais le choix du pôle  $y$  n'est pas quelconque. Prenons l'exemple suivant : Soit  $T$  le groupe des translations de  $\mathbb{R}^2$ , considérons l'espace de figures  $(\mathbb{R}^2, \mathbb{R}^2, T)$ .

Soit  $\{ a, b \} = A$  l'alphabet décrit à la figure 6.

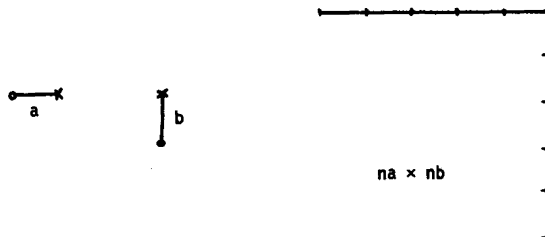


Figure 6

On montre que quel que soit  $r \in \mathcal{R}(\{a, b\}, \Sigma)$ , si le support de  $r$  est contenu dans celui de  $na \times nb$ , alors les pôles de  $r$  sont soit tous les deux sur la « partie horizontale » du support, soit sur sa « partie verticale ».

**Théorème 7.** Soit  $\Sigma = \{+, -\}$ , pour toute  $A$ -représentation  $R$  de support  $K$ , pour tous les pôles  $x$  et  $y$  de  $R$ ,

$$(\overline{K, x, y}) \in \mathcal{F}(A, \Sigma)$$

(La démonstration est identique à celle que fait Shaw dans [13] page 57, dans le cadre d'une étude voisine.)

Par contre, le choix de  $\Sigma = \{+, -\}$  présente un inconvénient : par exemple, dans  $(\mathbb{R}^2, \mathbb{R}^2, T)$  avec les figures de figure 6, pour décrire  $(b \times a) + a$ , il faut faire  $b + (-a) + a + a$  (fig. 7) c'est-à-dire, décrire deux fois une des branches de la figure. Plus généralement, en utilisant  $\{+, -\}$  toute description de figures ayant au moins trois « branches », nous obligera à parcourir au moins une de ces branches deux fois.

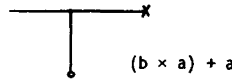


Figure 7

*Autre choix de  $\Sigma$*

Nous avons abouti à des résultats positifs pour

$$\Sigma = \{+, o, \times\} \quad \text{et} \quad \Sigma = \{+, -\}.$$

Si  $\Sigma' \subset \Sigma$  alors  $\mathcal{F}(A, \Sigma') \subset \mathcal{F}(A, \Sigma)$ ; donc pour tout ensemble d'opérateur plus grand que l'un des deux précédents, nous aurons les mêmes résultats positifs. Par contre, pour tout ensemble d'opérateurs plus petit que l'un des deux précédents, on montre que l'ensemble des supports de  $\mathcal{R}(A, \Sigma)$  est strictement contenu dans celui des  $A$ -représentations. Par exemple avec  $\Sigma = \{+, o\}$  on ne peut pas obtenir de figure dont le support soit celui de  $a \times b$  ( $a$  et  $b$  étant définis à la figure 6).

*Exemple d'utilisation :*

Nous allons reprendre l'exemple de [4] qui a décrit toutes les lettres latines manuscrites, à partir des quatre symboles de la figure 8.

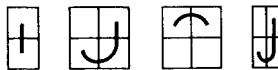


Figure 8

Cet alphabet de base est enrichi par application des symétries verticales ou horizontales, ou une des deux translations verticales de module 1. Chaque lettre est définie par une suite de ces symboles qui sont ensuite concaténées (fig. 9).

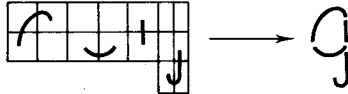


Figure 9

En utilisant, en plus des lissages et des raccordements de tangentes, il engendre ainsi tout l'alphabet.

Plaçons nous sur l'espace de figure défini à l'exemple 3 de 4.1 ( $\mathbb{R}^2$  muni des déplacements) et considérons l'alphabet  $\{ a, b, c, d, \langle \text{espace} \rangle \}$  défini à la figure 10 :

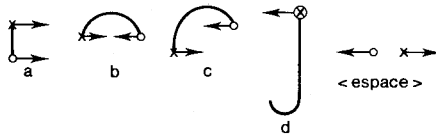


Figure 10

Nous allons ainsi construire par raccordement des tangentes les mêmes figures ; par exemple :  $b + ((-a + c) \circ d)$  donne la lettre « g » et  $b + ((-a + c) \circ d) + \langle \text{espace} \rangle + a + d + (-a) + b$  donne « gl » (fig. 11).

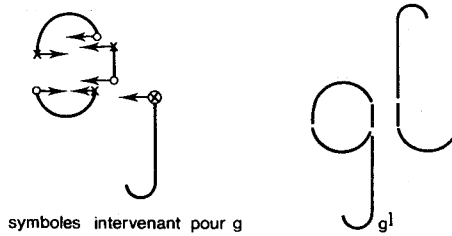


Figure 11

Une telle écriture ne suit pas cependant le mouvement de la main ; celui-ci correspondrait plutôt au schéma (fig. 12) introduit par l'expression

$$-c + ((c + c) \circ \lambda) + a + d$$

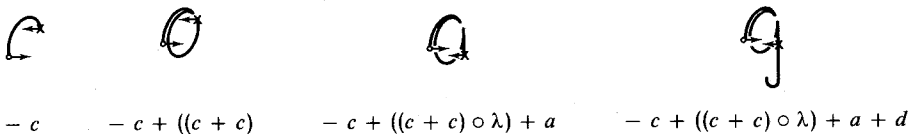


Figure 12

### 4.3. Longueurs d'une représentation, d'une figure

*Définition* : Soit  $(E, F, G)$  un espace de figures. On appelle *longueur d'une représentation*  $r = (K, o, e)$  et on note  $l(r)$  l'élément  $g$  de  $G$  tel que  $g(o) = e$ .

