RAIRO. INFORMATIQUE THÉORIQUE

MICHÈLE SORIA

Une extension des langages déterministes

RAIRO. Informatique théorique, tome 17, nº 4 (1983), p. 301-319

<http://www.numdam.org/item?id=ITA_1983__17_4_301_0>

R.A.I.R.O. Informatique théorique/Theoretical Informatics
(vol. 17, n° 4, 1983, p. 301 à 319)

UNE EXTENSION DES LANGAGES DÉTERMINISTES (*)

par Michèle SORIA (1)

communiqué par J.-F. PERROT

Résumé. — *Nous définissons dans cet article une famille de langages algébriques, appelés quasi déterministes, qui s'insère strictement entre la famille des langages déterministes et celle des langages non ambigus. Cette famille est fermée par passage au complémentaire, intersection avec un langage rationnel, morphisme inverse... Elle n'est pas fermée par image miroir, mais on sait caractériser les langages dont l'image miroir est quasi déterministe. Nous prouvons aussi un théorème d'itération, qui fournit une condition nécessaire pour qu'un langage soit quasi déterministe.*

Abstract. — *In this article, we define a family of context free languages, the quasi deterministic languages, which is strictly inserted between the families of deterministic and non ambiguous context free languages. The family is closed under complement, intersection with a rational set, inverse homomorphism... On the contrary, it is not closed under mirror image but the languages the mirror image of which is quasideterministic, can be characterized. We also prove an iteration theorem giving a necessary condition for a language to be quasideterministic.*

INTRODUCTION

L'étude de la famille des langages algébriques, ou context-free, a mis en évidence certaines sous-familles strictes possédant des propriétés particulières : langages linéaires, à un compteur, déterministes, non ambigus...

La famille des langages déterministes — notée Det — et celle des langages non ambigus — notée NA — se distinguent des autres par leur structure : elles sont fermées par les opérations de morphisme inverse et d'intersection avec un langage rationnel, mais leur clôture par morphisme direct donne la famille des langages algébriques toute entière.

Les langages non ambigus sont ceux que l'on peut engendrer de telle sorte que tout mot du langage n'admette qu'un seul arbre de dérivation; de là vient la propriété de fermeture par image miroir de la famille NA.

Les langages déterministes — i. e. ceux reconnu par des automates à pile dont le mouvement est déterminé à tout instant — sont les langages algébriques

(*) Reçu janvier 1981, révisé avril 1983.

(1) Laboratoire de Recherche en Informatique, Bât. n° 490, Orsay 91405 Cedex.

R.A.I.R.O. Informatique théorique/Theoretical Informatics, 0399-0540/1983/301/$ 5.00

les plus adéquats à la mise en œuvre de procédés simples d'analyse syntaxique. Det est de plus la seule famille de langages algébriques fermée par l'opération de passage au complémentaire.

La famille des langages déterministes est strictement incluse dans celle des langages non ambigus. Le caractère strict de cette inclusion est, par exemple, établi par le fait que Det n'est pas fermée par image miroir : par exemple le langage :

$$L = \{ ca^n \& a^n \mid n \geqq 1 \} \cup \{ da^n \& a^{2n} \mid n \geqq 1 \}$$

est déterministe mais, à cause du sens de lecture de gauche à droite des automates à pile, le langage :

$$\tilde{L} = \{ a^n \& a^n c \mid n \geqq 1 \} \cup \{ a^{2n} \& a^n d \mid n \geqq 1 \}$$

n'est pas déterministe, bien que les mots de $\tilde{L}$ contiennent eux aussi l'information (existence d'un c ou d'un d) qui permettait l'analyse déterministe des mots de L. Dans la classification existante des sous-familles de langages algébriques, le langage $\tilde{L}$ est donc catalogué comme langage non ambigu au même titre que, par exemple, le langage :

$$L' = \{ a^n \& a^n \mid n \geqq 1 \} \cup \{ a^n \& a^{2n} \mid n \geqq 1 \}$$

ne contenant, lui, aucune information qui permettrait le déterminisme des calculs d'un automate à pile.

Il paraît naturel de différencier les langages $\tilde{L}$ et L' et donc de chercher à insérer entre Det et NA une famille de langages, jouissant si possible des propriétés principales de ces deux familles, qui élargisse la notion de déterminisme à tous les langages non ambigus qui contiennent un certain type d'information globale permettant une analyse déterministe. Nous nous proposons ici de sélectionner les informations de type rationnel en faisant appel à des transducteurs finis — des mécanismes capables de distinguer des informations plus complexes, tels les transducteurs à pile, sont trop puissants et feraient sortir de la famille des algébriques.

Nous introduisons la notion de quasi déterminisme, en faisant précéder la reconnaissance par un automate à pile déterministe ordinaire, par une lecture de droite à gauche à l'aide d'un automate fini avec sortie. Cela revient à définir les langages quasi déterministes comme image inverse de langages déterministes par des fonctions rationnelles, au sens d'Eilenberg [4].

La famille des langages quasi déterministes, notée QDet, s'insère strictement entre Det et NA. Considérablement plus étendue que la famille Det et conservant les principales propriétés de celle-ci, elle présente donc l'intérêt d'élargir le champ des techniques développées pour l'étude des langages déterministes.

Un résultat de régularité, analogue au théorème d'Ogden [11] pour les langages déterministes, fournit une condition nécessaire pour qu'un langage soit quasi déterministe.

D'autre part la famille QDet est fermée par les opérations de morphisme inverse, intersection avec un langage rationnel, et passage au complémentaire.

Elle n'est pas fermée par image miroir, du fait de l'impuissance des transducteurs finis à détecter des informations de type algébrique qui permettraient une analyse déterministe : par exemple le langage :

$$\bar{L} = \{ a^n b^n c^p d^p \mid n, p \geqq 1 \} \cup \{ a^n b^{2n} c^q d^p \mid n, p, q \geqq 1 \text{ et } p \neq q \}$$

n'est pas quasi déterministe alors que son image miroir est un langage déterministe.

On peut cependant caractériser les langages qui sont quasi déterministes dans les deux sens de lecture, à partir des langages déterministes dans les deux sens de lecture.

Il faut noter d'autre part que les langages quasi déterministes ont trouvé une application dans un domaine voisin : ils sont supports de certaines séries algébriques et variables non commutatives [12].

En outre il existe des liens intéressants entre la famille QDet et d'autres surclasses de la famille des langages déterministes [9, 13].

I. PRÉLIMINAIRES

On supposera connues les propriétés classiques des langages rationnels et algébriques [2, 5, 14].

1. Étant donnés deux alphabets X et Y, une transduction rationnelle τ est par définition une partie rationnelle du monoïde $X^* \times Y^*$ [4]. L'étude et l'utilisation des transductions rationnelles sont facilitées par leur caractérisation en termes de bimorphismes alphabétiques — rappelons qu'un morphisme φ de X^* dans Y^* est dit alphabétique si $\varphi(X) \subset Y \cup \{\varepsilon\}$.

