

F. G. COUSINEAU

J.-M. RIFFLET

Langages d'interprétation des schémas récursifs

*Revue française d'automatique informatique recherche opérationnelle.
Informatique théorique*, tome 9, n° R1 (1975), p. 21-42

http://www.numdam.org/item?id=ITA_1975__9_1_21_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/>

LANGAGES D'INTERPRETATION DES SCHEMAS RECURSIFS (*)

par F. G. COUSINEAU ⁽¹⁾ et J.-M. RIFFLET ⁽²⁾

Communiqué par M. NIVAT.

Résumé. — Dans une première partie, les langages d'interprétation des schémas récursifs monadiques sont définis et caractérisés. Ils sont ensuite utilisés pour étudier certaines propriétés de clôture de ces schémas et pour ramener leur équivalence d'interprétation à l'équivalence des automates quasi-simples. Dans une deuxième partie, on définit une généralisation des schémas récursifs monadiques en ajoutant la possibilité de tester le résultat de procédures. On montre que cette possibilité augmente la puissance des schémas récursifs monadiques ce qui se traduit notamment par le fait que leurs langages d'interprétation d'algébriques deviennent context-sensitive. On montre également pour ces schémas l'indécidabilité des problèmes d'équivalence forte et d'équivalence d'interprétation, problèmes qui restent ouverts pour les schémas récursifs monadiques ordinaires.

PREMIERE PARTIE : LES SCHEMAS RECURSIFS MONADIQUES

A) Introduction

Notre définition des Schémas Récursifs Monadiques sera celle de Garland et Luckham ([3]). Nous dirons que ces schémas sont fortement équivalents s'ils s'arrêtent pour les mêmes interprétations en donnant le même résultat. Nous dirons qu'ils sont I-équivalents s'ils s'arrêtent pour les mêmes interprétations. Nous noterons respectivement \equiv et \equiv_I ces deux relations. Garland et Luckham associent à chaque schéma son Langage-Valeur (LV) et son

(*) Une partie des résultats contenus dans cet article a fait l'objet d'un mémoire du Groupe d'Informatique Théorique de l'Université Paris VII (mai 1973).

(1) Université Paris VI, Institut de Programmation.

(2) Université Paris VII.

Langage-Valeur Interprété (LVI) qui entretiennent avec l'équivalence forte les relations suivantes :

Étant donnés deux schémas R et S , $R \equiv S \Rightarrow LV(R) = LV(S)$ et $R \equiv S \Leftrightarrow LVI(R) = LVI(S)$. L'utilisation des Langages-Valeur leur permet surtout de montrer des résultats de non-fermeture pour certaines sous-classes de schémas. L'utilisation des Langages-Valeur Interprétés leur permet de montrer la décidabilité de l'équivalence forte pour certaines sous-classes et de la ramener dans le cas général à l'équivalence d'automates déterministes et plus précisément des automates quasi-simples comme nous l'avons montré en [2].

Nous nous intéressons ici à la I -équivalence ou équivalence d'interprétation. Dans le cas de schémas ayant un seul symbole de fonction, nous ramenons cette équivalence à celle d'automates quasi-simples par l'intermédiaire des Langages d'Interprétation (LI) et d'une opération Fr de réduction sur les langages tels qu'étant donnés deux schémas R et S , $R \equiv_I S \Leftrightarrow Fr(LI(R)) = Fr(LI(S))$.

B) Définitions

Étant donnés trois ensembles finis

$$\begin{aligned} \mathcal{B} &= \{ b_1, \dots, b_n \} \text{ ensemble de symboles de fonction de base} \\ \mathcal{P} &= \{ P_1, \dots, P_k \} \text{ ensemble de symboles de prédicat de base} \\ \mathcal{F} &= \{ F_1, \dots, F_l \} \text{ ensemble de variables fonctionnelles} \end{aligned}$$

Un schéma *Récurif Monadique* (S.R.M.) est défini par l équations de la forme $x F_i := \text{si } x P_{r_i} \text{ alors } x \alpha_i \text{ sinon } x \beta_i$ où $i \in [l]$ $P_{r_i} \in \mathcal{P}$ et $\alpha_i, \beta_i \in (\mathcal{B} \cup \mathcal{F})^*$

Une *Interprétation* I d'un S.R.M. S est constituée par :

- (1) un domaine $D_I \neq \emptyset$,
- (2) la donnée pour chaque élément b_i de \mathcal{B} d'une application totale $(b_i)_I$ de D_I dans D_I ,
- (3) la donnée pour chaque élément P_j de \mathcal{P} d'une application totale $(P_j)_I$ de D_I dans $\{0, 1\}$.