**Proposition 6.** Soit  $r_1$  une représentation et soit  $r_2 = g(r_1)$

$$l(r_2) = gl(r_1)g^{-1}.$$

*Définitions* : Soit  $f$  une figure sur  $(E, F, G)$ . On appelle longueur de  $f$  et on note  $l(f)$  l'ensemble  $\{gl(r)g^{-1} \mid g \in G, \bar{r} = f\}$ .

Soit  $0$  un point de  $F$ . On appelle longueur de  $f$  relative à  $0$  et on note  $l_0(f)$  la longueur de la représentation de  $f$  dont l'origine est en  $0$ .

$$\begin{aligned} l_0(f_1 + f_2) &= l_0(f_1) \cdot l_0(f_2) \\ l(f_1 \circ f_2) &= l(f_2) \\ l(f_1 \times f_2) &= l(f_1) \\ l(-f_1) &= l(f_1)^{-1} \end{aligned}$$

REMARQUE :  $+$  muni les figures sur  $(\emptyset, G, G)$  d'une structure de groupe : il suffit de remarquer que  $-f$  est le symétrique de  $f : f + (-f) = \lambda$ .

Remarquons alors que l'application  $l_0$  de  $\mathcal{F}$  dans  $G : l_0 : f \rightarrow l_0(f)$  est un isomorphisme de groupe.

### 4.4. Ensemble algébrique de figures

Soit  $\mathcal{F}$  un espace de figures. Selon le choix des opérateurs dont nous munissons  $\mathcal{F}$ , nous pouvons choisir plusieurs théories  $\Pi$ , pour lesquelles  $\mathcal{F}$  sera une  $\Pi$ -algèbre. Les éléments  $\omega$  de  $F_0$  de  $\Pi$  seront dans ce qui suit identifiés à leur réalisation  $\varphi(\omega)$ , aucune confusion n'étant possible.  $T$  désignera en particulier l'alphabet des figures de base et pourra désigner aussi  $F_0$ ; de même on identifiera les morphismes  $\omega \in M$  et leur réalisation  $\hat{\omega}$ .  $T'$  désignera  $\{-a \mid a \in T\}$  et  $\lambda$  désigne toujours l'élément neutre pour l'addition.

Considérons alors les théories engendrées par les ensembles suivants (dans tous les cas,  $F_i = \emptyset$  pour  $i > 2$ ) :

Théorie $\Pi_1$	$F_2 = \{+, \circ, \times\}$ $F_1 = \{-\}$ $F_0 = T$
Théorie $\Pi_2$	$F_2 = \{+, \circ\}$ $F_1 = \{-\}$ $F_0 = T$

Théorie $\Pi_3$	$F_2 = \{ +, \circ \}$ $F_1 = \emptyset$ $F_0 = T \cup T' \cup \{ \lambda \}$
Théorie $\Pi_4$	$F_2 = \{ +, \circ, \times \}$ $F_1 = \emptyset$ $F_0 = T$
Théorie $\Pi_5$	$F_2 = \{ + \}$ $F_1 = \{ - \}$ $F_0 = T$
Théorie $\Pi_6$	$F_2 = \{ + \}$ $F_1 = \emptyset$ $F_0 = T \cup T'$

Nous noterons  $\mathcal{A}_i$  la classe des ensembles de  $\mathcal{F}$  qui sont algébriques pour la théorie  $\Pi_i$  ( $i = 1, 2, 3, 4, 5, 6$ ).

Nous avons vu (4.2) que pour toute  $T$ -représentation  $R$ , il existe une représentation  $r$  de même support que  $R$  telle que  $\{ \bar{r} \} \in \mathcal{A}_i$  ( $i = 1, 2, 3, 4, 5, 6$ ). C'est ce qui justifie le choix de ces six structures algébriques.

Nous allons comparer  $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_5, \mathcal{A}_6$ .

**Théorème 6.**  $\mathcal{A}_1 = \mathcal{A}_2$

Il est clair que  $\mathcal{A}_2 \subset \mathcal{A}_1$ . La réciproque est fondée sur le résultat suivant :

$$\forall f_1, f_2 \in \mathcal{F} \quad f_1 \times f_2 = -((-f_2) \circ (-f_1)).$$

**Théorème 7.**  $\mathcal{A}_1 = \mathcal{A}_3$  ,  $\mathcal{A}_5 = \mathcal{A}_6$

Il est clair que :  $\mathcal{A}_3 \subset \mathcal{A}_2$

$$\mathcal{A}_6 \subset \mathcal{A}_5.$$

La réciproque s'appuie sur le lemme suivant :

**Lemme :** Pour tout morphisme  $\omega$  de  $\Pi_2$  (resp.  $\Pi_5$ ) il existe un morphisme  $\omega'$  de  $\Pi_3$  (resp.  $\Pi_6$ ) qui a même valeur d'interprétation que  $\omega$ .

Le lemme se démontre par récurrence sur le degré des morphismes ; par exemple :  $-(f_1 + f_2 \circ f_3) = (-f_3) + (f_2 \circ \lambda) + (-f_1)$ .

**REMARQUE :** Un groupe  $G$  est muni canoniquement d'une structure de  $\Pi_5$ -algèbre et d'une structure de  $\Pi_6$ -algèbre. Le même lemme peut être établi sur  $G$  et on démontre ainsi que la classe des ensembles algébriques sur la  $\Pi_5$ -algèbre  $G$  est égale à celle des ensembles algébriques sur la  $\Pi_6$ -algèbre  $G$ .

Énonçons encore deux résultats :

**Théorème 8.**  $\mathcal{A}_4 \subset \mathcal{A}_1$  ,  $\mathcal{A}_4 \neq \mathcal{A}_1$ .

**Théorème 9.**  $\mathcal{A}_4 \not\subset \mathcal{A}_6$  ,  $\mathcal{A}_6 \not\subset \mathcal{A}_4$ .

*Cas des figures fermées*

Dans  $\mathbb{R}^2$  muni des translations, soient  $a, b, c$  les figures décrites à la figure 13.

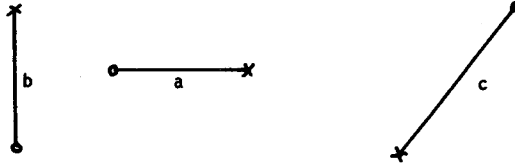


Figure 13

La figure  $na + nb + nc$  est une « figure fermée ».