THÉORÈME DE NIVAT [10] : *La transduction $\tau \subset X^* \times Y^*$ est rationnelle si, et seulement si, il existe un alphabet Z, deux morphismes alphabétiques φ et ψ de Z^* dans X^* et Y^* respectivement, et un langage rationnel K sur Z, tels que :*

$$\tau = \{ \varphi h, \psi h); h \in K \}.$$

Étant donné un langage L sur X, l'image par τ de L s'écrit alors :

$$\tau L = \psi(\varphi^{-1} L \cap K)$$

On appelle fonction rationnelle, une transduction rationnelle $\tau : X^* \to Y^*$ qui est une fonction, i. e. pour tout mot f de X^*, $\operatorname{card}(\tau f) \leq 1$. Un résultat de Elgot et Mezei [4] permet de décomposer toute fonction rationnelle en produit de composition de 2 transductions rationnelles réalisables au moyen d'automates finis pourvus de fonction de sortie, appelés applications séquentielles.

Une application séquentielle droite se réalise au moyen d'un automate fini avec sortie qui lit et écrit de droite à gauche :

$$\delta = \langle Q, X, Y, \square, *, q_0 \rangle,$$

où :

– Q est un ensemble fini d'états parmi lesquels on distingue l'état initial q_0 ;

– X et Y sont les alphabets d'entrée et de sortie;

– $\square$ et $*$, fonctions de transition et de traduction, sont des applications partielles de $X \times Q$ dans, respectivement, Q et Y^*.

On étend $\square$ et $*$ en des applications de $X^* \times Q$ respectivement dans Q et Y^* par :

$$\varepsilon \square q = q, \qquad xf \square q = x \square (f \square q)$$

et :

$$\varepsilon * q = \varepsilon, \qquad xf * q = (x * (f \square q))\,(f * q).$$

L'application séquentielle définie est $\delta(f) = f * q_0$, pour tout mot f de X^*.

On définit de façon analogue une application séquentielle gauche à l'aide d'un automate fini avec sortie qui lit et écrit de gauche à droite. Notons que les « generalized sequential machines » – gsm –, utilisées en particulier dans [7] sont des applications séquentielles gauches partout définies.

Manifestement les applications séquentielles sont des fonctions rationnelles, mais en revanche il existe des fonctions rationnelles qui ne sont pas des applications séquentielles. Cependant :

THÉORÈME [4] : *Toute fonction rationnelle* $\tau : X^* \to Y^*$ *est produit de composition d'une application séquentielle gauche* (*resp. droite*) $\sigma_1 : X^* \to Z^*$, *et d'une application séquentielle droite* (*resp. gauche*) $\sigma_2 : Z^* \to Y^*$.

Notons de plus que la première application séquentielle σ_1, est par construction strictement alphabétique – i. e. $\sigma_1(x) \in Z$ pour tout $x \in X$.

2. Un *automate à pile déterministe* est défini par :

$$\mathscr{D} = \langle Q, X, \Gamma, \lambda, q_0, Z_0, F \rangle,$$

où :

Q est un ensemble fini d'états, parmi lesquels on distingue l'état initial q_0, F est un sous-ensemble de Q formé des états terminaux;

– X est l'alphabet d'entrée, Γ l'alphabet de pile et Z_0 le symbole initial de pile;

– la fonction de transition λ est une application partielle de $Q \times X \cup \{\varepsilon\} \times \Gamma$ dans $Q \times \Gamma^*$ telle que :

$\forall q \in Q$, $\forall Z \in \Gamma$, si $\lambda(q, \varepsilon, Z)$ est défini alors $\lambda(q, x, Z)$ est non défini $\forall x \in X$.

On appelle *langages déterministes* les langages reconnus par état terminal par un automate à pile déterministe.

Les langages reconnus par pile vide par un automate à pile déterministe sont exactement les langages déterministes préfixes [6, 8]. (Rappelons qu'un langage L de X^* est dit préfixe si pour tous mots u, v de X^*, si $u \in L$ et $uv \in L$ alors $v = \varepsilon$.)

La famille des langages déterministes est fermée par les opérations de morphisme inverse et d'intersection avec un langage rationnel – ce qui lui confère une structure de cylindre – et elle est aussi fermée par passage au complémentaire [6].

Avant d'énoncer le théorème d'Ogden, donnant une condition nécessaire pour qu'un langage soit déterministe, nous introduisons la notion de paire itérante [3], qui formalise un phénomène de régularité dans tous les langages algébriques.

Étant donné un langage L sur l'alphabet X, on appelle *paire itérante* du mot f dans L un quintuplet $\pi = (a, u, b, v, c)$ de mots de X^* tels que;

– $f = aubvc$;
– $au^n bv^n c \in L$, $\forall n \in \mathbb{N}$;
– $uv \neq 1$.

Si L est un langage algébrique, certaines paires itérantes traduisent des phénomènes d'enchâssement dans les grammaires produisant L : si L est engendré par la grammaire algébrique G d'axiome S (on utilise pour les grammaires algébriques les notations classiques – *voir* par exemple [7, 12]), on dira que la paire itérante π dans L est *grammaticale* relativement à G, d'axiome S, s'il existe un symbole non terminal T tel que $S \xrightarrow{*} aTc$, $T \xrightarrow{*} uTv$ et $T \xrightarrow{*} b$.

Les paires itérantes qui apparaissent dans les théorèmes d'itération (Bar Hillel Perlès Shamir, Ogden) sont grammaticales [3]. Pour les langages déterministes les paires itérantes du théorème d'Ogden sont aussi liées au fonctionnement des automates déterministes.

Une paire itérante $\pi=(a, u, b, v, c)$ est dite *automatique* [8, 13] relativement à l'automate déterministe $\mathcal{D}=\langle Q, X, \Gamma, \lambda, q_0, Z_0, F\rangle$ s'il existe $p, \bar{p}\in Q, y\in\Gamma$ et $\eta, \gamma\in\Gamma^*$ tels que :

$$(q_0, a, Z_0)\overset{*}{\vdash}(p, \varepsilon, \eta Y), \qquad (p, u, Y)\overset{*}{\vdash}(p, \varepsilon, \gamma Y),$$

$$(p, b, Y)\overset{*}{\vdash}(\bar{p}, \varepsilon, \varepsilon) \qquad \text{et} \qquad (\bar{p}, v, \gamma)\overset{*}{\vdash}(\bar{p}, \varepsilon, \varepsilon).$$

En d'autres termes, la lecture du facteur « v » ôte de la pile ce que la lecture du facteur « u » y a placé. Ainsi pour tout n, la lecture du mot au^nbv^n conduit l'automate dans la même configuration $(\bar{p}, \eta)\in Q\times\Gamma^*$ que la lecture du mot $aubv$.