Un schéma interprété est un programme qui fournira un calcul si on attribue à la variable une *donnée* initiale, c'est-à-dire un élément de D_I .

Une interprétation I est dite *libre* si

- (1) $D_I = \mathcal{B}^*$
- (2) $\forall w \in \mathcal{B}^* \forall b \in \mathcal{B} (w)(b)_I = wb$

Pour toute interprétation libre, nous considérerons implicitement le mot vide comme donnée initiale.

Le calcul d'un S.R.M. S pour une interprétation I et une donnée d est une suite \mathcal{C} de couples $C_j = (x_j, y_j)$ où $\forall j \in \mathbb{N}$, $x_j \in \mathcal{D}_I$, $y_j \in (\mathcal{B} \cup \mathcal{F})^*$ et qui est définie récursivement de la façon suivante :

- (*) $x_0 = d$
 $y_0 = F_1$
- (*) si $y_j = F_i y'_j$ avec $F_i \in \mathcal{F}$ et $y'_j \in (\mathcal{B} \cup \mathcal{F})^*$ et si $(x_j)(P_{r_i})_I = 0$
alors $x_{j+1} = x_j$ et $y_{j+1} = \beta_i y'_j$
- (*) si $y_j = F_i y'_j$ avec $F_i \in \mathcal{F}$ et $y'_j \in (\mathcal{B} \cup \mathcal{F})^*$ et si $(x_j)(P_{r_i})_I = 1$
alors $x_{j+1} = x_j$ et $y_{j+1} = \alpha_i y'_j$
- (*) si $y_j = b_i y'_j$ avec $b_i \in \mathcal{B}$ et $y'_j \in (\mathcal{B} \cup \mathcal{F})^*$
alors $x_{j+1} = (x_j)(b_i)_I$ et $y_{j+1} = y'_j$
- (*) si $y_j = e$ alors $x_{j+1} = x_j$ et $y_{j+1} = y_j$

On dit que le calcul se termine s'il existe un entier j tel que $y = e$.

Le résultat du calcul que nous noterons $\text{Val}_{S,I}(d)$ est alors x_j . Si le calcul ne se termine pas, le résultat n'est pas défini.

Deux S.R.M. R et S définis sur les mêmes alphabets sont dits *fortement équivalents* (nous noterons $R \equiv S$) si pour toute interprétation I et pour toute donnée d , ou bien les calculs de R et S bouclent tous les deux, ou bien ils se terminent tous les deux et $\text{Val}_{S,I}(d) = \text{Val}_{R,I}(d)$.

Deux S.R.M. R et S définis sur les mêmes alphabets sont dits *I-équivalents* (nous noterons $R \equiv_I S$) si pour toute interprétation I et pour toute donnée d , ou bien les calculs de R et S bouclent tous les deux ou bien ils se terminent tous les deux. Il est bien connu que dans les deux définitions qui précèdent, on peut remplacer la condition « pour toute interprétation » par « pour toute interprétation libre ». Cette propriété permet de ramener les équivalences de schémas à des égalités de langages associés.