On montre que l'ensemble des  $\{ na + nb + nc \mid n \geq 1 \}$  n'est pas algébrique dans la  $\Pi_1$ -algèbre.

Nous avons le même résultat pour les carrés de côté  $n$ .

Les résultats sont à rapprocher de ceux de [6] : en n'utilisant que l'opération  $+$  introduite ici, il montre que l'ensemble des courbes fermées convexes n'est pas algébrique.

## 5. LE PROBLEME DE LA RECONNAISSANCE

$\mathcal{G}$  désigne dans ce qui suit une grammaire sur une des théories que nous avons considérée précédemment (4.4).

$\mathcal{F} = (E, F, G)$  désigne un espace de figures et  $\mathcal{R}$  son ensemble de représentations.

Le problème de la reconnaissance est alors le suivant : existe-t-il un algorithme qui, étant donnée une grammaire  $\mathcal{G}$ , étant donnée une figure  $f$ , permette de décider si  $f \in L(\mathcal{G})$ ?

### 5.1. Résultats théoriques

Nous avons un premier résultat négatif :

**Théorème 10.** Le problème de la reconnaissance des figures est indécidable.

Pour montrer ce résultat, ramenons-nous au cas du groupe libre où ce résultat est connu.

Soit  $G$  un groupe ;  $(\emptyset, G, G)$  est un espace de figures qui, muni de l'addition, est isomorphe à  $G$  (cf. 4.3). Le problème de la reconnaissance se pose donc dans les mêmes termes que pour  $G$ , or il est indécidable dans le cas général sur un groupe (1).

Ce résultat n'est pas surprenant ; nous sommes placés ici dans un cas extrême où les supports vides ne pouvaient pas nous renseigner que la « composition de la figure » (c'est-à-dire sur l'arrangement des éléments de l'alphabet qui composent notre figure). Montrons que si on se donne la décomposition de la figure en éléments de l'alphabet, l'analyse syntaxique devient possible.

Cela revient à partir, non d'une figure ou d'une représentation, mais d'une  $T$ -représentation de  $T$ , (c'est-à-dire d'un graphe connexe dont les éléments sont des représentations de  $T$ , la relation binaire : avoir un pôle en commun, cf. 4.2). Le support d'une  $T$ -représentation est l'union des supports de ses éléments et son ensemble des pôles est l'ensemble des pôles de ses éléments.

Montrons d'abord par un exemple quel est le problème qui se pose.

Prenons comme espace de figures l'ensemble  $\mathbf{R}$  des réels muni du groupe des translations. Posons :

$$a = ([0, 1], 0, 1)$$

$$b = ([0, 3/2], 0, 3/2) \quad (\text{cf. figure 14}).$$

Soit  $\mathcal{G}_1$  la  $\Pi_3$ -grammaire dont l'unique règle est :  $X_1 \rightarrow a + a + a$  et soit  $\mathcal{G}_2$  la  $\Pi_3$ -grammaire dont l'unique règle est :  $X_2 \rightarrow b + b$ . Donnons-nous alors la  $T$ -représentation  $R$  décrite à la figure 15.

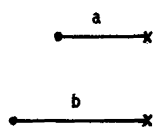


Figure 14



Figure 15

On peut alors se poser le problème de la reconnaissance de la façon suivante : existe-t-il une représentation  $r$  telle que le support de  $r$  soit celui de  $R$ , telle que l'origine et l'extrémité de  $r$  soient des pôles de  $R$  et telle que  $\bar{r} \in L(\mathcal{G})$ ? Dans ce cas, la réponse est positive pour nos deux grammaires  $\mathcal{G}_1$  et  $\mathcal{G}_2$ . Mais si on impose que la « décomposition » de  $r$  sur l'alphabet soit la même que  $R$ , alors seule  $\mathcal{G}_1$  convient.

(1) Ce résultat est bien connu sur les groupes où on démontre même des résultats plus puissants, voir par exemple l'appendice de Shoenfield [20]. Une démonstration plus proche des méthodes utilisées en théorie des langages peut être trouvée dans [13].



C'est ce point que nous allons préciser.

Soit  $\mathcal{G} = (N, ::=, X)$  une  $\Pi_3$ -grammaire.  $\mathcal{B}(\mathcal{G})$  est l'ensemble des structures associées à  $\mathcal{G}$  et pour un élément  $s \in \mathcal{B}(\mathcal{G})$ ,  $\Omega(s)$  est une figure (cf. 3.2).

Soit 0 un point de l'espace des pôles  $F$ . Nous allons associer à toute ramification  $s$  de  $\mathcal{B}(\mathcal{G})$  une  $T$ -représentation  $\mathcal{T}_0(s)$  :

- si  $s$  est de taille 1, alors  $\mathcal{T}_0(s)$  est la représentation de  $\Omega(s)$  d'origine 0
- si  $s = p (s_1 + \dots + s_q)$ , distinguons suivant le second membre  $\omega$  de la règle  $p$  :

- si  $\omega \in N$  alors  $q = 1$  et  $\mathcal{T}_0(s) = \mathcal{T}_0(s_1)$
- si  $\omega = A \circ B$  alors  $q = 2$  et  $\mathcal{T}_0(s) = \mathcal{T}_0(s_1) \cup \mathcal{T}_0(s_2)$
- si  $\omega = A + B$  alors  $q = 2$  et  $\mathcal{T}_0(s) = \mathcal{T}_0(s_1) \cup l_0(\Omega(s_1))(\mathcal{T}_0(s_2))$

où  $l_0(\Omega(s_1))$  est la longueur de  $\Omega(s_1)$  relativement au point 0 (3.2).

$\mathcal{T}_0(s)$  est une  $T$ -représentation associée à la structure  $s$ .

Soit par exemple,  $\mathcal{G}_1$  définie par :

$$\begin{aligned} 1 : X & ::= A + B \\ 2 : B & ::= a \\ 3 : A & ::= B + B \end{aligned}$$

$\mathcal{G}'_1$  est une grammaire  $C$ -réduite de  $\mathcal{G}'_1$ .

