

NOËLLE BLEUZEN-GUERNALEC

GEORGES BLANC

**Production en temps réel et complexité de
structure de suites infinies**

RAIRO. Informatique théorique et applications, tome 23, n° 2 (1989),
p. 195-216

http://www.numdam.org/item?id=ITA_1989__23_2_195_0

© AFCET, 1989, tous droits réservés.

L'accès aux archives de la revue « RAIRO. Informatique théorique et applications » 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/>

PRODUCTION EN TEMPS RÉEL ET COMPLEXITÉ DE STRUCTURE DE SUITES INFINIES (*)

par Noëlle BLEUZEN-GUERNALEC ⁽¹⁾ et Georges BLANC ⁽¹⁾

Communiqué par J. E. PIN

Abstract. – *We study the complexity structure of real time product sequences.*

We use finite states automata which reread themselves. These give a class of sequences large enough but whose control of structure is possible by the iterative feature of the process. We get results about factors, which make it possible to compare this process of production to others: Tag Sequences, Counter Machines, Turing Machines.

In the growing case and the uniform case, we get an exact bound for the complexity of factors.

Résumé. – *Nous étudions la complexité de structure de suites infinies produites en temps réel.*

Les automates finis se relisant que nous utilisons fournissent une classe de suites suffisamment large mais dont le contrôle de structure reste possible par le caractère itératif du processus. Nous obtenons des résultats concernant les facteurs, qui permettent de comparer ce processus de production à d'autres: Tag Sequences, Machines à compteurs, Machines de Turing.

Dans le cas croissant et le cas uniforme, nous obtenons une borne exacte de la complexité de facteurs.

1. INTRODUCTION

L'étude des suites infinies présente un intérêt évident non seulement pour la théorie des nombres, mais aussi pour l'informatique théorique, et les questions que celles-ci posent sont à l'origine de nombreux travaux. C'est par exemple l'article de Hartmanis et Stearns [HAR; 10] sur le temps de construction des suites infinies — que l'on considère souvent à l'origine de l'étude de la complexité de calcul [HOP; 11] — rapidement suivi d'autres aussi importants sur le temps limité de production des suites [ROS; 16],

(*) Reçu en octobre 1986, révisé en mars 1987.

⁽¹⁾ Département de Mathématique-Informatique, Faculté des Sciences de Luminy, 70, route Léon-Lachamp, 13288 Marseille Cedex 9.

[FIS; 8], [FIS; 9]. Indépendamment du temps de construction, les suites infinies sont étudiées par leur complexité structurelle: le fameux article de Cobham [COB; 4] sur les Uniform Tag Sequences suscite nombre de travaux sur la structure des facteurs — facteurs inévitables, répétitions, rigidité... [BER; 1], [PAN; 13] — ou sur les supports de la suite (ensembles des indices des termes ayant une valeur donnée) [CHR; 3].

La production et la connaissance des suites Constructibles en Temps Réel (C.T.R. : un index des sigles se trouve à la fin), suites de la classe \mathcal{S} [HAR; 10] jouent un rôle important pour l'étude de la complexité de temps nécessaire à des calculs plus généraux, ne serait-ce que parce que toute suite constructible est image homomorphe d'une suite C.T.R.

La définition par restriction du temps de production par Machine de Turing [HAR; 10] ne permettant guère de maîtriser la complexité structurelle des suites C.T.R., il est important de proposer des moyens de production assurant un meilleur contrôle. C'est le cas par exemple des Machines à Compteurs de Fischer, Meyer et Rosenberg [FIS; 8].

La notion, optimale pour la faiblesse des moyens de production relativement à la richesse des propriétés exploitables, est sans doute celle d'Uniform Tag Sequence (U.T.S.) de Cobham. Le rapprochement de l'idée de production itérative de Cobham, contrôlant au mieux la complexité intrinsèque, et du fait que toute suite constructible est image homomorphe d'une suite C.T.R., conduit à l'étude des Tag Machines Généralisées (T.M.G.) [COB; 4], [MIN; 12] produisant les suites dites Tag Sequences Généralisées (T.S.G.).

Les T.M.G. peuvent être conçues comme :

- 1, version des Machines à Compteurs [FIS; 8] où toute la mémoire active des compteurs est remplacée par l'usage auxiliaire de l'unique bande, lecture-écriture-production;
2. extension des Tag Machines de Cobham-Post à une mémoire de contrôle d'états finis.

La suite produite, ou extraite de la production par effacement des lettres de l'alphabet auxiliaire de travail, reflète alors toute la complexité de temps, d'espace, de structure.

En effet :

- la fonction $r^{\Sigma}(n)$, indice de n -ième occurrence d'un symbole de l'alphabet utile Σ , correspond au temps de construction [HAR; 10] de la suite Σ -suite extraite;
- la fonction $\pi^T(n)$, nombre de symboles de l'alphabet de travail T dans le préfixe de longueur $n + 1$, section réciproque de $r^T(n)$, fournit une mesure

raisonnable de l'espace nécessaire à la construction de la Σ -suite;

– le cardinal fini de l'alphabet de travail T correspond au nombre de mémoires auxiliaires nécessaires à la construction;

– enfin, le caractère itératif de la production (prop. 1) reflète la complexité structurelle de la Σ -suite.

On essaie, dans le travail qui suit, de dégager quelques propriétés de structure des suites Constructibles en Temps Réel obtenues par Tag Machines Généralisées, comparativement à d'autres processus de production. La section 2 présente les notions de T.M.G., Tag production, point fixe d'itération, et les premières observations associées aux suites infinies ainsi produites. La section 3 s'intéresse aux T.S.G. relativement à la classe des suites Constructibles en Temps réel. Elles en constituent une sous-classe, stricte: la suite de Liouville, par exemple, n'est pas constructible par T.M.G. (corollaire 8). Néanmoins, comme les suites C.T.R., les T.S.G. gardent la propriété de produire par image homomorphe toutes les suites constructibles (prop. 4).

Quel est le comportement des T.S.G. par rapport au codage? La production par morphisme itéré étant extrêmement rigide, c'est le comportement par codage [COB; 4], [PAN; 14] qui, préservant suffisamment la structure dans les Tag Sequences, apporte des propriétés nouvelles de fermeture. La génération par T.M.G. est évidemment plus souple par nature, et le prolongement par codage semble moins important. En effet, se donner la possibilité d'appliquer des morphismes quelconques permet d'obtenir beaucoup trop: toutes les suites constructibles. Le prolongement par morphisme h « peu effaçant » (densité non nulle du support de h) fournit encore des suites C.T.R. (prop. 5). Notons que la complexité de facteurs maximale peut être atteinte par codage littéral d'une T.S.G. ε -free (cf exemple 1), alors qu'on ne sait pas l'atteindre par T.S.G. seule. L'étude de la classe des T.S.G. paraît toutefois intéressante, même sans codage, celle-ci contenant la classe des Uniform Tag Sequences de Cobham.