C) Les Langages d'interprétation

Le langage d'interprétation d'un S.R.M. peut être défini en toute généralité et permet de donner des démonstrations simples des principales propriétés des S.R.M. (voir [2]). Nous nous contenterons ici de le définir pour les schémas qui nous intéressent c'est-à-dire pour ceux qui ont un seul symbole de fonction ($\mathcal{B} = \{ b \}$). Étant donné un S.R.M. S à un seul symbole de fonction b et une interprétation libre I telle que le calcul de S pour I se termine, le mot d'interprétation $w_{S,I}$ associé à S et I est le mot $\theta_0 \theta_1 \dots \theta_n \in (\{ 0, 1 \}^k)^*$ tel que :

- le mot le plus long effectivement testé au cours du calcul de S pour I est b^n
- $\forall i \leq n \forall p \in [k] (b^i)(P_p)_I = (\theta_i) \pi_p$

k est le nombre de prédicats du schéma et les fonctions π_p sont les fonctions de projection.

Le mot d'interprétation contient l'information minimale à donner sur l'interprétation pour que le calcul puisse être mené à son terme.

Le *Langage d'Interprétation* $LI(S)$ d'un schéma S est l'ensemble des mots d'interprétation $w_{S,I}$ correspondant à S et à toutes les interprétations libres I pour lesquelles le calcul de S se termine.

Nous allons maintenant définir les automates à l'équivalence desquelles nous ramènerons l'équivalence d'interprétation des S.R.M. :

Un automate quasi-simple est un quadruplet $A = \langle X, Y, \delta, y_0 \rangle$ où

- X est un alphabet d'entrée
- Y est un alphabet de pile
- y_0 est le symbole initial de pile
- δ est une application de $X \times Y$ dans $\{0, 1\} \times Y^*$ où les symboles 0 et 1 indiquent respectivement lorsqu'ils figurent dans le membre droit d'une règle appliquée que l'automate doit continuer à lire la lettre qu'il lisait ou progresser d'une lettre.

On étend la fonction de transition δ en une application $\hat{\delta}$ de $X^* \times Y^*$ dans lui-même de la façon suivante :

$$\begin{aligned} \forall x \in X \quad \forall m \in X^* \quad \forall y \in Y \quad \forall \omega \in Y^* \quad \hat{\delta}(xm, y\omega) \\ = (m, z\omega) \text{ si } \delta(x, y) = (1, z) \\ = (xm, e) \text{ si } \omega = e \text{ et } \delta(x, y) = (0, e) \\ = \delta(xm, z\omega) \text{ si } \delta(x, y) = (0, z) \text{ et } \omega z \neq e \end{aligned}$$

Le langage reconnu par l'automate A (langage quasi-simple) est :

$$L(A) = \{ m \in X^+ / \hat{\delta}(m, y_0) = (e, e) \}$$

Il faut noter que ces automates ne réalisent pas des ε -transitions mais ont la vertu de pouvoir rester stationnaires pendant quelque temps, leur comportement dépendant alors de la lettre qu'ils persistent à lire. Comme nous allons le voir, on peut astreindre ces périodes de stationnement à n'intervenir que lorsque la pile décroît. A partir de $\hat{\delta}$, nous définissons une nouvelle fonction de transition δ' par :

$$\begin{aligned} \delta'(x, y) = (0, z) \quad \text{si} \quad \hat{\delta}(x, y) = (x, z) \\ = (1, z) \quad \text{si} \quad \hat{\delta}(x, y) = (e, z) \end{aligned}$$

D'après la définition de $\hat{\delta}$, on voit que $\hat{\delta}(x, y) = (x, z)$ implique $z = e$. δ' est donc une fonction de $X \times Y$ dans $\{(0, e)\} \cup \{1\} \times Y^*$. Nous laissons d'autre part au lecteur le soin de se convaincre que l'automate

$$A' = \langle X, Y, \delta', y_0 \rangle$$

est équivalent à A . Dans tout ce qui suit nous donnerons les fonctions de transition de nos automates sous cette forme.

Les langages quasi-simples forment une sous-classe *stricte* des langages algébriques déterministes et préfixes. On peut montrer par exemple que le langage $L = \{ a^n b a^n b / n \in \mathbb{N} \} \cup \{ a^n c a^n c / n \in \mathbb{N} \}$ n'est pas quasi-simple.

Propriété : Les langages d'interprétation des S.R.M. sont quasi-simples.