Le résultat suivant [8, 13] établit un lien entre paires itérantes grammaticales et automatiques d'un langage déterministe :

Proposition : *Soit $\mathcal{D}$ un automate à pile déterministe et G la grammaire canoniquement associée à $\mathcal{D}$, les paires itérantes grammaticales relativement à G sont automatiques relativement à $\mathcal{D}$.*

Énonçons enfin le théorème d'Ogden pour les langages déterministes [11] :

Théorème : *Si L est un langage déterministe, il existe un entier N tel que si dans un mot f de L on distingue au moins N occurrences de lettres, f admette une factorisation $aubvc=f$ telle que :*

1° *(a, u, b, v, c) est une paire itérante dans L;*

2° *$(a, u$ et $b)$ ou $(b, v$ et $c)$ contiennent des occurrences distinguées, et ubv en contient au plus N;*

3° *si $c\neq\varepsilon$ alors :*

Pour tout mot c' de X^*, s'il existe p_0 tel que $au^{p_0}bv^{p_0}c'$ est un mot de L alors (a, u, b, v, c') est une paire itérante dans L.

Remarque [4] : Si L est un langage déterministe préfixe, alors la restriction $c\neq 1$ n'est plus nécessaire dans la condition 3 du théorème d'Ogden.

II. DÉFINITION ET STRUCTURE

Définition 1 : Un langage L sur l'alphabet X est dit quasi déterministe s'il existe un langage déterministe D sur l'alphabet Y, et une fonction rationnelle τ de X^* dans Y^*, tels que $L=\tau^{-1}D$.

Nous avons rappelé dans les préliminaires d'une part que toute fonction rationnelle τ est produit de composition d'une application séquentielle droite δ et d'une application séquentielle gauche γ, et d'autre part que la famille des langages déterministes est fermée par application séquentielle gauche inverse.

Donc si $L=\tau^{-1} D$ alors $L=\delta^{-1}\gamma^{-1} D=\delta^{-1} D'$ avec $D' \in \mathrm{Det}$. D'où la définition équivalente :

DÉFINITION 2 : Un langage L sur l'alphabet X est dit *quasi déterministe* si, et seulement si, il existe un langage déterministe D sur l'alphabet Y, et une application séquentielle droite δ de X^* dans Y^*, tels que $L=\delta^{-1} D$.

REMARQUE 1 : La lecture par l'application séquentielle commençant à droite, tout langage quasi déterministe peut provenir d'un langage déterministe préfixe.

En effet, soit $L=\delta^{-1} D$ avec $\delta=\langle Q, X, Y, \square, *, q_0\rangle$ et $D \in \mathrm{Det}$, soit $\#$ un symbole n'appartenant pas à Y, et soit $\delta'=\langle Q, X, Y \cup \{\#\}, \square', *', q_0\rangle$ avec :

$$\forall x \in X, \quad \forall q \in Q, \qquad x \square' q = x \square q$$

et :

$$x *' q = \begin{cases} (x * q)\# & \text{si} \quad q=q_0, \\ x * q & \text{si} \quad q \neq q_0. \end{cases}$$

Alors $L=\delta'^{-1}(D\#)$.

Nous présentons maintenant quelques exemples de langages quasi déterministes :

Exemple 1 : Tous les langages déterministes.

Exemple 2 : Tous les langages compilables [10], qui sont image inverse par une fonction rationnelle de langages déterministes particuliers : les langages de Dyck.

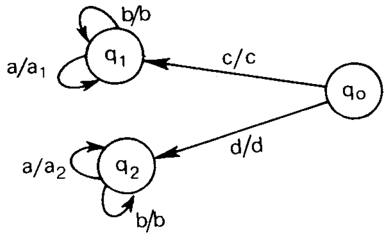
Exemple 3 : $L_1=\{a^n b^n c \mid n \geqq 1\} \cup \{a^n b^{2n} d \mid n \geqq 1\}$.

— Notons tout d'abord que le langage L_1 n'est pas déterministe. En effet son quotient à droite par le langage rationnel $R=\{c, d\}$ est le langage non déterministe $L=\{a^n b^n, n \geqq 1\} \cup \{a^n b^{2n}, n \geqq 1\}$ [6]. Or la famille des langages déterministes est fermée par quotient à droite par un langage rationnel [6].

— Par contre L_1 est un langage quasi déterministe : $L_1=\delta_1^{-1} D_1$, où δ_1 est le transducteur séquentiel qui donne a_1 pour images des « a » lus après c et a_2 pour images des « a » lus après d; et D_1 est un langage déterministe car union marquée de deux langages déterministes :

$$D_1=\{a_1^n b^n c ; n \geqq 1\} \cup \{a_2^n b^{2n} d ; n \geqq 1\}.$$

δ_1 :



Remarque 2 : Nous avons ainsi établi que la famille des langages déterministes est incluse strictement dans celle des langages quasi déterministes.

Le langage L_1 est quasi déterministe car les langages déterministes dont il est union peuvent être triés par le rationnel sous-jacent à une application séquentielle droite. Cet exemple illustre une propriété générale des langages quasi déterministes :

Définition 3 : Deux langages L_1 et L_2 sont dits *rationnellement séparables* s'il existe un langage rationnel R tel que $L_1 \subset R$ et $L_2 \cap R = \emptyset$.

Propriété 1 : La famille des langages quasi déterministes est fermée par union rationnellement séparable.

Preuve : Soient L_1 et L_2 des langages quasi déterministes :

$$L_1 = \delta_1^{-1} D_1 \quad \text{avec} \quad D_1 \in \text{Det} \quad \text{et :} \quad \delta_1 = \langle Q_1, X, Y, \square_1, *_1, q_{01} \rangle,$$

$$L_2 = \delta_2^{-1} D_2 \quad \text{avec} \quad D_2 \in \text{Det} \quad \text{et :} \quad \delta_2 = \langle Q_2, X, Z, \square_2, *_2, q_{02} \rangle.$$

Soit R un langage rationnel tel que $L_1 \subset R$ et $L_2 \cap R = \emptyset$; soit $\mathscr{A}$ un automate fini déterministe qui reconnaît le rationnel $\tilde{R}$ image miroir de R :

$$\mathscr{A} = \langle K, X, ., k_0, F \rangle.$$

Soient $\cent$ et $\$$ deux symboles n'appartenant ni à Y ni à Z; on construit l'application séquentielle droite

$$\delta = \langle Q_1 \times Q_2 \times K, X, Y \cup Z \cup \{\cent, \$\}, \square, *, [q_{01}, q_{02}, k] \rangle$$

avec :

$$\forall q_1 \in Q_1, \quad \forall q_2 \in Q_2, \quad \forall k \in K, \quad \forall x \in X :$$

$$x \,\square\, [q_1, q_2, k] = \begin{cases} \text{non défini si } x \,\square_1\, q_1 \text{ ou } x \,\square_2\, q_2 \text{ non défini,} \\ [x \,\square_1\, q_1, x \,\square_2\, q_2, k \,.\, x] \text{ sinon} \end{cases}$$

et :

$$x * [q_1, q_2, k] = \begin{cases} ¢\,(x *_1 q_1) & \text{si} \quad k \,.\, x \in F, \\ \$\,(x *_2 q_2) & \text{si} \quad k \,.\, x \notin F. \end{cases}$$