La ramification  $s = 1 \times (3 \times (2 + 2) + 2)$  (fig. 16) appartient à  $\mathcal{B}(\mathcal{G}'_1)$ .

Si on prend pour 0 le zéro de  $\mathbf{R}$ ,  $\mathcal{T}_0(s)$  est représentée à la figure 15.

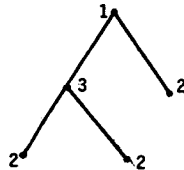


Figure 16

$\mathcal{T}(s)$  est l'ensemble des  $T$ -représentations associées à la structure  $s$  :  $\mathcal{T}(s) = \{ \mathcal{T}_0(s) \mid 0 \in F \}$ .

De même, on définit  $\mathcal{R}_0(s)$  comme une représentation de  $\Omega(s)$  qui a pour origine 0.

La question que nous posons maintenant est : étant donnée une représentation  $R$ , existe-t-il une structure  $s \in \mathcal{B}\mathcal{L}(\mathcal{G}, X)$  telle que  $\mathcal{T}(s)$  contienne  $R$ ? Quels sont tous les éléments de  $\mathcal{B}\mathcal{L}(\mathcal{G}, X)$  vérifiant cette propriété?

La réponse est donnée par :

**Théorème 11.** Il existe un algorithme qui, pour toute  $T$ -représentation  $R$ , pour tout couple de pôle  $i$  et  $j$  de  $R$  permet de construire tous les éléments  $r$

de  $\mathcal{B}\mathcal{L}(\mathcal{G}, X)$  tels que  $R = \mathcal{T}_i(r)$ , et dont l'extrémité de la représentation  $\mathcal{R}_i(r)$  soit  $j$ .

L'algorithme est une adaptation de l'algorithme d'analyse globale ascendante de [21]; il construit une grammaire donnant toutes les ramifications  $r$  de  $\mathcal{B}(\mathcal{G})$  telles que pour un  $i$ ,  $R \supset \mathcal{T}_i(r)$ . Ensuite il reste à déterminer celles dont la  $T$ -représentation associée est exactement  $R$ , c'est-à-dire celles dont le mot des feuilles contient au moins une occurrence de chaque élément de  $R$  ce qui nous ramène à l'intersection d'un langage régulier et d'un langage à contexte libre.

Ce résultat, positif, est cependant théorique car nous partons d'une  $T$ -représentation, c'est-à-dire qu'une représentation où tous les morceaux élémentaires ont déjà été identifiés; d'autre part il est très lent.

## 5.2. Un algorithme d'analyse syntaxique

L'algorithme d'analyse syntaxique précédent a pour donnée l'ensemble de tous les constituants élémentaires de la figure à analyser; or réaliser ce découpage de prétraitement suppose l'existence d'une procédure capable d'extraire toutes ces formes élémentaires; une telle procédure est difficile à réaliser et il serait souhaitable que l'analyse syntaxique guide une telle procédure. Nous supposons simplement l'existence d'une procédure de reconnaissance pour chaque forme élémentaire et dont le seul paramètre est l'origine de la représentation que l'on suppose présente.

En analysant une phrase d'un langage, on parcourt la phrase de gauche à droite et chaque élément de la phrase n'est pris en considération qu'une fois. Nous allons voir sur un exemple que ce n'est pas toujours le cas ici; dans  $\mathbf{R}^2$  muni des translations considérons le système suivant :

$$A = a + A \cup a + B \cup a$$

$$B = b + B \cup b + C$$

$$C = -a + C \cup -a + D$$

$$D = -b + D \cup -b + A$$

La figure 17 donne un élément de la solution :

$$a + (a + b) + (-a) + (-b) + a \quad \text{mais qui est aussi} \\ a + n(a + b + (-a) + (-b)) + a;$$

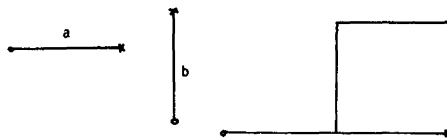


Figure 17

ce qui signifie que lors de l'analyse on peut parcourir un nombre arbitrairement grand de fois la représentation.

Nous ne nous préoccupons pas de ce problème ici : il y a deux manières simples de la résoudre : soit fixer un nombre maximum de parcours de l'analyseur sur une même représentation, soit se restreindre arbitrairement en supposant que ce cas ne se présente jamais.

Ceci nous amène à donner une définition qui va nous servir par la suite.

Soit  $\mathcal{G} = (N, ::=, X)$  une grammaire. Soit 0 un point de l'espace des pôles nous pouvons associer à chaque structure  $s \in \mathcal{B}\mathcal{L}(\mathcal{G}, X)$  une  $T$ -représentation  $\mathcal{T}_0(s)$ . Pour toute  $T$ -représentation  $r$  nous noterons  $P(r)$  son ensemble des pôles.

Soit  $s = p \times (s_1 + s_2)$  une structure. Nous dirons que  $s$  se superpose par les pôles si

- ou  $s_1$  se superpose par les pôles
- ou  $s_2$  se superpose par les pôles
- ou si la règle de numéro  $p$  est  $A ::= B \circ C$  alors

$$P(\mathcal{T}_0(s_1)) \cap P(\mathcal{T}_0(s_2)) \neq \{0\}$$

ou si la règle de numéro  $p$  est  $A ::= B + C$  alors

$$P(\mathcal{T}_0(s)) \cap P(l_0(\Omega(s_1))(\mathcal{T}_0(s_2))) \neq \{l_0(\Omega(s_1))(0)\}.$$

Une structure se superpose donc par les pôles si lors de la construction associée à  $s$  deux pôles se superposent sans que cette superposition soit explicitement indiquée par les opérations.

Nous dirons que  $f$  ne se superpose pas par le pôle si pour toute structure  $s \in \mathcal{B}\mathcal{L}(\mathcal{G}, X)$  telle que  $\Omega(s) = f$ ,  $s$  ne se superpose pas par les pôles.