Preuve : Nous nous contentons ici d'indiquer la construction d'un automate quasi-simple reconnaissant le langage d'interprétation d'un S.R.M. donné. On pourra trouver la preuve complète en [2].

Nous aurons besoin d'une fonction μ de

$$(\{ b \} \cup \mathcal{F})^* \text{ dans } (\{ b \} \cup \mathcal{F} \cup \mathcal{F}')^*$$

définie comme suit : $\forall w \in \{ b \}^* (w)\mu = e$

$$\text{si } w = uF_j v \text{ avec } v \in \{ b \}^* (w)\mu = uF'_j$$

Étant donné un S.R.M. S défini sur les alphabets de base $\mathcal{B} = \{ b \}$, $\mathcal{P} = \{ P_1, \dots, P_k \}$, $\mathcal{F} = \{ F_1, \dots, F_l \}$ par l équations de la forme :

$x F_i := \text{si } x P_{r_i} \text{ alors } x \alpha_i \text{ sinon } x \beta_i$ avec $\forall i [l] r_i \in [k] \alpha_i, \beta_i \in (\{ b \} \cup \mathcal{F})^*$, nous posons pour tout $i \in [l]$ et $\theta \in \{ 0, 1 \}^k$

$$\begin{aligned} \varepsilon_{i,\theta} &= \alpha_i \quad \text{si } (\theta)\pi_{r_i} = 1 \\ &= \beta_i \quad \text{si } (\theta)\pi_{r_i} = 0 \end{aligned}$$

Le langage d'interprétation du S.R.M. S est reconnu par l'automate quasi-simple $A_S = \langle \{ 0, 1 \}^k, \{ b \} \cup \mathcal{F} \cup \mathcal{F}' \cup \{ F''_1 \}, \delta, F''_1 \rangle$ où la fonction de transition δ est défini par : $\forall \theta \in \{ 0, 1 \}^k, \forall j \in [l]$

$$\begin{aligned} \delta(\theta, F''_1) &= (0, (\varepsilon_{1,\theta})\mu) \\ \delta(\theta, F_j) &= (0, \varepsilon_{j,\theta}) \\ \delta(\theta, F'_j) &= (0, (\varepsilon_{j,\theta})\mu) \quad \text{si } \varepsilon_{j,\theta} \neq e \\ &= (1, e) \text{ sinon} \\ \delta(\theta, b) &= (1, e) \end{aligned}$$

Les symboles de \mathcal{F}' permettent de repérer le dernier symbole de variable fonctionnelle et par conséquent le dernier test effectué au cours d'un calcul de schéma.

Nous allons maintenant examiner les rapports qu'entretiennent les langages d'interprétation avec l'équivalence d'interprétation. Tout d'abord, on constate aisément que deux schémas peuvent être équivalents sans avoir le même langage d'interprétation. Considérons en effet les deux S.R.M. suivants :

- (1) $x F := \text{si } x P \text{ alors } x \text{ sinon } x F$
- (2) $x G := \text{si } x P \text{ alors } x b H \text{ sinon } x G$
 $x H := \text{si } x P \text{ alors } x \text{ sinon } x$

Ils sont I -équivalents puisque leur calcul s'arrête si et seulement si $xP = 1$. Cependant le langage d'interprétation du premier est $\{1\}$ tandis que celui du second est $\{10, 11\}$. Ceci est dû au fait que le second effectue des tests inutiles pour ce qui est de l'arrêt. Notre but va être maintenant de définir une réduction sur les langages d'interprétation qui ôte les facteurs droits de certains mots qui résultent de tests inutiles. Il faut noter que ces tests inutiles ne sont pas nécessairement en nombre borné comme le montre l'exemple suivant :

$xF := \text{si } xP \text{ alors } x \text{ sinon } xbGH$

$xG := \text{si } xP \text{ alors } x \text{ sinon } xbGI$

$xH := \text{si } xP \text{ alors } xb \text{ sinon } xb$

$xI := \text{si } xP \text{ alors } x \text{ sinon } x$

Pour tout $n \geq 1$ le langage d'interprétation de ce schéma contient les mots $0^n 11^n$ et $0^n 11^i 0$ pour tout i . Or, à partir du moment où ce schéma a réalisé au cours de son calcul un test égal à 1, il s'arrêtera nécessairement.