La section 4 étudie la complexité de facteurs des T.S.G. De ce point de vue:

– la restriction au temps réel n'implique pas de limitation: la suite $\sigma_0 = 010111001101\dots$ des entiers consécutifs en notation binaire renversée, de complexité de facteurs maximale, par exemple, est constructible en temps réel;

– l'itération par morphisme est de complexité quadratique [EHR; 6], [PAN; 14];

— la production par Machine à Compteurs est de complexité polynomiale [FIS; 8];

— la production par T.M.G. peut, comme on s'en doute, atteindre des complexités exponentielles (exemples 3 et 4).

Nous montrons que le caractère itératif de la production par Tag Machine Généralisée permet un contrôle efficace de la complexité de facteurs dans le cas uniforme (à rapprocher du cas Uniform Tag Sequences). Nous donnons pour les T.S.G. croissants (prop. 13) et uniformes (prop 14) une borne polynomiale de la complexité de facteurs, que nous montrons être optimale (corollaire 15).

2. TAG SÉQUENCES GÉNÉRALISÉES

Un automate qui a la possibilité de se relire a la puissance du calculable. On se propose d'exploiter cette idée pour produire des suites infinies sur un alphabet fini A . On disposera ainsi d'une classe assez large de « A -suites », qui resteront toutefois suffisamment contrôlables dans leur structure, par le caractère itératif de leur génération.

Une Tag Machine Généralisée (T.M.G.), fonctionnant sur l'alphabet A , se présente comme :

— une bande finie à gauche infinie à droite, ses cases pouvant être occupées par un symbole de A ou « vides » ($\# \notin A$);

— une tête de lecture R et une tête d'écriture W , se déplaçant toutes deux uniquement vers la droite;

— un bloc état.

Plus précisément :

DEFINITION 0 : Une T.M.G. \mathcal{M} est un 5-uplet $\langle A, Q, q_0, \delta, \lambda \rangle$ avec

A , alphabet fini;

Q , ensemble fini d'états;

$q_0 \in Q$, état initial;

δ et λ , fonctions de transition;

$\delta: Q \times A \rightarrow Q$ (transition d'état);

$\lambda: Q \times A \rightarrow A^*$ (règle d'écriture).

Initialement, l'état est q_0 , la bande est occupée par un mot $u \in A^*$ (un symbole par case, depuis la plus à gauche), suivi par des $\#$.

R pointe le premier symbole de u , W le premier symbole $\#$ suivant u .

Un pas de la machine est décrit par la figure 1.

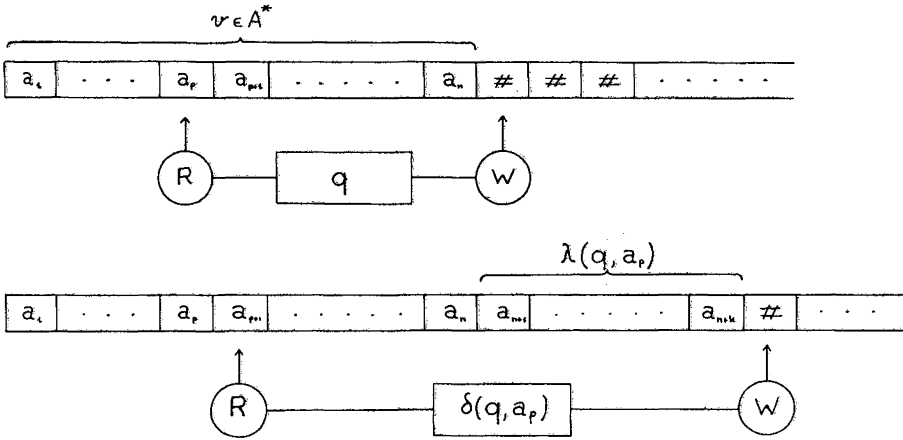


Figure 1

DÉFINITION 1 : Soit \mathcal{M} T.M.G. sur l'alphabet A , $u \in A^*$.

On dit que, sur l'entrée u , \mathcal{M} produit la suite $\mathcal{M}(u) \in A^* \cup A^\omega$ si, à chaque pas de \mathcal{M} , le mot $v \in A^*$ précédant le symbole $\#$ pointé par W est préfixe de $\mathcal{M}(u)$.

Remarque: $\mathcal{M}(u) \in A^*$ ssi une situation de blocage est atteinte (R et W pointant le même symbole $\#$, ou défaut d'instruction).

DÉFINITION 2 : La classe des Tag Machines Généralisées sur A est désignée par T.M.G. (A).

On appelle Tag Sequences Généralisées sur A , et on désigne par T.S.G. (A) la classe des suites infinies $\alpha \in A^\omega$ telles qu'existent \mathcal{M} élément de T.M.G. (A) et $u \in A^*$ vérifiant $\alpha = \mathcal{M}(u)$.

Exemple 1. — Soit $\mathcal{M}_0 = \langle \{0, 1, S\}, \{q_0, q_1\}, q_0, \delta, \lambda \rangle$ la T.M.G. sur $\{0, 1, S\}$ décrite par son graphe de transitions (fig. 2).

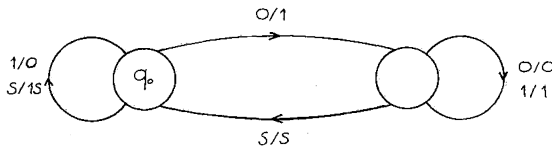


Figure 2

$\mathcal{M}_0(0.S)$ est la suite σ'_0 des entiers consécutifs en notation binaire renversée séparé par S .

$\sigma'_0 = 0S1S01S11S001S\dots$

DÉFINITION 3 : \mathcal{M} désignant une Tag Machine Généralisée sur A ,

(1) on définit le module (respectivement la valuation) de \mathcal{M} par :

$$k_1(\mathcal{M}) = \text{Max} \{ |\lambda(q, a)| / q \in Q, a \in A \}$$

$$\text{(resp. } k_0(\mathcal{M}) = \text{Min} \{ |\lambda(q, a)| / q \in Q, a \in A \}).$$

(2) on dit que \mathcal{M} est une Tag Machine ε -free (T.M. ε (A)) si : $k_0(\mathcal{M}) \geq 1$.

(3) on dit que \mathcal{M} est une Tag Machine Croissante (T.M.C.(A)) si : $k_0(\mathcal{M}) \geq 2$.

(4) on dit que \mathcal{M} est une Tag Machine k -Uniforme (T.M.U. $_k$ (A)) si : $k = k_0(\mathcal{M}) = k_1(\mathcal{M})$.

DÉFINITION 4 : Une suite est dite non dégénérée si elle est infinie et non ultimement constante.

Rappelons que $\alpha \in A^\omega$ est dite ultimement périodique si :

$$\exists u \in A^*, \exists v \in A^+ \quad \text{tels que } \alpha = u.v^\omega.$$

Si de plus $|v| = 1$, α est ultimement constante.