Pour simplifier les énoncés des algorithmes nous nous restreindrons à des grammaires  $C$ -réduites sur la théorie  $\Pi_3$ ; ceci nous oblige en particulier à autoriser la figure vide  $\lambda$  (cf. 4.4). Tout second membre de règle sera donc de la forme  $A, A + B, A \circ B$  ou  $a$ , où  $A$  et  $B$  désignent des éléments de l'alphabet non terminal  $N$ , et  $a$  désigne une figure de l'alphabet terminal  $T$  (dans  $T$  on fait aussi figurer  $\lambda$  et  $T$  vérifie la propriété suivante : si  $a \in T, \neg a \notin T$ ). Tout ce qui est dit ici se généralise sans difficultés au cas où la grammaire n'est pas  $C$ -réduite. Mais si nous nous plaçons sur la théorie  $\Pi_1$  ou  $\Pi_2$ , c'est-à-dire si nous introduisons en plus les opérateurs  $\times$  ou  $-$ , nous serons contraints à introduire une procédure analysant de l'origine vers l'extrémité et une autre analysant de l'extrémité vers l'origine, alors que seule la première est nécessaire ici.

La grammaire pour laquelle nous écrivons l'algorithme d'analyse syntaxique est donnée par :

$$\text{pour } i = 1, \dots, n \quad A_i ::= \omega_{i_1} \mid \dots \mid \omega_{i_j}.$$

La procédure que nous considérons effectue l'analyse d'une représentation à partir d'un pôle  $p$ , cette représentation étant supposée appartenir à la catégorie syntaxique  $A$ . Si l'analyse a été menée avec succès, son résultat est l'extrémité  $q$  de cette représentation. C'est une procédure indéterministe<sup>(1)</sup> à choix et l'analyse ne se solde par un échec que si tous les choix possibles mènent à une *erreur* (ici erreur a plutôt la signification d'impasse).

Analyse ( $A, p$ ) :

si  $A \in N$  alors début choix d'une règle  $A ::= \omega$  ;  
     si  $\omega$  est de la forme  $B + C$  alors début  $q = \text{Analyse}(B, p)$  ;  
         Analyse := Analyse ( $C, q$ ) fin ;  
     si  $\omega$  est de la forme  $B \circ C$  alors début Analyse ( $B, p$ ) ;  
         Analyse := Analyse ( $C, p$ ) fin ;  
     si  $\omega = B$  alors Analyse := Analyse ( $B, p$ )  
     fin  
 si  $A \in T$  alors début s'il existe une représentation de  $A$  d'origine  $p$  et d'extrémité  $q$  alors Analyse :=  $q$  sinon erreur fin

Pour que cet algorithme se termine il faut, comme dans le cas de l'analyse syntaxique descendante habituelle, que la grammaire ne soit pas récursive à gauche. Cette condition devient aussi suffisante si l'on ne s'autorise, lors de l'analyse à ne parcourir qu'un nombre fini de fois chaque représentation de  $T$  (la démonstration est la même que celle faite pour les langages [15]).

Remarquons que si  $a \in T$ , Analyse ( $a, p$ ) est une procédure qui se borne à vérifier la présence d'une représentation de  $a$  d'origine  $p$  ; et c'est dans ce sens que nous pouvons dire que la *reconnaissance est guidée par l'analyse syntaxique*.

#### Réduction de l'indéterminisme

L'inconvénient d'une telle procédure indéterministe est de nous obliger à de nombreux retours arrière en cas d'erreur. En procédant à des tests lors de nos choix nous pouvons réduire l'indéterminisme.

Définissons plusieurs relations sur  $N \cup T$ .

La relation  $\mathfrak{J}$  nous permet de trouver quelles sont les procédures de reconnaissance d'un terminal qui seront activées les premières lors de l'appel d'Analyse ( $A, p$ ) pour un non terminal  $A$  donné. L'appel d'Analyse ( $A, p$ ) va conduire à l'appel d'Analyse ( $B, p$ ) pour un certain  $B$  tel que :

$$\text{soit } A ::= B \quad (1)$$

$$\text{soit } A ::= B \circ C \quad (2)$$

$$\text{soit } A ::= B + C \quad (3)$$

(1) La programmation classique de procédures indéterministes se fait soit en essayant successivement tous les choix et en faisant un retour arrière en cas d'erreurs, soit en traitant tous les cas en parallèle.

Si  $B = \lambda$  il n'y a pas en fait de reconnaissance dans le cas (1) et dans les cas (2) et (3) l'appel d'Analyse  $(B, p)$  sera alors suivi de celui d'Analyse  $(C, p)$ ;

il peut en être de même si  $B \succ^* \lambda$ .

Notons donc :  $A \exists B \Leftrightarrow B = N \cup T - \{\lambda\}$  et  $A ::= B$

ou  $(\exists C \in N \text{ et } (A ::= B + C \text{ ou } A ::= B \circ C))$

ou  $(\exists C \in N \text{ et } C \succ^* \lambda \text{ et } (A ::= C = B \text{ ou } A ::= C \circ B).$

Nous appellerons ensemble initial de  $A$  et nous noterons  $\exists n(A)$  l'ensemble  $\exists n(A) = \{a \in T - \{\lambda\} \mid A \exists^+ a\}$ .

(Par définition,  $\exists^+$  est la fermeture transitive de  $\exists$ . Si  $\exists^+$  admet un circuit, la grammaire est récursive à gauche.)

L'origine de chaque figure qui dérive de  $A$  peut être origine de plusieurs représentations terminales; c'est en particulier le cas si on applique la règle  $A ::= B \circ C$ ; nous allons définir l'ensemble  $\mathcal{O}(A)$  des familles de terminaux qui peuvent être placés à l'origine d'une figure dérivant de  $A$ .

Auparavant, nous allons introduire une notation supplémentaire.

Si  $T$  est un ensemble, nous noterons  $p(T)$  l'ensemble de ses parties. Nous noterons  $p^2(T)$  l'ensemble  $p(p(T))$ .

Sur  $p^2(T)$  définissons l'opération :

$$A \dot{\cup} B = \{a \cup b \mid a \in A \text{ et } b \in B\}$$

en particulier  $A \dot{\cup} \{\emptyset\} = A$ .