Définition : Étant donné un langage préfixe $L \subset X^*$, la forme réduite $FR(L)$ de L est l'ensemble de tous les mots $w \in X^*$ tels que :

- (1) il existe un code préfixe fini complet C tel que $wC \subset L$,
- (2) pour tout facteur gauche propre w' de w , il n'existe pas de code préfixe fini complet C' tel que $w'C' \subset L$.

Si nous considérons le schéma donné en exemple ci-dessus, nous avons :

$$\begin{aligned} LI &= \{1\} \cup \{0^n 11^n / n \geq 1\} \cup \{0^n 11^i 0 / n \geq 1 \text{ et } i \leq n\} \\ FR(LI) &= \{0^n 1 / n \geq 0\} \end{aligned}$$

Théorème : Deux S.R.M. sont I -équivalents si et seulement si leurs langages d'interprétation ont même forme réduite.

Preuve :

$$(1) \quad FR(LI(R)) = FR(LI(S)) \Rightarrow R \equiv_I S$$

ceci est la partie facile. Si R et S ne sont pas I -équivalents, cela signifie qu'il existe une interprétation libre I pour laquelle le calcul de l'un des schémas se termine et pas celui de l'autre. Supposons que celui de R se termine et que celui de S boucle. L'interprétation I étant représentée par un mot infini ω_I sur l'alphabet $\{0, 1\}^k$, $LI(R)$ contient un facteur gauche fini de ω_I alors que $LI(S)$ n'en contient aucun. Par conséquent, $FR(LI(R))$ contient un facteur gauche fini de ω_I alors que $FR(LI(S))$ n'en contient aucun.

$$(2) \quad R \equiv_I S \Rightarrow FR(LI(R)) = FR(LI(S))$$

Supposons $FR(LI(R)) \neq FR(LI(S))$ et soit w un mot minimal appartenant à l'un des langages sans appartenir à l'autre. Supposons que $w \in FR(LI(R))$ et $w \notin FR(LI(S))$. Le schéma R converge pour une interprétation I ayant

w comme facteur gauche (et w est alors un facteur gauche de $w_{R,I}$ le mot d'interprétation correspondant à R et à I). On sait qu'aucun facteur gauche de w n'appartient à $FR(LI(R))$ puisque cet ensemble est préfixe. Donc, en vertu de la minimalité de la longueur de w , aucun facteur gauche de w n'appartient à $FR(LI(S))$. Donc il n'existe pas de code préfixe fini complet C tel que $wC \subset LI(S)$ et par conséquent, si l'on considère l'arborescence représentant $w(\{0, 1\}^k)^*$, on peut en extraire un chemin infini dont aucun facteur gauche n'appartient à $FR(LI(S))$. Soit alors l'interprétation libre I définie par la concaténation de w et de la suite infinie des valuations des sommets de ce chemin. Le schéma S diverge pour cette interprétation alors que le schéma R converge. Les deux schémas R et S ne sont donc pas I -équivalents.

Il ne nous reste plus qu'à montrer que la forme réduite d'un langage quasi-simple est quasi-simple pour ramener l'équivalence d'interprétation des S.R.M. à l'équivalence des automates quasi-simples. Pour cela, nous allons montrer qu'on peut obtenir un automate quasi-simple reconnaissant $FR(L)$ par des réductions successives opérées sur l'automate reconnaissant L . Il nous sera commode de distinguer dans l'alphabet de pile des automates les lettres de fond de pile des autres c'est-à-dire que nous considérerons des automates dont l'alphabet de pile est l'union disjointe de deux alphabets Y et Z , le langage de pile de l'automate étant inclu dans Y^*Z . (Nous notons le fond de pile à droite.) Tout automate peut être mis sous cette forme : il suffit de prendre pour Y l'ancien alphabet de pile et pour Z une copie de Y

$$(Z \subset \{y'/y \in Y\});$$

y'_0 est le nouveau symbole initial de pile et δ est définie sur Z par :

$$\begin{aligned} \delta(x, y') &= (1, \omega t') \quad \text{si } \delta(x, y) = (1, \omega t) \\ &= \delta(x, y) \quad \text{sinon} \end{aligned}$$