Alors :

$$L_1 \cup L_2 = \delta^{-1} ((\varphi_1^{-1} D_1 \cap ¢\,(Y \cup ¢)^*) \cup (\varphi_2^{-1} D_2 \cap \$\,(Z \cup \$)^*)$$

où :

$$\varphi_1 : Y \cup \{¢\} \to Y \quad \text{avec :} \quad \varphi_1(y) = y, \quad \forall y \in Y \quad \text{et} \quad \varphi_1(¢) = \varepsilon,$$
$$\varphi_2 : Z \cup \{\$\} \to Z \quad \text{avec :} \quad \varphi_2(z) = z, \quad \forall z \in Z \quad \text{et} \quad \varphi_2(\$) = \varepsilon.$$

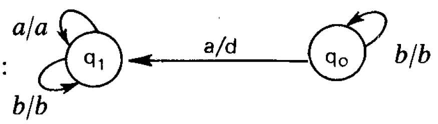
Ainsi $L_1 \cup L_2$, qui est l'image inverse par une application séquentielle droite de l'union marquée de 2 langages déterministes, est un langage quasi déterministe. ■

Notons cependant que la condition d'être rationnellement séparables n'est pas nécessaire pour que l'union de deux langages quasi déterministes soit quasi déterministe cela est une conséquence triviale de la fermeture, que nous démontrons plus loin, de QDet par passage au complémentaire.

Par ailleurs les langages quasi déterministes ne sont pas tous des unions finies de langages déterministes, comme le montre l'exemple suivant :

Exemple 4 : $L_2 = \{ w \in \{a, b\}^*;\ w = a^p bfab^p \text{ avec } f \in \{a, b\}^* \text{ et } p \in \mathbb{N} \}$:

— L_2 est quasi déterministe : $L_2 = \delta_2^{-1} D_2$, où δ_2 est l'application séquentielle droite qui traduit par « d » la dernière occurrence de « a » dans les mots de L_2, et D_2 est le langage déterministe $\{ a^p bfdb^p ;\ f \in \{a, b\}^* \text{ et } p \in \mathbb{N} \}$.

δ_2 :


— L_2 n'est pas union finie de langages déterministes : supposons en effet qu'il existe des langages déterministes $D_1, \ldots, D_l$, $l \in \mathbb{N}$. tels que $L_2 = \bigcup_{i=1}^{l} D_i$.

Soient $N_1, \ldots, N_l$ les entiers associés par le lemme d'Ogden à des grammaires algébriques engendrant respectivement $D_1, \ldots, D_l$; et soit $N = \sup\{N_i;\ i=1, \ldots, l\}$.

Considérons les mots :

$$\begin{aligned} w_1 &= a^N bab^N \\ &\vdots \\ w_{i+1} &= w_i ab^N \\ &\vdots \\ w_{l+1} &= w_l ab^N. \end{aligned}$$

Ces $l+1$ mots sont tous dans L_2. Étant donné que le nombre de langages D est l, il y a nécessairement deux mots w_{k_1} et w_{k_2}, avec $k_1 < k_2$, qui appartiennent au même langage D_i, pour quelque $i \in [1, l]$:

$$w_{k_1} = a^N b \underbrace{ab^N \ldots ab^N}_{k_1 \text{ fois}}$$

et :

$$w_{k_2} = a^N b \underbrace{ab^N \ldots ab^N ab^N \ldots ab^N}_{k_2 \text{ fois}}.$$

Si l'on marque dans w_{k_1} les N premières occurrences de « a », toute paire itérante $\pi = (\alpha, u, \beta, v, \gamma)$ dans L_2, devant être telle que α, u et β, ou β, v et γ contiennent des occurrences marquées, s'écrit :

$$\pi = (a^{p_1}, a^p, a^{p_2} \underbrace{ab^N \ldots ab^N}_{(k_1 - 1) \text{ fois}} ab^{q_1}, b^p, b^{q_2})$$

avec :

$$p_1 + p_2 = q_1 + q_2.$$

Or le mot w_{k_2}, qui appartient à D_i, est obtenu en remplaçant le dernier terme de π par $b^{q_2} \underbrace{ab^N \ldots ab^N}_{(k_2 - k_1) \text{ fois}}$.

Mais

$$(a^{p_1}, a^p, a^{p_2} \underbrace{ab^N \ldots ab^N}_{(k_1 - k_1) \text{ fois}} ab^{q_1}, b^p, b^{q_2} \underbrace{ab^N \ldots ab^N}_{(k_2 - k_1) \text{ fois}}$$

n'est pas une paire itérante dans D_i, aucune de ses itérées n'appartenant à L_2. D'après le lemme d'Ogden, ceci est en contradiction avec le fait que D_i est déterministe.

Donc L_2 n'est pas union finie de langages déterministes. ■

Pour terminer ce paragraphe, nous examinerons quelques propriétés de clôture de la famille des langages quasi déterministes.

PROPRIÉTÉ 2 : QDet est fermée par passage au complémentaire.

Preuve . Soit $L=\delta^{-1}D$ où $D\in$ Det et δ est une application séquentielle droite de X^* dans Y^* alors

$$\complement L=\{\omega\in X^*;\ \delta\omega\notin D\}=\{\omega\in X^*;\ \delta\omega\in\complement D\}\cup\{\omega\in X^*;\ \delta(\omega)\text{ pas défini}\}$$

ainsi $\complement L=\delta^{-1}(\complement D)\cup\complement\,\delta^{-1}(Y^*)$; $\complement L$ est l'union du langage quasi déterministe $\delta^{-1}\complement D$ et du langage rationnel $\complement\,\delta^{-1}(Y^*)$. C'est donc un langage quasi déterministe; en effet la fermeture de QDet par union rationnellement séparable entraîne sa fermeture par union rationnelle : $L\cup R=(L-R)\cup R$. ■

PROPRIÉTÉ 3 : QDet est fermée par intersection avec un langage rationnel.

Preuve : Soit $L=\delta^{-1}D$ où $D\in$ Det et $\delta=\langle Q, X, Y, \square, *, q_0\rangle$; Soit R un langage rationnel et $\mathscr{A}=\langle K, X, ., k_0, F\rangle$ un automate fini déterministe qui reconnaît le langage rationnel $\tilde{R}$, image miroir de R.