Posons alors :

$$\text{pour } A \in N \quad \mathcal{O}(A) = \bigcup_{A ::= \omega} \mathcal{O}(\omega)$$

$$\mathcal{O}(B + C) = \mathcal{O}(B) \cup (\text{si } B \succ^* \lambda \text{ alors } \mathcal{O}(C) \text{ sinon } (\emptyset))$$

$$\mathcal{O}(B \circ C) = \mathcal{O}(B) \dot{\cup} \mathcal{O}(C)$$

$$\text{si } a \in T : \quad \mathcal{O}(a) = \{\{a\}\}$$

$$\mathcal{O}(\lambda) = \{\emptyset\}$$

Nous aurons par la suite surtout besoin de la définition de  $\mathcal{O}(\omega)$  lorsque  $\omega$  est un second membre de règle.

Soit  $s$  une structure,  $s \in \mathcal{BL}(\mathcal{G}, A)$ ; soit  $p$  un point de l'espace des pôles; on note  $\mathcal{O}(s)$  l'ensemble des représentations de  $T - \{\lambda\}$  d'origine  $p$  qui appartiennent à la  $T$ -représentation  $\mathcal{C}_p(s)$ .

On démontre alors facilement les propositions suivantes :

**Proposition 1.** Soit  $s \in \mathcal{BL}(\mathcal{G}, A)$  une structure et soit  $p$  un point de l'espace des pôles. Si  $s$  ne se superpose pas par les pôles alors  $\mathcal{O}(s) \in \mathcal{O}(A)$ .

**Proposition 2.** Si l'appel d'Analyse  $(A, p)$  conduit à un succès, si  $Up$  désigne l'ensemble des représentations d'origine  $p$  que l'analyse a reconnues, alors il existe  $V \in \mathcal{O}(A)$  tel l'ensemble  $Vp$  des représentations de  $V$  d'origine  $p$  vérifie  $Vp \subset Up$ .

**Proposition 3.**  $\forall a \in \mathcal{Jn}(A)$  ,  $\exists V \in \mathcal{O}(A)$  ,  $a \in V$

**Proposition 4.**  $\forall V \in \mathcal{O}(A)$   $(V \cap \mathcal{Jn}(A) = \emptyset \Leftrightarrow V = \emptyset)$

de plus, si  $A$  ne dérive pas  $\lambda$ , alors

$$\forall \perp \in \{ +, \circ \} , \forall V \in \mathcal{O}(A \perp B) \quad V \cap \mathcal{Jn}(A) \neq \emptyset.$$

En conséquence, dans la procédure analyse  $(A, p)$ , le choix d'une règle  $A ::= \omega$  ne doit se faire que s'il existe un élément de  $\mathcal{O}(\omega)$  dont toutes les représentations d'origine  $p$  sont présentes dans l'image à analyser (proposition 2).

Un test plus faible est déduit de la proposition 4 : si de  $B$  ne dérive pas le mot vide alors on ne choisit une règle  $A ::= B \perp C$  que si une représentation d'un élément de  $\mathcal{Jn}(B)$  apparaît aux pôles  $p$ ; et si de  $B$  dérive le mot vide ce choix ne doit se faire que si une représentation d'un élément de  $\mathcal{Jn}(B) \cup \mathcal{Jn}(C)$  apparaît au pôle  $p$ .

Le test sur  $\mathcal{O}(\omega)$  réduit évidemment plus l'indéterminisme que le test sur  $\mathcal{Jn}(B)$ , cependant il demande un appel plus nombreux des procédures de reconnaissance, alors que le test sur les initiales ne demande pas plus d'appels de procédure de reconnaissance que le programme d'analyse syntaxique lorsque tous les choix sont faits successivement. Si le temps d'exécution de ces procédures est long, on préférera donc faire le test le plus faible et traiter plus de retours arrières; et dans le cas contraire on testera sur  $\mathcal{O}(\omega)$  (ce peut être en particulier le cas lorsqu'on opère non plus sur des figures mais sur des graphes que l'on peut munir d'une structure analogue [18]).

#### *Amorce de l'étude du déterminisme*

Le mot vide joue un rôle particulier dans l'étude des langages et les figures dont l'origine et l'extrémité sont confondues (c'est-à-dire dont la longueur est l'élément neutre, ce que nous notons  $l(f) = 1$ ), jouent un rôle analogue.

Nous allons transformer la grammaire en faisant apparaître les non-terminaux ( $N''$ ) d'où ne dérivent que des figures dont l'origine et l'extrémité sont confondues, et les non-terminaux ( $N'$ ) où ceci ne se produit pas si les figures ne se superposent pas par les pôles, ce que nous supposons.

Construisons une grammaire  $\mathcal{G}'$  à partir de  $\mathcal{G}$  :

si dans $\mathcal{G}$ nous avons les règles	Alors dans $\mathcal{G}'$ nous avons les règles
$A ::= B + C$	$A' ::= B' + C' \mid B'' + C' \mid B' + C''$
$A ::= B \circ C$	$A' ::= B' \circ C' \mid B'' \circ C'$
$A ::= B$	$A' ::= B'$
	$A'' ::= B''$
$A ::= a \quad (a \in T)$	si $l(a) = 1$ alors $A' ::= a$ sinon $A' ::= a$

Si on autorise  $\mathcal{G}'$  à avoir deux axiomes  $X'$  et  $X''$ ,  $\mathcal{G}'$  engendre le même langage que  $\mathcal{G}$  et, à une transcription près ( $A' \rightarrow A$ ,  $A'' \rightarrow A$ ), les structures de  $\mathcal{G}'$  sont celles de  $\mathcal{G}$ .

Remarquons que  $A'' \succ^* f$  entraîne  $l(f) = 1$ , mais que la réciproque n'est vraie que si  $f$  ne se superpose pas par ses pôles, ce qui est supposé ici.

Remarquons enfin que si  $l(f) = 1$  alors  $f + g = f \circ g$ .