Nous définissons la transformation ρ qui à un automate

$$A^{(1)} = \langle X, Y \cup Z^{(1)} \cup T^{(1)}, \delta^{(1)}, z_0 \rangle$$

et à un ensemble $V^{(1)} \subset Z^{(1)} \cup T^{(1)}$ fait correspondre un automate

$$A^{(2)} = \langle X, Y \cup Z^{(2)} \cup T^{(2)}, \delta^{(2)}, z_0 \rangle$$

et un ensemble $V^{(2)} \subset Z^{(2)} \cup T^{(2)}$ où $T^{(1)}$ et $T^{(2)}$ sont des copies partielles de Y et $Z^{(1)} \cup T^{(1)}$ et $Z^{(2)} \cup T^{(2)}$ constituent respectivement les symboles de fond de pile de $A^{(1)}$ et $A^{(2)}$, z_0 le symbole initial de pile appartenant à $Z^{(1)}$ et $Z^{(2)}$ (si $Z^{(2)} \neq \emptyset$) :

— s'il n'existe pas de symbole $\gamma \neq z \in Z^{(1)} \cup T^{(1)}$ tel que

$$\forall x \in X, \delta^{(1)}(x, \gamma) = (1, e)$$

alors $A^{(2)} = A^{(1)}$ et $V^{(2)} = V^{(1)}$

— s'il existe un tel symbole γ alors pour tout couple

$$(x, \gamma_1) x \in X, \gamma_1 \in Z^{(1)} \cup T^{(1)},$$

on définit $\delta^{(2)}(x, \gamma_1)$ de la façon suivante :

$$\begin{aligned} \text{si } \delta^{(1)}(x, \gamma_1) & [(1, y_{j_n} \dots y_{j_i} \gamma) \\ \delta^{(2)}(x, \gamma_1) & = (1, y_{j_n} \dots y'_{j_k}) \end{aligned}$$

où k est le plus petit entier i tel que $y'_{j_i} \notin T^{(1)}$

$V^{(2)} = V^{(1)} \cup \{ \gamma \}$. $T^{(2)}$ est obtenu à partir de $T^{(1)}$ en ajoutant tous les y'_{j_k} introduits dans ce qui précède et en retranchant γ si $\gamma \in T^{(1)}$.

Si $\gamma \in Z^{(1)}$ alors $Z^{(2)} = Z^{(1)} - \{ \gamma \}$.

La fonction de transition $\delta^{(2)}$ est définie pour ces y'_{j_k} par :

$$\begin{aligned} \forall x \in X \quad \delta^{(2)}(x, y'_{j_k}) &= (1, \omega t') \quad \text{si } \delta^{(1)}(x, y_{j_k}) = (1, \omega t) \\ \delta^{(2)}(x, y'_{j_k}) &= (1, e) \quad \text{sinon} \end{aligned}$$

Partant d'un automate $A^{(0)} = \langle X, Y \cup Z^{(0)}, \delta^{(0)}, z_0 \rangle$ ($T^{(0)} = \emptyset$) et de l'ensemble $V^{(0)} = \emptyset$, nous pouvons appliquer de façon répétitive la réduction ρ au couple $(A^{(0)}, V^{(0)})$. Nous noterons $(A^{(n)}, V^{(n)}) = \rho^n(A^{(0)}, V^{(0)})$. Nous allons montrer que le processus converge, c'est-à-dire qu'il existe un entier n tel que $\forall p (A^{(n+p)}, V^{(n+p)}) = (A^{(n)}, V^{(n)})$, et qu'alors

$$L(A^{(n)}) = FR(L(A^{(0)})).$$

Lemme : $\exists n_0 \forall p (A^{(n_0+p)}, V^{(n_0+p)}) = (A^{(n_0)}, V^{(n_0)})$.