Ces cas particulièrement simples de suites infinies se rattachent par exemple aux définitions précédentes par les remarques :

Remarque 1 : Une suite sur l'alphabet A est finie ou ultimement périodique si et seulement si elle peut être produite par un élément de T.M.G. (A), de module ≤ 1 .

Remarque 2 : Une suite sur l'alphabet A est ultimement périodique si et seulement si elle peut être produite par un élément de T.M.G. (A), uniforme de module 1.

Il sera particulièrement utile d'envisager les T.S.G. comme points fixes d'itérations, aspect inhérent à leur production, qui en fait précisément tout l'intérêt pour l'étude structurelle [PAN; 15].

DÉFINITION 5 : On appelle transducteur sur l'alphabet A une Machine Séquentielle Généralisée Déterministe (D.G.S.M.),

$$\mathcal{T} : \langle X, Y, Q, q_0, \delta, \lambda \rangle \quad [\text{EIL; 7}], [\text{HOP; 11}],$$

de même alphabet d'entrée et de sortie $X = Y = A$, Q désignant l'ensemble fini des états, q_0 l'état initial, δ et λ les fonctions de transition :

$$\delta : Q \times A \rightarrow Q, \quad \lambda : Q \times A \rightarrow A^*.$$

Les fonctions de transition se prolongent naturellement, de $Q \times A^*$ dans Q par $\delta(q, ua) = \delta(\delta(q, u), a)$ et de $Q \times A^*$ dans A^* (respectivement de $Q \times A^\omega$ dans $A^* \cup A^\omega$) par $\lambda(q, ua) = \lambda(q, u) \cdot \lambda(\delta(q, u), a)$.

DÉFINITION 6 : On appelle Point Fixe de Transduction sur l'alphabet A toute suite infinie $\alpha \in A^\omega$ pour laquelle existe un transducteur \mathcal{T} sur A transformant tout préfixe u de α appartenant à A^+ en un autre préfixe $\lambda(q_0, u)$ strictement plus long de α .

On désigne par P.F.T. (A) la classe de ces suites.

Remarque 3. — Les éléments α de P.F.T. (A) sont ainsi obtenus par itération de transducteur \mathcal{T} sur leur premier symbole [PAN; 15].

Si $\alpha \in \{a\} \cdot A^\omega$, le transducteur \mathcal{T} vérifie $\lambda(q_0, a) \in \{a\} \cdot A^+$, et on a $\alpha = \lim u_n$ en posant :

$$\begin{aligned} u_0 &= a \\ &\vdots \\ u_{n+1} &= \lambda(q_0, u_n), \quad \forall n \geq 0. \end{aligned}$$

Naturellement, la génération même de Tag Sequence ou de Point Fixe de Transduction permet facilement de vérifier :

PROPOSITION 1 : *Les Tag Sequences généralisées sur A sont exactement les Points Fixes de Transduction sur A .*

La vérification consiste à aménager le départ des constructions.

Il semble clair que le concept envisagé soit indépendant des perturbations finies :

DÉFINITION 7 : α et α' sont égales à une perturbation finie près si existent u et u' dans A^* , et w dans A^ω tels que $\alpha = u \cdot w$ et $\alpha' = u' \cdot w$.

PROPOSITION 2 : *La classe P.F.T. (A) (= T.S.G. (A)) est stable par perturbations finies.*

La vérification est immédiate.

Néanmoins, la distinction paraît s'assumer différemment suivant les particularités de la production.

Ainsi, de même que l'on peut considérer les Tag Sequences Généralisées ε -free (T.S. ε), croissantes (T.S.C.), k -uniformes (T.S.U $_k$), obtenues par Tag Machines correspondantes, on peut envisager les Points Fixes de Transduction ε -free (P.F. ε), croissants (P.F.C.), k -uniformes (P.F.U $_k$) obtenus par transducteur correspondant.

On a vu : T.S.G. = P.F.T. (prop. 1).

Les relations

$$\text{T.S.}\varepsilon = \text{P.F.}\varepsilon \quad \text{et} \quad \text{T.S.C.} \subset \text{P.F.C.}$$

se vérifient aussi aisément, mais nous ne connaissons pas d'autres inclusions.

Aux perturbations finies près, apparaissent en plus les relations

$$\text{T.S.C.} \equiv \text{P.F.C.} \quad \text{et} \quad \text{P.F.U.} \simeq \text{T.S.U.},$$

la dernière relation s'interprétant par : chaque élément de P.F.U. est perturbation finie d'un élément de T.S.U.

Seules, actuellement, quelques remarques partielles ont pu être notées :

- (1) $\text{T.S.U.}_2 \equiv \text{P.F.U.}_2$ (aux perturbations finies près).
- (2) $\text{T.S.C.} = \text{P.F.C.}$ si et seulement si $\text{P.F.U.}_2 \subset \text{T.S.C.}$

Jusqu'où s'étend cette classe, qui prolonge naturellement la classe des suites points fixes de morphismes itérés? Ce n'est certainement pas la classe de toutes les suites, pour des raisons évidentes de cardinalité, ni celle des suites constructibles, pour des raisons de complexité de temps de construction (les T.S.G. sont constructibles en temps réel) ou de présentation récursive (les T.S.G. sont récursivement présentables, par les suites constructibles). On se propose d'étudier les propriétés de ces suites, Points Fixes de Transduction.

3. COMPLEXITÉ DE PRODUCTION

Pour α une A -suite et B une partie de A , on désigne par $\alpha \upharpoonright B$ la B -suite obtenue par restriction d'alphabet, c'est-à-dire l'image de α par l'homomorphisme $h: A \rightarrow B$ qui efface les lettres de $A-B$.

- $r_\alpha^B(n)$ désigne la fonction indice de n -ième occurrence d'une lettre de B dans α ;
- $\pi_\alpha^B(n)$ désigne la section réciproque de r_α^B : nombre de lettres de B dans le $(n+1)$ -préfixe $\alpha_0\alpha_1 \dots \alpha_n$ de α .

Une suite α est dite constructible en temps f [HAR; 10] s'il existe une Machine de Turing M qui écrit sur une bande supplémentaire de sortie, sans jamais effacer ni revenir en arrière, les termes de α , le n -ième symbole étant écrit au plus tard au $(f(n))$ -ième pas de M .

On désigne par $\mathbb{S}(f)$ la classe des suites constructibles en temps f , par \mathbb{S} celle des suites Constructibles en Temps Réel (C.T.R.): $\mathbb{S} = \mathbb{S}(\text{id})$.

Les suites C.T.R. sont exactement les suites constructibles en temps linéaire [HAR; 10].

Les suites P.F.T. sont évidemment Constructibles en Temps Réel, et plus généralement :

PROPOSITION 3 : Si $\alpha \in \text{P.F.T.}(A)$ et $B \subset A$, $\alpha \upharpoonright B$ est constructible en temps r_α^B .

Preuve : La suite α étant construite en temps réel, un symbole par pas, l'indice de n -ième occurrence de B correspond à un nombre de pas permettant de fournir ce n -ième symbole de la suite $\alpha \upharpoonright B$.