Soit $\#$ un symbole n'appartenant pas à l'alphabet Y. On construit une application séquentielle droite :

$$\delta'=\langle Q\times K, X, Y\cup\{\#\}, \dot{\square}', \dot{*}', [q_0, k_0]\rangle$$

avec :

$$\forall q\in Q,\quad \forall x\in X,\quad \forall k\in K,$$

$$x\,\square\,[q,k]=\begin{cases}[x\,\square\,q,\ k\,.\,x] & \text{si}\quad x\,\square\,q\text{ est défini},\\ \text{non défini sinon,}\end{cases}$$

$$x*[q,k]=\begin{cases}\#\,(x*q) & \text{si}\quad k\,.\,x\in F,\\ x*\ q & \text{si}\quad kx\notin F.\end{cases}$$

Alors $L\cap R=\delta^{-1}(h^{-1}D\cap\#(Y\cup\#)^*)$ où $h: Y\cup\{\#\}\to Y: h(y)=y$, $\forall y\in Y$ et $h(\#)=\varepsilon$. Det étant fermée par morphisme inverse et intersection avec un rationnel, le langage $L\cap R$ est donc quasi déterministe. ■

La composition de 2 fonctions rationnelles étant une fonction rationnelle, la famille QDet est fermée par fonction rationnelle inverse. En particulier QDet est donc fermée par morphisme inverse. Cette propriété, jointe à la propriété 3 signifie :

PROPOSITION : *La famille* QDet *possède une structure de cylindre.*

Nous avons par ailleurs montré [15] que QDet est un cyclindre non principal; i. e. on ne peut pas engendrer toute la famille à partir d'un seul langage quasi déterministe, à l'aide des opérations de morphisme inverse et intersection avec un langage rationnel.

III. THÉORÈME D'ITÉRATION

Un théorème d'itération, analogue au théorème d'Ogden [11] pour les langages déterministes, donne une condition nécessaire pour qu'un langage soit quasi déterministe. En application de ce théorème, nous montrons que la famille QDet n'est pas fermée par image miroir.

THÉORÈME D'ITÉRATION : *Soit L un langage quasi déterministe sur l'alphabet X. Il existe deux entiers N et P tels que si dans un mot f de L, on distingue au moins N occurrences de lettres, f admette une factorisation $f=aubvc$ telle que :*

1° *$\pi=(a, u, b, v, c)$ est une paire itérante dans L;*

2° *soit a et u et b, soit b et v et c contiennent chacun des occurrences distinguées, et ubv en contient au plus N;*

3° *s'il existe des entiers p et q tels que $au^pbv^{p+q}c \in L$, alors $(a, u, b, v, v^q c)$ est une paire itérante dans L;*

4° *si $|c| \geqq P$, il existe une factorisation $c=c_1c_2c_3$ telle que s'il existe des entiers s et p tels que $au^pbv^pc_1c_2^sc_3 \in L$, alors $(a, u, b, v, c_1c_2^sc_3)$ est une paire itérante dans L.*

Preuve : Soient D un langage déterministe préfixe sur Y, et δ une application séquentielle droite de X^* dans Y^*, tels que $L=\delta^{-1} D$:

$$\delta=\langle X, Y, Q, \square, *, q_0 \rangle \qquad \text{avec} \quad P=\operatorname{card}(Q).$$

D'après le théorème de Nivat, il existe un langage rationnel $K \subset Z^*$ et 2 morphismes alphabétiques $\varphi: Z^* \to X^*$ et $\psi: Z^* \to Y^*$ tels que l'image de D dans la transduction rationnelle δ^{-1} est :

$$L=\delta^{-1} D=\psi(\varphi^{-1} D \cap K).$$

Soit $\mathscr{D}$ un automate à pile déterministe reconnaissant D; et soit G_0 la grammaire canoniquement associée à $\mathscr{D}$.

A partir de G_0, on construit de manière classique (*voir* [5]) la grammaire $G=\psi(\varphi^{-1} G_0 \cap K)$ qui engendre L.

Soit N l'entier associée à cette grammaire algébrique G par le théorème d'Ogden. Si f est un mot de L dans lequel on distingue au moins N occurrences de lettres, le théorème d'Ogden pour les langages algébriques assure l'existence d'une paire itérante $\pi=(a, u, b, v, c)$ de f dans L, grammaticale par rapport à G, telle que a et u et b, ou b et v et c contiennent des occurrences distinguées, et le sous-mot ubv n'en contient pas plus de N.

De cette paire π dans L, on peut déduire, d'après la construction de G à partir de G_0, une paire itérante $\hat{\pi}=(\hat{a}, \hat{u}, \hat{b}, \hat{v}, \hat{c})$ dans D, grammaticale relativement à G_0 (*voir* [3]).

La paire $\hat{\pi}$ est donc, comme nous l'avons rappelé dans le paragraphe de préliminaires, automatique pour $\mathscr{D}$; elle vérifie la condition de changement de finale du théorème d'Ogden pour les langages déterministes préfixes : $\forall g \in X^*$, si $\exists p_0 \in N$ teq. $\hat{a}\hat{u}^{p_0} \hat{b}\hat{v}^{p_0} g \in D$, alors $(\hat{a}, \hat{u}, \hat{b}, \hat{v}, g)$ est une paire itérante dans D.

Supposons alors qu'il existe deux entiers p et t tels que $au^p bv^p v^t c \in L$; alors $\hat{a}\hat{u}^p \hat{b}\hat{v}^p \hat{v}^t \hat{c} \in D$; donc d'après le théorème d'Ogden, $(\hat{a}, \hat{u}, \hat{b}, \hat{v}, \hat{v}^t \hat{c})$ est une paire itérante dans D. Et puisque $L = \delta^{-1} D$ cela implique que $(a, u, b, v, v^t c)$ est une paire itérante dans L.

D'autre part, si $|c| > P$, où P est le nombre d'états de l'application séquentielle δ, c se factorise en $c_1 c_2 c_3$ avec $\delta(c_1 c_2 c_3) = \hat{c}_1 \hat{c}_2 \hat{c}_3$, et $\delta(c_1 c_2^n c_3) = \hat{c}_1 \hat{c}_2^n \hat{c}_3$ pour tout n.

Donc s'il existe deux entiers p et s tels que $au^p bv^p c_1 c_2^s c_3 \in L$, on aura $\hat{a}\hat{u}^p \hat{b}\hat{v}^p \hat{c}_1 \hat{c}_2^s \hat{c}_3 \in D$. D'où, d'après le théorème d'Ogden, $(\hat{a}, \hat{u}, \hat{b}, \hat{v}, \hat{c}_1 \hat{c}_2^s \hat{c}_3)$ est une paire itérante dans D; et donc $(a, u, b, v, c_1 c_2^s c_3)$ est une paire itérante dans L. ■

Ce théorème d'itération permet de montrer que certains langages non ambigus classiques qui ne sont pas déterministes, ne sont pas non plus quasi déterministes.