Reprenons alors la définition  $\mathcal{O}$  en distinguant l'ensemble des origines de figures de longueur 1 ( $\mathcal{O}_2$ ) et celui des origines des figures de longueur différentes ( $\mathcal{O}_1$ ).

Posons : pour  $A \in N$  ,  $\mathcal{O}_1(A) = \mathcal{O}(A')$   $A' \in N'$   
 $\mathcal{O}_2(A) = \mathcal{O}(A'')$   $A'' \in N''$

où  $\mathcal{O}$  est définie comme précédemment, mais sur la grammaire  $\mathcal{G}'$ .

Nous vérifions que  $\mathcal{O}(A) = \mathcal{O}_1(A) \cup \mathcal{O}_2(A)$ .

Nous supposons l'existence d'une procédure cherche ( $p$ ) dont le paramètre  $p$  est un pôle et le résultat est l'ensemble des terminaux ayant dans l'image une représentation d'origine  $p$ .

La procédure d'analyse comportera cette fois un paramètre supplémentaire : l'ensemble  $V$  des représentations à prendre en considération au pôle  $p$ . Lors de l'appel de Analyse ( $A, V, p$ ) la procédure choisit une règle  $A ::= \omega$  telle que  $\mathcal{O}(\omega) = V$ ; puis elle rappelle les procédures correspondant à  $\omega$ . Si  $\omega$  est de la forme  $B \circ C$  alors, l'origine de la représentation dérivant de  $B$  coïncidant avec celle de  $C$ , il faut choisir  $X$  et  $Y$  appartenant à  $\mathcal{O}(B)$  et  $\mathcal{O}(C)$  tel que  $X \cup Y = V$  et reprendre l'analyse de  $B$  avec  $X$  et celle de  $C$  avec  $Y$ ; le même cas se produit si  $\omega = A + B$  et si de  $A$  dérive une figure de longueur 1. Ceci nous amène en particulier à introduire une procédure supplémentaire qui n'analyse que les figures de longueur 1; nous l'appellerons  $\lambda$  Analyse.

La procédure normale d'analyse retourne toujours comme résultat l'extrémité  $q$  de la figure analysée.

Si  $p$  est choisi pour le début de l'analyse (et si  $X$  est l'axiome de  $G$ ) l'appel initial sera

Analyse ( $X$ , cherche ( $p$ ),  $p$ ).

La procédure analyse étant :

Analyse ( $A$ ,  $V$ ,  $p$ )

si  $A \in N$  : choix d'une règle  $A ::= \omega$  telle que  $V \in \mathcal{O}(\omega)$ ;

si  $\omega = B + C$  alors début choix (choix de  $X \in \mathcal{O}_2(B)$  et  $Y \in \mathcal{O}(C)$   
tels que  $X \cup Y = V$ ;  $\lambda$  Analyse ( $B$ ,  $X$ ,  $p$ );  
Analyse := Analyse ( $C$ ,  $Y$ ,  $p$ ),  
(choix de  $X \in \mathcal{O}_1(B)$  tel que  $X = V$ ;  
 $q =$  Analyse ( $B$ ,  $X$ ,  $p$ );  
Analyse := Analyse ( $C$ , cherche ( $q$ ),  $q$ ))

fin

si  $\omega = B \circ C$  alors début choix de  $X$  et  $Y$ ,  $X \in \mathcal{O}(B)$ ,  $Y \in \mathcal{O}(C)$  tels que  
 $X \cup Y = V$ ; Analyse ( $B$ ,  $X$ ,  $p$ );  
Analyse := Analyse ( $C$ ,  $Y$ ,  $p$ )

fin

si  $\omega = B$  alors Analyse := Analyse ( $B$ ,  $V$ ,  $p$ )

si  $A \in T$  : si (il existe une représentation de  $A$  d'origine  $p$ , de longueur 1  
alors (soit  $q$  son extrémité, Analyse :=  $q$ )  
sinon erreur

la procédure  $\lambda$  Analyse est peu différente :

$\lambda$  Analyse ( $A$ ,  $V$ ,  $p$ )

si  $A \in N$  : choix d'une règle  $A ::= \omega$  telle que  $V \in \mathcal{O}_2(\omega)$ ;

si  $\omega = B + C$  alors début choix de  $X \in \mathcal{O}_2(B)$ , de  $Y \in \mathcal{O}_2(C)$   
tels que  $X \cup Y = V$ ;  
 $\lambda$  Analyse ( $B$ ,  $X$ ,  $p$ );  $\lambda$  Analyse ( $C$ ,  $Y$ ,  $p$ )

fin

si  $\omega = B \circ C$  alors début choix de  $X \in \mathcal{O}(B)$ , de  $Y \in \mathcal{O}_2(C)$   
tels que  $X \cup Y = V$ ;  
Analyse ( $B$ ,  $X$ ,  $p$ );  $\lambda$  Analyse ( $C$ ,  $Y$ ,  $p$ )

fin

si  $\omega = B$  alors  $\lambda$  Analyse ( $B$ ,  $V$ ,  $p$ )

si  $A \in T$  : si  $\neg$  (il existe une représentation de  $A$  d'origine  $p$ , de longueur 1  
et  $\{A\} = V$ ) alors erreur

On en déduit une condition suffisante de déterminisme :

$D_1 : A ::= \omega, A ::= \omega' \text{ et } \omega \neq \omega' \Rightarrow \mathcal{O}(\omega) \cap \mathcal{O}(\omega') = \emptyset$

$D_2 : A ::= B \circ C \Rightarrow (\forall X \in \mathcal{O}(B), \forall Y \in \mathcal{O}(C), X \cap Y = \emptyset$

$D'_2 : A ::= B + C \Rightarrow (\forall X \in \mathcal{O}_2(B), \forall Y \in \mathcal{O}(C), X \cap Y = \emptyset)$

$D_3 : A ::= B + C \Rightarrow (\forall X \in \mathcal{O}_1(B), \forall X' \in \mathcal{O}_2(B), \forall Y \in \mathcal{O}(C), X \neq X' \cup Y)$



En effet ces conditions ne permettent pour chacun des choix qu'une seule possibilité au plus.