Plus généralement, si $\alpha \in \mathbb{S}(f)$ alors $\alpha \upharpoonright B \in \mathbb{S}(f \circ r_\alpha^B)$.

La classe des suites Constructibles en Temps Réel fournit par image homomorphe toutes les suites constructibles. Il suffit par un homomorphisme « d'effacer la trace des pas inutiles ». La classe des suites Points Fixes de Transduction, bien que plus restreinte (corollaire 8), possède encore cette propriété.

PROPOSITION 4 : Toute suite constructible est image homomorphe d'une suite P.F.T.

Il suffit de vérifier qu'une Machine de Turing produisant une suite infinie peut être simulée par une Tag Machine Généralisée utilisant une alphabet auxiliaire de description.

PROPOSITION 5 : Soit α une suite de la classe P.F.T. (A), h un homomorphisme défini sur A^* , de support de densité non nulle dans α ; alors $h(\alpha)$ est constructible en temps réel.

Preuve : Soit B la partie de A support de h , $B = \{a \in A / h(a) \neq \varepsilon\}$: $h(\alpha)$ est évidemment constructible en temps linéaire par rapport au temps de construction de $\alpha \upharpoonright B$.

Si la densité inférieure de B dans α , $\liminf \pi_\alpha^B(n)/n$ est non nulle, il existe $\theta > 0$ tel qu'à partir d'un rang N on ait $\pi_\alpha^B(n)/n > \theta$ pour tout n , et donc aussi $r_\alpha^B(k) < k/\theta$ pour tout k dès que $r_\alpha^B(k) \geq N$.

$\alpha \upharpoonright B$, et donc aussi $h(\alpha)$, sont constructibles en temps linéaire.

On a la réciproque avec la limite supérieure non nulle si r_α^B est borné linéairement.

En particulier, si la densité d de B dans α existe

$$\left(d = \liminf \frac{\pi_\alpha^B(n)}{n} = \limsup \frac{\pi_\alpha^B(n)}{n} \right),$$

elle est non nulle si et seulement si r_α^B est linéairement borné.

Ces observations justifient tout à fait la production de certaines suites C.T.R. comme suites P.F.T., de la même manière que certaines suites sturmiennes sont naturellement produites comme points fixes d'homomorphisme [COB; 4]. En particulier, les Σ -suites images homomorphes de suites α , [α dans P.F.T. ($\Sigma \cup T$)], obtenues par effacement des lettres de l'alphabet de travail T , sont constructibles en temps réel lorsque la densité de T est nulle dans α .

La suite σ_0 par exemple (§ 1), produite à partir du pont fixe σ'_0 (exemple 1), limite le temps et l'espace au seul usage d'un séparateur de densité nulle dans σ'_0 . Remarquons qu'elle peut aussi s'obtenir comme image par homomorphisme *non effaçant* d'une suite P.F.T. sur $\{0, 1, S_0, S_1\}$, les séparateurs S_0 et S_1 prévoyant la place respective d'un 0 ou d'un 1.

La fonction r_α^B n'a évidemment plus de signification intéressante sur la complexité temps, dans le cas où B est réduit à un seul élément ($B = \{b\}$). Elle garde néanmoins toute sa signification structurelle puisqu'elle mesure les longueurs des facteurs sans occurrence de b ; de même la fonction $r_\alpha^{A-\{b\}}$ mesure les longueurs des facteurs « constants de b » (i. e. : appartenant à $\{b\}^+$) : notons C_α^b cette dernière fonction, pour l'utiliser dans la propriété suivante.

PROPOSITION 6 : Soit α dans P.F.T. (A), de module k , non dégénérée (déf. 4, § 2) et $a \in A$. Alors

$$\overline{\lim} \frac{C_\alpha^a(n+1)}{C_\alpha^a(n)} \leq k.$$

La démonstration s'appuie en partie sur le lemme suivant :

LEMME 7 : Soit α un Point Fixe de Transducteur à N états, α non dégénérée. Un facteur de longueur supérieure ou égale à N et son image ne peuvent se trouver à l'intérieur d'un même facteur constant.

Preuve : Ce lemme généralise une forme utilisée par Cobham pour les Uniform Tag Sequences : si un point fixe α d'homomorphisme est non dégénéré, un facteur et son image ne peuvent se trouver dans un même facteur constant. Le lemme se vérifie facilement par l'absurde. En effet, supposons $vb = a^m b$ facteur de α , u et son image $\mathcal{F}(u)$ facteurs de v , avec $|u| \geq N$. Alors b n'a pu être écrit qu'à la lecture d'une occurrence de a apparaissant dans v , et placée au-delà du facteur u . Ceci est contradictoire,

puisque à la lecture de ua , le transducteur \mathcal{T} n'a pu qu'entrer dans une boucle d'états, n'écrivant que des a tant qu'il lit des a .

Vérifions la proposition 6 :

Soit N le nombre d'états du transducteur \mathcal{T} dont α est point fixe.

Soit $\alpha_{[p+1, m]} = a^m$ ($m > N$) un facteur, constant de a , maximal

$$(i. e. : \alpha_p \neq a \text{ et } \alpha_{p+m+1} \neq a),$$

et donc

$$C_\alpha^a(n) = p \Rightarrow C_\alpha^a(n+1) = p + m + 1.$$

Montrons par l'absurde que, nécessairement :

$$m < (k-1)p + k(N+1) - 1.$$

Soit α_r le terme de α dont la lecture par \mathcal{T} écrit le facteur $\alpha_q \alpha_{q+1} \dots \alpha_{q+l}$ ($l \geq 0$) contenant α_{p+m+1} .

On a $q > p + m + 1 - k$, et donc $kr \geq q > p + m + 1 - k$.

Dans l'hypothèse où $m \geq (k-1)p + k(N+1) - 1$, on aurait donc $r > p + N$, et le facteur $\alpha_{[p+1, N]}$ se trouverait dans l'hypothèse interdite par le lemme.

Ainsi $C_\alpha^a(n+1) < k C_\alpha^a(n) + k(N+1)$ ce qui, compte tenu de l'hypothèse α non dégénérée, fournit le résultat.

Remarquons que cette borne est optimale : on vérifie que pour le transducteur uniforme (fig. 3) :

on a

$$\lim_{n \rightarrow +\infty} \frac{C_\alpha^1(n+1)}{C_\alpha^1(n)} = \lim_{n \rightarrow +\infty} \frac{r_\alpha^{(0)}(n+1)}{r_\alpha^{(0)}(n)} = k.$$

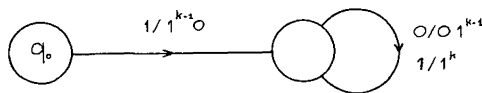


Figure 3

COROLLAIRE 8 : *Les suites Points Fixes de Transduction ne constituent pas toutes les suites Constructibles en Temps Réel.*

En effet, la suite de Liouville $\lambda = (\lambda_i)_{i \in \omega}$, avec $\lambda_i = 1$ si i est de forme $n!$, $\lambda_i = 0$ sinon, est constructible en temps réel [HAR; 10], et

$$\lim \frac{C_\lambda^0(n+1)}{C_\lambda^0(n)} = +\infty.$$

La proposition 5 ne semble pas fournir une caractérisation des suites C.T.R. Il serait intéressant de savoir quel type d'homomorphisme s'appliquant sur des P.F.T. donne exactement la classe \mathbb{S} des suites C.T.R.