Exemple 1 : $L_1 = \{ a^n b^n ; n \geqq 1 \} \cup \{ a^n b^{2n} ; n \geqq 1 \}$ n'est pas quasi déterministe.

En effet, soit $w = a^{N!} b^{N!}$, où N est l'entier d'Ogden associé à une grammaire algébrique engendrant L_1. Toute paire itérante $(\alpha, u, \beta, v, \gamma)$ de w dans L_1 telle que $|u \beta v| < N$ s'écrit $\pi = (a^{n_1}, a^q, a^{n_2} b^{m_1}, b^q, b^{m_2})$ avec $n_1 + n_2 = m_1 + m_2$, et $q < N$; q divise donc $N!$: il existe un entier k_0 tel que $k_0 q = N!$.

Soient r_0 et s_o deux entiers tels que $r_0 - s_0 + 1 = k_0$; alors $r_0 q = s_0 q + m_1 + m_2$ puisque $N! = q + m_1 + m_2$.

Donc le mot $w' = a^{n_1} a^{s_0 q} a^{n_2} b^{m_1} b^{s_0 q} b^{r_0 q + m_2}$ appartient à L_1.

Si L_1 était quasi déterministe, on aurait donc, d'après le lemme d'itération, $a^{n_1} a^{sq} a^{n_2} b^{m_1} b^{sq} b^{r_0 q + m_2}$ dans L_1 pour tout entier s; or ceci est impossible car les mots de L_1 sont seulement de 2 formes : autant, ou alors deux fois plus, d'occurrences de « b » que d'occurrences de « a ». ■

Exemple 2 : $L_2 = \{ f\tilde{f}; f \in \{ a, b \}^* \}$ n'est pas quasi déterministe.

En effet, soit $w = (ba^N b)^{2(P+2)}$, où N est l'entier d'Ogden associé à une grammaire algébrique engendrant L_2, et P le nombre d'états d'une application séquentielle droite δ permettant de former L_2 à partir d'un langage déterministe. Si l'on distingue les N occurrences de « a » du $(P+2)\,i$-ième paquet « $ba^N b$ », toutes les paires itérantes de w dans L_2, respectant les conditions du théorème sur les occurrences distinguées, sont de la forme :

$$\pi = ((ba^N b)^{P+1} ba^{n_1}, a^q, a^{n_2} bba^{m_1}, a^q, a^{m_2} b (ba^N b)^{P+1})$$

avec $n_1 + n_2 = m_1 + m_2$.

D'autre part, le nombre de paquets $ba^N b$ dans le dernier terme de la paire étant strictement supérieur à P, il existe nécessairement une factorisation de ce terme en $c = c_1 c_2 c_3$ avec $c_1 = a^{m_2} b (ba^N b)^{P_1}$, $c_2 = (ba^N b)^{P_2}$ et $c_3 = (ba^N b)^{P_3}$, l'application séquentielle droite δ se trouvant dans le même état en début et en fin de lecture de c_2.

Soit alors $s = 2$ ou 3, selon que P_2 est pair ou impair; le mot $\mathrm{w}' = \mathrm{aubvc_1 c_2^s c_3}$ est dans L_1 car il est de la forme $(ba^N b)^{2r}$, où $r = P + 2 + ((P_2 (s-1)/2)$ (on a déplacé le milieu du mot!); mais pour tout $n \neq 1$, $au^n bv^n c_1 c_2^s c_3$ n'est pas dans L_1. Donc L_1 n'est pas quasi déterministe. ■

PROPOSITION : *La famille* QDet *est incluse strictement dans la famille* NA.

Preuve : La famille NA contient la famille QDet, car elle contient les langages déterministes et est fermée par fonction rationnelle inverse. Les deux exemples précédents de langages non ambigus prouvent que l'inclusion est stricte. ■

L'exemple suivant illustre le caractère dissymétrique de la famille QDet, qui résulte de la différence de nature entre les lectures par un automate fini et par un automate à pile.

Exemple 3 :

$$L_3 = \{ a^n b^n c^p d^q ;\ n, p, q \geqq 1 \text{ et } p \neq q \} \cup \{ a^{2n} b^n c^p d^p ;\ n, p \geqq 1 \}.$$

L_3 n'est pas quasi déterministe : soit $w = a^{2N!} b^{N!} c^{P+1} d^{P+1}$, où N est l'entier d'Ogden associé à une grammaire algébrique engendrant L_3, et P le nombre d'états d'une application séquentielle droite δ permettant de former L_3 à partir d'un langage déterministe.

Si l'on distingue dans w les $2N!$ occurrences de « a », toute paire itérante dans L_3 doit avoir ses trois premiers termes contenant des occurrences distinguées, et s'écrit donc $\pi = (a^{n_1}, a^{2q}, a^{n_2} b^{m_1}, b^q, b^{m_2} c^{P+1} d^{P+1})$ avec $n_1 + n_2 = 2(m_1 + m_2)$. Or $q < N$, donc il existe un entier k_0 tel que $k_0 q = N!$. Soient alors r_0 et s_0 deux entiers tels que $r_0 - s_0 + 1 = k_0$; on a $r_0 q = s_0 q + m_1 + m_2$ puisque $N! = q + m_1 + m_2$. De plus, le nombre d'occurrences de « d » étant strictement supérieur à P, il existe une factorisation de d^{P+1} en $d_1 = d^{r_1}$, $d_2 = d^{r_2}$ et $d_3 = d^{r_3}$, l'application séquentielle droite δ se trouvant dans le même état en début et en fin de lecture de d_2.

Le mot $w'=a^{n_1}(a^{2q})^{s_0}a^{n_2}b^{m_1}(b^q)^{s_0}b^{r_0q+m_2}c^{P+1}d^{P+1+r_2}$ est donc dans L_3 Aussi si L_3 était quasi déterministe

$$\sigma=(a^{n_1}, a^{2q}, a^{n_2}b^{m_1}, b^q, b^{r_0q+m_2}c^{P+1}d^{P+1+r_2})$$

serait une paire itérante dans L_3.

Or puisque $q\neq 1$, il est clair que la deuxième itérée de la paire σ n'est pas dans L_3. Donc L_3 n'est pas quasi déterministe. ■

Par contre le langage $\tilde{L}_3$ est déterministe : il est reconnu par un automate à pile déterministe qui en fonction du nombre de « c » par rapport au nombre de « d », se place soit dans un état à partir duquel il reconnaît le langage $\{b^na^{2n};\, n\geqq 1\}$, soit dans un état à partir duquel il reconnait le langage $\{b^na^n;\, n\geqq 1\}$. ■

On a donc montré la proposition suivante :

PROPOSITION : *La famille* QDet *n'est pas fermée par image miroir.*

Dans le paragraphe suivant nous montrons que c'est en fait le défaut de symétrie de la famille Det qui se transporte dans QDet.