Ces conditions ne sont pas trop restrictives et sont voisines des conditions de déterminisme des grammaires LL(1). En particulier la condition  $D_1$  est souvent satisfaite par factorisation;

par exemple, si  $A ::= a + A \mid a$

on peut très bien écrire :

$$A ::= a + B$$

$$B ::= A \mid \lambda.$$

Par contre, en supposant que les structures de nos grammaires ne se superposent pas par les pôles, nous faisons une hypothèse plus restrictive.

Si par exemple nous voulons décrire des carrés, nous pouvons considérer la grammaire suivante :

$$X ::= A \circ (B + C)$$

$$A ::= a + A + b \mid a + b$$

$$B ::= b + B \mid b$$

$$C ::= a + C \mid a$$

qui engendre les  $(na + nb) \circ (mb + pa)$  (fig. 18)  $m > 0, n > 0, p > 0$

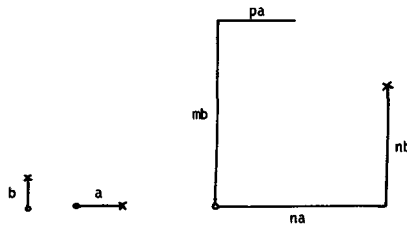


Figure 18

Et si nous voulons obtenir des figures fermées nous rajouterons une condition supplémentaire : la coïncidence de l'extrémité de  $C$  et de celle de  $A$ ; il est clair que dans un tel cas les structures se superposent par les pôles exactement là où nous imposons la superposition.

La question est aussi posée de savoir s'il est possible de déterminer si une grammaire possède ou non une structure qui se superpose par les pôles.

## CONCLUSION

Nous avons montré que l'analyse syntaxique, idée fondamentale mais simple, peut se généraliser sans trop de difficultés à des structures plus complexes que le monoïde libre.

Dans notre exemple sur les figures, il est clair que cette idée est nettement

insuffisante pour résoudre le problème de la reconnaissance des formes. D'abord une figure qui n'est pas un assemblage de lignes (ou de surfaces manipulées comme une ligne : par raccordement bout à bout) ne se décrit pas par une structure algébrique; il faudrait au moins y ajouter des relations telle que la relation topologique « à l'intérieur de ». Par ailleurs certaines figures n'admettent pas de description structurée et donc par là aucune analyse syntaxique n'est possible (par exemple pour la photographie de nuages).

Une autre difficulté vient du risque d'erreur lors de la reconnaissance d'un constituant syntaxique. Pour cela il faudrait introduire une reconnaissance avec une quantification du risque d'erreurs, munir les règles d'un degré de probabilité d'utilisation dans un certain contexte et on pourrait ainsi obtenir une reconnaissance avec un degré de vraisemblance.

Une telle méthode est actuellement mise en œuvre pour la reconnaissance de la parole, (dans ce cas la structure algébrique est à nouveau celle du monoïde libre) et les premiers résultats semblent prometteurs [8].

#### BIBLIOGRAPHIE

- [1] BAR-HILLEL Y., PERLES M. and SHAMIR E., *On formal properties of simple phrase structure grammars-Zeitschrift für Phonetik, Sprachwissenschaft und Kommunikationsforschung*, 14, p. 143-172 (1961).
- [2] BRUN J. M. et THERON M., *EUCLID, langage graphique tridimensionnel*, Journées graphiques de l'IRIA (1972).
- [3] DUCROT A. et LEMAIRE A. *METAVISU, un système graphique interactif*, Rapport de recherche IRIA n° 36 (1973).
- [4] EDEN M., *Handwriting and pattern recognition*, IRE Trans. IT, 8, p. 160-166 (1962).
- [5] EILENBERG S. and WRIGHT J. B., *Automata in general algebras*, IBM Research Note NC 725 (1967).
- [6] FEDER J., *Languages of encoded line patterns*, Information and Control, 13, p. 230-244 (1968).
- [7] FU S. K., *Syntactic Methods in Pattern Recognition*, Academic Press New-York, London (1974).
- [8] HATON J. P. et PIERREL J. M., *Une approche syntaxique de la reconnaissance de phrase dans un contexte donné*, Compte Rendu des 6<sup>e</sup> journées d'étude sur la parole (G.A.L.F.), Toulouse, mai 1975.
- [9] KIRSCH R., *Computer interpretation of english text and pictures patterns IEEE Trans. on Electronic Computers*, EC-13, 4, pp. 363-376 (1964).
- [10] KREISEL G. et KRIVINE J. L., *Éléments de la logique mathématique, Théorie des modèles*, Dunod, Paris (1967).
- [11] LESCANNE P., *Etude de quelques théories des langages et généralisation du théorème de Kleene*, Thèse de spécialité, Université de Nancy 1 (1971).

- [12] LEYDLEY R. S., *High-speed automatic analysis of biomedical pictures*, Sciences, 146, pp. 216-223 (1964).
- [13] MOHR R., *Modèles algébriques pour l'analyse syntaxique de figures*, Thèse de spécialité, Université de Nancy 1 (1973).
- [14] OTA P. G., *Mosaic grammars*, University of Pennsylvania, Moore School, Report n° 73-10.
- [15] PAIR C., *Etude de l'analyse syntaxique*, Cours de l'école d'Eté CEA, EDF, IRIA, Le Breau, septembre 1973.
- [16] PAIR C. et QUERE A., *Définition et étude des bilangages réguliers*, Information and Control, 13, p. 565-593 (1968).
- [17] QUERE A., *Etude des ramifications et des bilangages*, Thèse de spécialité, Faculté des Sciences, Université de Nancy 1 (1969).
- [18] SHAW A. C., *The formal description and parsing of pictures*, Ph. D. Thesis Comp. Sc. Dept. Stanford University (1968).
- [19] SHAW A. C., *Parsing of graph-representable pictures*, J. Assoc. Comp. Mach., 17, n° 3, p. 453-481 (1970).
- [20] SHOENFIELD J. R., *Mathematical Logic*, Addison Wesley (London) (1967).
- [21] YOUNGER D. M., *Recognition and parsing of context-free languages in time  $n^3$* , Information and Control, 10, p. 189-208 (1967).