Signalons d'autres critères, relatifs aux longueurs de facteurs constants dans les P.F.T., et conduisant à un « gap ».

Dans les deux propositions qui suivent, α désigne un point fixe de transducteur \mathcal{T} de module k , à N états.

PROPOSITION 9 : *Soit w un facteur constant de α tel que $|w| > N$. Désignons par $\lambda(w)$ son image par \mathcal{T} dans α .*

La longueur de $\lambda(w)$ ne peut appartenir à l'intervalle

$$\left] k(N-1), \frac{|w|+1}{N} - 1 \right[.$$

Preuve : Puisque $|w| > N$, la lecture de w conduit \mathcal{T} à entrer dans une boucle d'états, de longueur $p_2 \geq 1$, après une « entrée en boucle » de longueur p_1 ($p_1 + p_2 \leq N$).

Si on note p_3 la longueur de boucle inachevée de fin de lecture de w , on a $|w| = p_1 + qp_2 + p_3$, et donc $qp_2 = |w| + (p_2 - p_3) - (p_1 + p_2) \geq |w| + 1 - N$; ainsi

$$q \geq \frac{|w|+1}{N} - 1.$$

Si l'écriture durant la lecture en boucle d'états est non vide, alors $|\lambda(w)| \geq q$, sinon l'écriture ne peut que se produire à la lecture du préfixe de longueur p_1 de w , et $|\lambda(w)| \leq k(N-1)$.

PROPOSITION 10 : *Soit u un facteur de α constitué de r ($r \geq 2$) facteurs constants de longueurs supérieures ou égales à L , et espacés deux à deux par au plus p symboles.*

L'image de u ne peut alors contenir un facteur constant maximal de longueur appartenant à l'intervalle $]kr(p+N), (L/N) - 1[$.

Preuve : Désignons par w_1, w_2, \dots, w_r les r facteurs constants envisagés. Soit a^m un facteur constant maximal de l'image.

Supposons $m < (L/N) - 1$. On ne perd pas en généralité en supposant que l'écriture de a^m commence pendant la lecture de w_1 .

Compte tenu de la proposition 9, la contribution de u à l'écriture de a^m a pu être faite au plus par :

- (1) les $(r - 1)$ espaces séparant les w_i ;
- (2) le préfixe de longueur $N - 1$ de w_1 ou le suffixe de longueur N de w_1 ;
- (3) le préfixe de longueur $N - 1$ de w_2, w_3, \dots, w_{r-1} ;
- (4) le préfixe de longueur N de w_r ;

et donc : $m \leq k [(r - 1)p + 2N + (r - 2)(N - 1)]$.

Exemple 2 : En désignant par \bar{n} l'écriture binaire renversée de l'entier n , et par u_p le mot binaire $\bar{p}.1^{2^p}$, considérons la suite infinie $\beta = u_0 u_1 u_2 \dots$.

Pour $p < m$ et $q = 2^m - 2^p$, le facteur

$$\beta(p, m) = u_q u_{q+1} \dots u_{2^m-1} = \underbrace{0^p 1^{m-p} 1^{2^q}}_{u_q} \dots$$

se compose de $r = 2^p$ plages constantes de longueurs supérieures ou égales à $L = 2^q + m - p$.

Supposons β point fixe de transducteur (module k, N états). Si le facteur $v = 0^m$, préfixe de u_{2^m} qui suit $\beta(p, m)$ est écrit à partir de $\beta(p, m)$, et si $m < (L/N) - 1$, compte tenu que le préfixe 0^p de $\beta(p, m)$ n'a participé au plus qu'à l'écriture de kp symboles de v , et que v lui-même n'a participé au plus qu'à l'écriture de kN de ses symboles (lemme 7), la proposition 10 assure :

$$m \leq k \cdot 2^p (p + N) + k (p + N).$$

Ainsi : $\forall p, \exists m_0, \forall m \geq m_0$ le préfixe 0^m de u_{2^m} doit commencer à être écrit par des symboles précédant $\beta(p, m)$.

COROLLAIRE 11 : *Les suites P.F.T. ne sont pas stables par image d'homomorphisme non effaçant.*

Preuve : La suite β de l'exemple 2 fournit un contre-exemple. Elle est en effet obtenue par codage non effaçant ($h(0) = 0, h(1) = h(S) = 1$), à partir de $\alpha = 0S^{2^0}1S^{2^1}01S^{2^2} \dots \bar{n}S^{2^n} \overline{n+1} \dots$, elle-même point fixe de transducteur (par simple modification de \mathcal{M}_0 , exemple 1).

Il nous suffit, compte tenu des observations faites sur β de vérifier qu'il existe m_0 tel que, pour $m \geq m_0$, le facteur 0^m préfixe de u_{2^m} commence son

écriture depuis un facteur $u_n u_{n+1} \dots u_{2^m-1}$, avec

$$n \geq 2^m - \text{Log}_2(k+1).$$

Pour cela, désignons par :

$i(m)$ l'indice de β_i si l'image de β_i contient le premier zéro de u_{2^m} ;

$j(m)$ l'indice de β_j si β_j est le premier symbole de cette image;

$n(m)$ l'indice de u_n dans lequel apparaît $\beta_{i(m)}$.

On note i, j, n dans ce qui suit.

Si $H_t = \sum_{p=0}^{p=t} |\bar{p}|$, un simple dénombrement fournit :

$$i \leq 2^{n+1} - 1 + H_n, \quad j \geq 2^{2^m} + H_{2^m-1} - k;$$

ce qui, après majoration (respectivement minoration) grossière de H_n (de H_{2^m-1}) donne

$$i < 2^{n+1} + m 2^m, \quad j \geq 2^{2^m} + 1 - k.$$

Puisque k est la borne supérieure des longueurs de mots écrits à chaque pas, on a $j \leq ki$, soit :

$$2^{2^m} + 1 - k \leq k 2^{n+1} + km 2^m.$$

Par un calcul élémentaire, l'hypothèse :

$$\forall m_0, \exists m \geq m_0 \text{ t. q. } n+1 \leq 2^m - \text{Log}_2(k+1)$$

conduit à une contradiction.