IV. LANGAGES BIQUASI DÉTERMINISTES

L'étude de la sous-classe des langages quasi déterministes dont l'image miroir est aussi quasi déterministe montre qu'elle est exactement composée des langages que l'on peut former, par fonction rationnelle inverse, à partir de langages déterministes dans les deux sens de lecture.

DÉFINITION : Un langage B est *bidéterministe* si B et $\tilde{B}$ sont des langages déterministes.

Un langage L est *biquasi déterministe* si L et $\tilde{L}$ sont des langages quasi déterministes.

Un langage $L\subset X^*$ est *quasi bidéterministe* s'il existe une fonction rationnelle $\tau: X^*\to Y^*$ et un langage bidéterministe $B\subset Y^*$ tels que $L=\tau^{-1}B$.

PROPOSITION 1 : *Tout langage quasi bidéterministe est biquasi déterministe.*

Preuve : il est clair que si $L=\tau^{-1}B$ où B et $\tilde{B}$ sont déterministes, et $\tau=\gamma\circ\delta$ alors L est biquasi déterministe car $L=\delta^{-1}(\gamma^{-1}B)=\delta^{-1}B'$ où $B'=\gamma^{-1}B$ est déterministe et $\tilde{L}=\tau^{-1}\tilde{B}=\delta^{-1}(\gamma^{-1}\tilde{B})=\delta^{-1}B''$ où $B''=\gamma^{-1}\tilde{B}$ est déterministe. ■

Nous abordons maintenant la réciproque de cette proposition.

LEMME 1 : *Si L est biquasi déterministe alors il existe des langages déterministes D et D' et deux applications séquentielles alphabétiques l'une droite δ et l'autre gauche γ tels que $L=\delta^{-1}D=\gamma^{-1}\tilde{D}'$.*

Preuve : L étant biquasi déterministe, d'une part $L=\delta^{-1} D$ et d'autre part $\tilde{L}=\delta'^{-1} D'$, où δ' est une application séquentielle droite alphabétique et D' un langage déterministe.

Soit γ l'application séquentielle gauche formée à partir de δ' en gardant les mêmes fonctions de transition et de traduction, mais en inversant les sens de lecture et d'écriture; on a clairement $L=\gamma^{-1} \tilde{D}'$. ■

Ensuite, étant donné un langage biquasi déterministe $L=\delta'^{-1} D=\gamma^{-1} \tilde{D}'$, nous construisons un langage quasi déterministe C codant L, qui simule en même temps les langages D et $\tilde{D}'$.

Soit $d: X \rightarrow (Q \times Y \times Q)^*$ l'application qui décrit le comportement du transducteur $\delta=\langle Q, X, Y, q_1, \square, * \rangle$ (avec $Q=\{q_1, \ldots, q_n\}$) sur les lettres de X :

$$d(x)=[q_{l_r}, y_{j_r}, q_{i_r}] \ldots [q_{l_2}, y_{j_2}, q_{i_2}] [q_{l_1}, y_{j_1}, q_{i_1}]$$

avec $1=i_1<i_2 \ldots <i_r \leqq n$, et $\forall t \in [1, r]$, $x \square q_{i_t}=q_{l_t}$ et $x * q_{i_t}=y_{j_t}$.

De même, soit $g: X \rightarrow (K \times Z \times K)^*$ l'application qui décrit le comportement du transducteur $\gamma=\langle K, X, Z, k_1, \dot{\square}, \dot{*} \rangle$ (avec $K=\{k_1, \ldots, k_p\}$) sur les lettres de X :

$$g(x)=[k_{i_1}, z_{j_1}, k_{l_1}] [k_{i_2}, z_{j_2}, k_{l_2}] \ldots [k_{i_r}, z_{j_r}, k_{l_r}]$$

avec :

$$1=i_1<i_2<\ldots<i_r \leqq p$$

et :

$$\forall t \in [1, r], \quad k_{i_t} \overset{\circ}{\square} x=k_{l_t} \quad \text{et} \quad k_{i_t} \overset{\circ}{*} x=y_{j_t}.$$

On définit le langage C sur l'alphabet :

$$X \cup Y \cup Z \cup K \cup Q \cup \{\#\}=A \cup \{\#\}$$

par :

$$C=\{x_1 g(x_1) d(x_1) x_1 \# \ldots \# x_m g(x_m) d(x_m) x_m \# \mid x_1 \ldots x_m \in L\}$$

Si f est le morphisme de $X^* \rightarrow (A \cup \{\#\})^*$ tel que $f(x)=x g(x) d(x) x \#$ on a clairement :

LEMME 2 : $L=f^{-1} C$.

Il reste à montrer que C est un langage quasi bidéterministe.

LEMME 3 : *Le langage C est image inverse par une fonction rationnelle d'un langage bidéterministe B.*

Preuve : Soit $\overline{Q}$ (resp. $\overline{K}$) une copie de Q (resp. K) sur un alphabet disjoint.

On définit le morphisme $h: A \cup \bar{Q} \cup \bar{K} \to A$ tel que :

$$\begin{cases} h(l)=l & \text{pour tout } l \text{ dans } X \cup Y \cup Z, \\ h(q)=h(\bar{q})=q & \text{pour tout } q \text{ dans } Q, \\ h(k)=h(\bar{k})=k & \text{pour tout } k \text{ dans } K. \end{cases}$$

Pour tout x de X, posons :

$$d'(x)=h^{-1}(d(x) \cap (Q \times Y \times Q)^* (\bar{Q} \times Y \times Q)(Q \times Y \times Q)^*)$$

et :

$$g'(x)=h^{-1}(g(x) \cap (K \times Z \times K)^* (K \times Z \times \bar{K})(K \times Z \times K)^*),$$

et notons v_x (resp. u_x) la lettre barrée dans $d'(x)$ [resp. $g'(x)$].

Soit :

$$B = \left\{ \begin{array}{c} x_1 g'(x_1) d'(x_1) x_1 \# \ldots \# x_m g'(x_m) d'(x_m) x_m \# \\ \text{tels que } x_1 \ldots x_m \in L \\ v_{x_m} = x_m \underset{\circ}{\square} q_1, \text{ et } v_{x_{t-1}} = x_{t-1} \underset{\circ}{\square} v_{x_t} \text{ pour tout } t=2, \ldots, m \\ \text{et } u_{x_1} = k_1 \square x_1, \text{ et } u_{x_{t+1}} = u_{x_t} \square x_{t+1} \text{ pour tout } t=1, \ldots, m-1. \end{array} \right\}$$

Ainsi dans les mots de B, les lettres barrées $v_{x_m}, \ldots, v_{x_1}$ (*resp.* $u_{x_1}, \ldots, u_{x_m}$) représentent la suite des états de l'application séquentielle droite δ (resp. gauche γ) lors de la lecture du mot $x_1 \ldots x_m$.