Rappelons qu'une Uniform Tag Sequence [COB; 4] est l'image homomorphe d'un point fixe d'itération de morphisme uniforme. Une U.T.S. est donc construite par un Tag Système de module constant k , $\langle \Gamma, \beta_0, g, h, A \rangle$ avec

$$g : \Gamma \rightarrow \Gamma^k, \quad g(\beta_0) = \beta_0 u,$$

et $h : \Gamma \rightarrow A$ un codage littéral. Il est utile dans le cas de production uniforme de contrôler les facteurs par la notion de descendants utilisée par Cobham :

DÉFINITION 8 : Rappelant la notation $\alpha_{[t, p]} = \alpha_t a_{t+1} \dots \alpha_{t+p-1}$, pour une A -suite α , on appelle *descendant de m -ième génération* de module k du symbole α_i , et on désigne par $\text{desc}_k(\alpha, m, i)$, le facteur $\alpha_{[ik^m, k^m]}$ de α .

On notera $\text{desc}(\alpha, m, i)$ ou $\text{desc}(m, i)$, lorsqu'il n'y aura pas de confusion possible.

PROPOSITION 12 : *Les Uniform Tag Sequences sur l'alphabet A appartiennent à la classe P.F.T. (A).*

Preuve : Soit $\alpha = h(\beta)$ une U.T.S. construite par $\langle \Gamma, \beta_0, g, h, A \rangle$ de module k . Par le théorème 1 de Cobham, caractérisant les U.T.S. de module k , existent deux entiers $r \geq 0$ et $q > r$ satisfaisant : $\forall i, j$ entiers,

$$\text{desc}_k(\alpha, r, i) = \text{desc}_k(\alpha, r, j) \Rightarrow \text{desc}_k(\alpha, q, i) = \text{desc}_k(\alpha, q, j).$$

Considérons le nouvel alphabet \hat{A} , compression de A , ci-dessous :

$$\hat{A} = \{ \text{desc}_k(\alpha, r, i) / i \in \mathbb{N} \} \subset A^{kr}.$$

On peut alors définir :

(1) un morphisme G de \hat{A}^* dans $(\hat{A}^{kq-r})^*$, de façon non ambiguë, en posant :

$$G(\text{desc}(r, i)) = \text{desc}(q, i),$$

(2) un codage de décompression \hat{h} , de \hat{A} dans A^{kr} , en posant :

$$\hat{h}(\text{desc}(r, i)) = \alpha_{[ik^r, k^r]}.$$

La \hat{A} -suite $\hat{\alpha}$, point fixe de G , est obtenue depuis α par compression d'alphabet, et $\hat{h}(\hat{\alpha}) = \alpha$ [PAN; 14]. La structure même de ce point fixe d'homomorphisme $\hat{\alpha}$ sur l'alphabet \hat{A} permet alors d'obtenir la A -suite α comme point fixe d'un transducteur \mathcal{T} . Celui-ci progressera sur chaque facteur $\alpha_{[ik^r, k^r]}$ par ε -transition, de manière à n'écrire qu'après avoir identifié complètement ce facteur, la lecture de $\alpha_{(i+1)k^r-1}$ permettant l'écriture de $\text{desc}(q, i) = \alpha_{[ik^q, k^q]}$ en bout de bande.

4. COMPLEXITÉ DE FACTEURS

Soit α une A -suite; on désigne par $B(\alpha, n)$ l'ensemble des facteurs de longueur n apparaissant dans α et par $P_\alpha(n)$ le cardinal de $B(\alpha, n)$. On appelle P_α la fonction complexité de facteurs de α : $P_\alpha(n)$ est compris entre la fonction constante 1 (suite constante) et l'exponentielle r^n si r désigne le cardinal de l'alphabet A (suite σ_0 du paragraphe 1 par exemple). Désignons par $CF(f(n))$ la classe des A -suites de complexité de facteurs majorée par

f , et par $CF\mathbb{P} = \bigcup_{p \in \mathbb{N}} CF(n^p)$ la classe des A -suites de complexité de facteurs polynomiale.

Dans tout ce paragraphe, sauf mention contraire, \mathcal{T} désignera un transducteur à N états ($\mathcal{T} : \langle \mathcal{A}, Q, q_0, \delta, \lambda \rangle$; $\text{card } Q = N$), avec $0 \in A$ et $\alpha = \lambda(q_0, \alpha) \in \{0\}$. A^ω un point fixe de \mathcal{T} .

Suivant la nature de \mathcal{T} , la complexité de facteurs de α peut considérablement varier. La croissance exponentielle est atteinte pour certains \mathcal{T} (exemple 3), que l'on peut même choisir ε -free (exemple 4).

Exemple 3 : Soit \mathcal{T}_1 le transducteur défini sur $A = \{0, 1\}$ par son graphe de transitions (fig. 4), et $\sigma_1 = \lambda(q_0, \sigma_1)$ son point fixe.

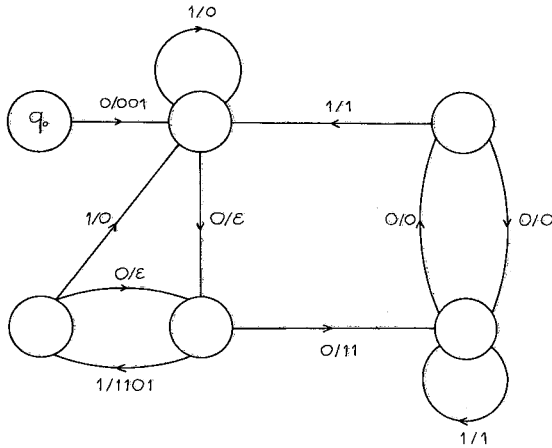


Figure 4

En désignant par \bar{n} l'écriture binaire renversée redoublée de l'entier n ($\bar{0} = 00, \bar{1} = 11, \bar{2} = 0011 \dots$), on vérifie :

$$\sigma_1 = 001 \bar{1} 01 \bar{2} 01 \dots \bar{n} 01 \overline{n+1} \dots$$

et existent deux constantes C_1 et C_2 strictement positives satisfaisant :

$$C_1 n (\sqrt{2})^n \leq P_{\sigma_1}(n) \leq C_2 n (\sqrt{2})^n.$$

Exemple 4 : Soit \mathcal{T}_2 sur $A = \{d, 0, 1, S\}$ défini par son graphe de transitions (fig. 5), et σ_2 son point fixe débutant par d .

$$\sigma_2 = d \bar{0} S \bar{1} S \bar{2} S \dots \bar{n} S \overline{n+1} \dots,$$

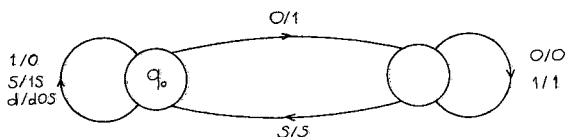


Figure 5

et existent deux constantes D_1 et D_2 strictement positives satisfaisant :

$$D_1 n 2^n \leq P_{\sigma_2}(n) \leq D_2 n 2^n.$$

Remarque 4 : Les exemples qui précèdent, bâtis sur la fonction successeur, pourraient s'étendre en écriture p -aire et fourniraient une complexité en $O(n(\sqrt{p})^n)$ pour σ_1 sur p lettres et en $O(np^n)$ pour σ_2 sur deux états et $p+2$ lettres. σ_2 n'est qu'une perturbation finie de la suite σ'_0 de l'exemple 1, et c'est la forme point fixe qui oblige ici à introduire soit une lettre de départ si on veut garder deux états, soit un état de début si on veut se limiter à trois lettres. On peut en particulier, suivant l'exemple 1, atteindre la complexité $O(np^n)$ avec deux états et $p+1$ lettres.