(*a*) *Le langage B est bidéterministe*

On peut construire un automate déterministe reconnaissant B de la manière suivante : pour les lettres de Y qui suivent immédiatement une lettre de $\bar{Q}$, on reproduit le mouvement de l'automate déterministe qui reconnait le langage D, et l'on ajoute un contrôle rationnel pour vérifier que les sous-mots restants [$g(x)$, début et fin de $d(x)$] sont correctement formés. La construction effective est laissée au lecteur.

De manière analogue, la preuve que $\tilde{B}$ est déterministe consiste en la donnée explicite d'un automate déterministe qui comporte une reproduction des mouvements d'un automate déterministe reconnaissant D' pour les letttres de Z précédant immédiatement une lettre de $\bar{K}$, un contrôle rationnel de bonne formation de sous mots finis.

(*b*) *Il existe une fonction rationnelle* τ *telle que* $C=\tau^{-1}B$

Soit τ :

$$\begin{tikzcd}
 & F \arrow[dl, "\varphi"'] \arrow[dr, "\psi"] & \\
C\subset A\cup\{\#\} & & B\subset A\cup\overline{Q}\cup\overline{K}\cup\{\#\}
\end{tikzcd}$$

où F est l'alphabet :

$$X\cup\{\#\}\cup(K\times Z\times(K\cup\overline{K}))\cup((Q\cup\overline{Q})\times Y\times Q),$$

où φ est le morphisme :

$$F\to A\cup\{\#\};\qquad \varphi(x)=x,\qquad \forall x\in X\cup\{\#\}$$

et :

$$\varphi(\overline{q},y,q')=\varphi(q,y,q')=(q,y,q')$$

et :

$$\varphi(k,z,\overline{k}')=\varphi(k,z,k')=(k,z,k'),$$

et ψ le morphisme identité sur F.

Soit R_1 le langage rationnel $U_1^*T_1\setminus U_1^*V_1U_1^*$ avec :

$$U_1=[K\times Z\times(K\cup\overline{K})]^*.\ (Q\times Y\times Q)^*.(\overline{Q}\times Y\times Q).(Q\times Y\times Q)^*X\#$$

$$T_1=X.[K\times Z\times(K\cup\overline{K})]^*.(Q\times Y\times Q)^*.(\overline{Q}\times Y\times\{q_1\})X\#.$$

$$V_1=\left\{\begin{array}{c}X.[K\times Z\times(K\cup\overline{K})]^*.(Q\times Y\times Q)^*.\ (\overline{q_j'},Y,q_j).(Q\times Y\times Q)^*X\#\\ X.[K\times Z\times(K\cup\overline{K})]^*.(Q\times Y\times Q)^*.(\overline{q_i'},Y,q_i).(Q\times Y\times Q)^*X\\ \text{avec } q_i,q_i',q_j,q_j'\in Q \text{ et } q_i'\neq q_j\}.\end{array}\right\}$$

De façon analogue, soit R_2 le langage rationnel $T_2U_2^*\setminus U_2^*V_2U_2^*$ avec :

$$U_2=X(K\times Z\times K)^*.(K\times Z\times\overline{K}).(K\times Z\times K)^*.((Q\cup\overline{Q})\times Y\times Q)^*.X\#$$

$$T_2=X(\{k_1\}\times Z\times\overline{K}).(K\times Z\times K)^*.((Q\cup\overline{Q})\times Y\times Q)^*.X\#$$

$$V_2=\left\{\begin{array}{c}X.(K\times Z\times K)^*.(k_i,Y,\overline{k_i'}).\ (K\times Z\times K)^*.[(Q\cup\overline{Q})\times Y\times Q]^*.X\#\\ X(K\times Z\times K)^*.(k_j,Y,\overline{k_j'}).\ (K\times Z\times K)^*.[(Q\cup\overline{Q})\times Y\times Q]^*X\#\\ \text{avec } k_i,k_i',k_j,k_j'\in Q \text{ et } k_i'\neq k_j\}.\end{array}\right\}$$

Définissons $R=R_1 \cap R_2$, alors la transduction τ est univoque, car le morphisme φ est injectif sur R, et il apparait clairement que $C=\tau^{-1} B$. ■

Nous avons ainsi démontré la réciproque de la proposition 1 et par conséquent le théorème suivant :

THÉORÈME : *Un langage L est biquasi déterministe si, et seulement si, il est quasi bidéterministe.*

BIBLIOGRAPHIE

1. J. M. AUTEBERT, G. COUSINEAU et M. NIVAT, *Théorie des automates et des langages formels*, Poly Institut de Programmation, Paris, 1976.
2. J. BERSTEL, Transduction and Context Free Languages, Teubner Verlag, 1978.
3. L. BOASSON, *Langages algébriques, Paires itérantes, et Transductions rationnelles*, Theorical Computer Science, vol. 2, 1976, p. 209-223.
4. S. EILENBERG, 1976, *Automata, Languages and Machines*, Vol. B, Academic Press, New York, London.
5. S. GINSBURG, *The Mathematical Theory of Context Free Languages*, Mac Graw Hill, 1966.
6. S. GINSBURG et S. GREIBACH, *Deterministic Context Free Languages*, Information and Control, vol. 9, 1966, p. 620-648.
7. S. GREIBACH, *Jump PDA's and Hierarchies of Deterministic CF Languages*, S.I.A.M. Journal of Computing, vol. 3, 1974, p. 111-127.
8. M. HARRISON et I. HAVEL, *Strict Deterministic Grammars*, J.C.S.S., vol. 7, 1973, p. 237-277.
9. T. HIBBARD, *A Generalisation of Context Free Determinism*, Information and Control, vol. 11, 1967, p. 196-238.
10. M. NIVAT, *Transduction des langages de Chomsky*, Annales de l'Institut Fourrier, 1968.
11. W. OGDEN, *Intercalation Theorems for Pusdown Store and Stack Languages*, Ph. D. Thesis, Standford, 1968.
12. C. REUTENAUER, *Propriétés arithmétiques et topologiques de séries rationnelles en variables non commutatives*. Thèse 3e cycle, Université Paris-VI, 1977.
13. J. SAKAROVITCH, *Syntaxe des langages de Chomsky. Essai sur le déterminisme*, Thèse d'État, Université Paris-VII, Rapport LITP n° 80-10, 1980.
14. A. SALOMAA, *Formal Languages*, Academic Press, New York, 1973.
15. M. SORIA, *Langages quasi déterministes*, Thèse 3e cycle, Université Paris-VII, 1978.
16. D. WOTSCHKE, *States Can Sometimes Do More Than Stack Symbols in PDA*. Technical Report, Penn State University, 1977.