On se propose dans la suite de montrer que le cas \mathcal{T} croissant, aussi bien que le cas \mathcal{T} uniforme, recouvrent la classe $CF\mathbb{P}$ de complexité polynomiale (prop. 14, [PAN; 15] et [BLE; 2]). Pour le cas croissant le résultat est obtenu par J. J. Pansiot dans la référence précédente. Nous en donnons une autre démonstration, améliorant légèrement la borne. Nous montrons que pour tout entier p existe un transducteur uniforme \mathcal{T} de point fixe $\alpha \notin CF(n^p)$ (corollaire 15).

Pour $J \in \mathbb{N}$, on appelle j -ancêtre minimal de $\alpha_{[t, p]}$, le facteur $\alpha_{[t', p']}$ de longueur minimale dont $\alpha_{[t, p]}$ soit facteur de l'image de J -ième génération dans α .

PROPOSITION 13 : *Si α est point fixe de transducteur croissant \mathcal{T} de valuation k_0 sur N états, alors :*

$$CF(\alpha) = O(n^{1 + \log_{k_0} N}).$$

Preuve : Pour chaque u de $B(\alpha, n)$, avec $n \geq 2k_1 + 1$, soit j le plus petit entier pour lequel la première occurrence de u dans α possède un j -ancêtre minimal de longueur inférieure ou égale à $2k_1$, et soit β cet ancêtre.

Par ce choix,

- (1) $|\beta| \leq 2k_1$;
- (2) $\beta = a_1 \beta' a_2$ avec $\beta' \in A^+$, $a_1 \in A$, $a_2 \in A$;
- (3) l'image I de j -ième génération de β' dans α est facteur strict de u ; ainsi $k_0^j \leq |I| < n$, et donc $j \leq \lceil \log_{k_0} n \rceil$.

Le nombre des u de $B(\alpha, n)$ auxquels peuvent être associés ce β et ce j est majoré par nN^j . Ainsi existe une constante $C > 0$ telle que

$$P_\alpha(n) \leq C \cdot n(1 + N + N^2 + \dots + N^{\lceil \log_{k_0} n \rceil}),$$

d'où le résultat si $N > 1$.

Remarquons que la majoration de $P_\alpha(n)$ pour $N=1$ fournit la borne optimale dans le cas des DOL-langages croissants (GDOL) [EHR; 6].

Considérons les transducteurs uniformes \mathcal{T} sur $A = \{0, 1\}$ de module k , satisfaisant les conditions (C1) et (C2).

(C1) Les fonctions δ et λ vérifient :

$$(1) \quad \forall i \in \{0, 1, \dots, N-1\},$$

$$\delta(q_i, 1) = q_i, \quad \delta(q_i, 0) = q_{(i+1) \bmod N}$$

$$\lambda(q_i, 1) = 1^k, \quad \lambda(q_i, 0) \in \{0\} A^{k-1}.$$

$$(2) \quad \forall q, q' \in Q, \quad q \neq q' \rightarrow \lambda(q, 0) \neq \lambda(q', 0).$$

(C2) Si g_i est le nombre d'occurrences de 0 dans $\lambda(q_i, 0)$

$$K = \sum_{i=0}^{i=N-1} g_i \equiv 1 \pmod N.$$

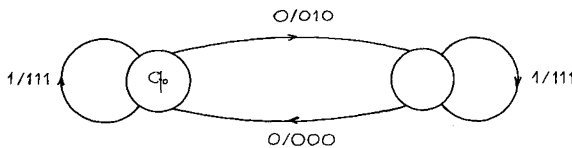


Figure 6

La figure 6 fournit un tel exemple.

Remarques : Soit α point fixe d'un transducteur vérifiant (C1) et (C2).

(1) $\lambda(q_0, 0)$ doit contenir au moins une occurrence de 0 autre que la première si on veut un point fixe α non constant.

(2) la condition (C 1) nécessite $N = \text{card } Q \leq 2^{k-1}$.

(3) la condition (C 1) implique l'injectivité de λ étendue à $Q \times (A^+ - \{1\}^*)$.

(4) une des conséquences essentielles de ces conditions réside dans la propriété suivante (lemme 4 de [BLE; 2]) : si un facteur u a une occurrence dans α , il a dans α des occurrences lues dans chacun des états de \mathcal{F} .

PROPOSITION 14 : *Pour tout transducteur uniforme \mathcal{F} , de module $k \geq 2$ satisfaisant les conditions (C 1) et (C 2), existe une constante $C > 0$ telle que le point fixe $\alpha = \lambda(q_0, \alpha) \in \{0\} A^\omega$ de \mathcal{F} vérifie :*

$$\forall n \geq k, P_\alpha(n) \geq C \cdot n^{1 + \log_k N}.$$

Considérons l'ensemble $B'_n = (\{1\} \cdot A^{n-1} - \{1\}^n) \cap B(\alpha, n)$, $P'(n)$ son cardinal, et minorons $P'(km)$ en fonction de $P'(m)$.

On dénombre d'abord $N \cdot P'(m)$ blocs distincts de B'_{km} , du fait de l'injectivité de λ (remarque 3) et de la remarque 4.

Soit \bar{B}'_{km} l'ensemble de ces $N \cdot P'(m)$ blocs. Pour $x \in \bar{B}'_{km}$ existe un entier i_x compris entre 1 et $m-1$ tel que $x = 1^{k \cdot i_x} \cdot 0 \cdot u_x$; associons à x une de ses occurrences dans α : $f(x) = \alpha_{[s, km]}$, et soit, pour $j \leq k-1$, le facteur

$$\tau_j(\alpha_{[s, km]}) = \alpha_{[s+j, km]} \text{ de } \alpha.$$

Il résulte des hypothèses que l'application $(j, x) \rightarrow \tau_j(f(x))$, de $\{0, 1, \dots, k-1\} \times \bar{B}'_{km}$ dans B'_{km} est une injection.

En effet, soient (j, x) et (j', x') distincts. Alors, les blocs $\tau_j(f(x))$ et $\tau_{j'}(f(x'))$ diffèrent par le préfixe précédant la première occurrence de 0 lorsque $i_x \neq i_{x'}$ ou lorsque $i_x = i_{x'}$ et $j \neq j'$; ils diffèrent comme ayant des facteurs distincts u_x et $u_{x'}$ à même distance de l'origine

lorsque $i_x = i_{x'}$ et $j = j'$.

Ainsi, $P'(km) \geq k N P'(m)$ pour tout $m \geq 1$ et donc, si $[.]$ désigne la fonction partie entière, on a

$$P_\alpha(n) \geq P'(n) \geq (k N)^{[\log_k n] - 1} \cdot P'(k)$$

ce dernier terme se minorant aisément par :

$$\frac{P'(k)}{k^2 N} \cdot n^{1 + \log_k N}.$$

COROLLAIRE 15 : *Pour tout entier p existe une suite α de la classe PFTU(A) n'appartenant pas à CF(n^p).*

Il nous suffit de vérifier que les conditions (C 1) et (C 2) sont compatibles avec la contrainte

$$\log_k N = \frac{\log N}{\log k} \geq p.$$

Soit $k \geq 3$ et une numérotation :

$$u_0 = 0^{k-1}, \dots, u_1, u_2, \dots, u_{2^{k-1}-3}$$

des éléments de A^{k-1} autres que 1^{k-1} et $0^{k-2} \cdot 1$.

On satisfait facilement (C 1) en prenant :

$$Q = \{q_0, q_1, \dots, q_{2^{k-1}-3}\} \quad (\text{et donc } N = 2^{k-1} - 2),$$

$$\lambda(q_i, 0) = 0 \cdot u_i, \quad \delta(q_i, 0) = q_{(i+1) \bmod N}.$$

Un simple dénombrement fournit K , nombre total des 0 figurant dans les $\lambda(q_i, 0)$: $K = 2^{k-2}(k+1) - k = ((k+1)/2)N + 1$, qui satisfait la condition (C 2) en se limitant aux k impairs.

Donc, $\log_k N = \log(2^{k-1} - 2) / \log k$ peut être choisi plus grand que p , par un k impair suffisamment grand.

L'incidence du nombre N d'états sur la complexité de facteurs, dans les suites infinies obtenues par processus itératif, apparaît comme suit :

		Cas général	ε -free	Croissant	Uniforme
$k_1 \leq 1$		Le cardinal N de Q , ou l'uniformité, n'ont pas d'incidence sur la complexité, dégénérée en $O(1)$.			
	$N = 1$	$O(n^2)$		$O(n \log n)$	$O(n)$
$k_1 \geq 2$	$N \geq 2$	La complexité exponentielle est atteinte (exemples 3 et 4)		$O(n^{1 + \log_{k_0} N})$	

- Pour $k_1 \leq 1$, résultat évident par les remarques 1 et 2.

- Pour $k_1 \geq 2$ et $N = 1$, les résultats apparaissent dans [EHR; 6], [PAN; 15], [COB; 4].

- La borne polynomiale dans le cas de D.G.S.M. croissant est obtenue dans [PAN; 15].

- Il serait intéressant d'affiner les observations concernant les complexités exponentielles :

- la complexité maximale p^n sur un alphabet à p lettres peut-elle être atteinte ?

– peut-on atteindre la complexité exponentielle sur un alphabet à deux lettres dans le cas ε -free?

– peut-on préciser les liens entre le cardinal de l'alphabet, le nombre des états et la complexité?

INDEX DES ABRÉVIATIONS

C.T.R.	: Constructible en Temps Réel.
U.T.S.	: Uniform Tag Sequence.
T.M.G. (A)	: Tag Machine Généralisée sur l'alphabet A.
T.S.G. (A)	: Tag Sequence Généralisée sur l'alphabet A.
T.M. ε	: Tag Machine ε -free.
T.M.C.	: Tag Machine Croissante.
T.M.U _k	: Tag Machine Uniforme de module k.
P.F.T. (A)	: Point Fixe de Transducteur sur l'alphabet A.
T.S. ε ; P.F. ε	: Tag Sequence ε -free; Point Fixe de Transducteur ε -free.
T.S.C.; P.F.C.	: Tag Sequence Croissante; Point Fixe de Transducteur Croissant;
T.S.U _k ; P.F.U _k	: Tag Sequence k-uniforme; Point Fixe de Transducteur k-uniforme.

Nous tenons à remercier les correcteurs dont les observations, tant sur la forme que sur le fond, ont été particulièrement fructueuses pour ce travail.

BIBLIOGRAPHIE

- [BER; 1] J. BERSTEL, *Mots sans carré et morphismes itérés*, Discrete Mathematics, vol. 29, 1980, p. 235-244.
- [BLE; 2] N. BLEUZEN-GUERNALEC, *Suites points fixes de transduction uniforme*, C. R. Acad. Sci. Paris, t. 300, série I, n° 3, 1985, p. 85-88.
- [CHR; 3] C. CHRISTOL, T. KAMAE, M. MENDES-FRANCE et G. RAUZY, *Suites algébriques, automates et substitutions*, Bull. Soc. Math. France, vol. 108, 1980, p. 401-419.
- [COB; 4] A. COBHAM, *Uniform Tag Sequences*, Math. Syst. Theory, vol. 6, n° 2, 1972, p. 164-192.
- [COV; 5] E. COVEN et G. HEDLUND, *Sequences with Minimal Block Growth.*, Math. Syst. Theory, vol. 7, 1973, p. 138-153.
- [EHR; 6] A. EHRENFUCHT, K. LEE et G. ROZENBERG, *Subword Complexities of Various Classes of deterministic Developmental Languages Without Interaction*, Theor. Comput. Sci., vol. 1, 1975, p. 59-75.
- [EIL; 7] S. EILENBERG, *Automata*, Languages and Machines, Academic Press, volume A.
- [FIS; 8] P. C. FISCHER, A. R. MEYER et A. L. ROSENBERG, *Counter Machines and Counter Languages*, Math. Syst. Theory, vol. 2, 1968, p. 265-283.
- [FIS; 9] P. C. FISCHER, A. R. MEYER et A. L. ROSENBERG, *Time Restricted Sequence Generation*, J. Comp. Syst. Sci., vol. 4, 1970, p. 50-73.
- [HAR; 10] J. HARTMANIS et R. E. STEARNS, *On the Computational Complexity of Algorithms*, Trans. Amer. Math. Soc., vol. 117, 1965, p. 285-306.

- [HOP; 11] J. HOPCROFT et J. ULLMAN, *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley Publishing Company.
- [MIN; 12] M. L. MINSKY, *Computation: Finite and Infinite Machines*, Prentice-Hall, Englewood Cliffs N. J., 1967.
- [PAN; 13] J. J. PANSIOT, *Mots infinis de Fibonacci et morphismes itérés*, RAIRO Informatique Théorique, vol. 17, n° 2, 1983, p. 131-135.
- [PAN; 14] J. J. PANSIOT, *Accessibilité et morphismes itérés*, Thèse publiée par I.R.M.A. Strasbourg, 1983, p. 105-143.
- [PAN; 15] J. J. PANSIOT, *On Various Classes of Infinite Words Obtained by Iterated Mappings*, Lecture Notes in Computer Science, n° 192, p. 188-197.
- [ROS; 16] A. L. ROSENBERG, *Real Time Definable Languages*, J. Assoc. Comput. Mach., 1967, p. 645-662.