Preuve : A toute étape $(A^{(n)}, V^{(n)}) \xrightarrow{\rho} (A^{(n+1)}, V^{(n+1)})$, si
 $(A^{(n+1)}, V^{(n+1)}) \neq (A^{(n)}, V^{(n)})$

alors $\text{Card}(V^{(n+1)}) = \text{Card}(V^{(n)}) + 1$. D'autre part

$$\forall n \text{ Card}(Z^{(n)} \cup T^{(n)}) = \text{Card}(Y \cup Z^{(0)}) - \text{Card}(V^{(n)})$$

Donc, en désignant par k le nombre d'éléments de $Y \cup Z^{(0)}$, si

$$(A^{(k)}, V^{(k)}) \neq (A^{(k-1)}, V^{(k-1)}) \quad \text{alors} \quad \text{Card}(V^{(k)}) = k$$

et donc $T^{(k)} = \emptyset$ et $Z^{(k)} = \emptyset$ ce qui implique : $(A^{(k+1)}, V^{(k+1)}) = (A^{(k)}, V^{(k)})$
 et donc $\forall p (A^{(k+p)}, V^{(k+p)}) = (A^{(k)}, V^{(k)})$.

Le nombre d'étapes de notre réduction est borné par le nombre de symboles de pile de l'automate de départ.

Lemme : Si $(A^{(n+1)}, V^{(n+1)}) = (A^{(n)}, V^{(n)})$ alors $FR(L(A^{(n)})) = L(A^{(n)})$.

Preuve : Supposons que $(A^{(n+1)}, V^{(n+1)}) = (A^{(n)}, V^{(n)})$ et que le langage $L(A^{(n)})$ n'est pas sous forme réduite. Alors $\exists w \in X^*$ tel que $\forall x \in X wx \in L(A^{(n)})$.

Notons $\pi_2(\delta^{(n)}(w, z_0)) = y_{j_m} \dots y_{j_1} \gamma$. On a

$$\forall i \ 1 \leq i \leq m \ \forall x \in X; \delta^{(n)}(x, y_{j_i}) = (0, e) \quad (1)$$

et $\forall x \in X \delta^{(n)}(x, \gamma) = (1, e)$

Mais alors γ est un symbole susceptible d'être éliminé et par conséquent

$$(A^{(n+1)}, V^{(n+1)}) \neq (A^{(n)}, V^{(n)}).$$

Lemme : $\forall n \ FR(L(A^{(n+1)})) = FR(L(A^{(n)}))$.

Preuve : Nous commencerons par montrer que tout mot de $L(A^{(n)})$ a un facteur gauche unique dans $L(A^{(n+1)})$. Notons tout de suite que l'unicité est assurée par la préfixité de nos langages. Soit wa un mot de $L(A^{(n)})$ dont nous isolons la dernière lettre a et posons $\pi_2(\hat{\delta}^{(n)}(w, z_0)) = y_{j_m} \dots y_{j_1} \gamma$. Considérons la suite C_1, \dots, C_p des mots de pile utilisés dans la reconnaissance de w et $\gamma_1, \dots, \gamma_p$ les symboles de fond de pile correspondants. On a : $\gamma_{p-1} = \gamma$ et $\gamma_p = e$. Il est à noter que parmi tous les symboles de fond de pile γ_i utilisés au cours de la dérivation, seul γ a pu être éventuellement supprimé dans la transformation ρ puisque, pour être supprimé un symbole γ_i doit vérifier :

$$\forall x \in X \delta^{(n)}(x, \gamma_i) = (1, e).$$

Si γ n'a pas été supprimé lors du passage de $A^{(n)}$ à $A^{(n+1)}$, alors toutes les règles utilisées lors de la reconnaissance du mot wa par $A^{(n)}$ existent encore dans $A^{(n+1)}$ et $wa \in L(A^{(n+1)})$.

Supposons maintenant que γ a été supprimé lors du passage de $A^{(n)}$ à $A^{(n+1)}$. Ceci implique que $\forall x \in X \delta^{(n)}(x, \gamma) = (1, e)$. Nous désignerons par q le plus grand entier tel que $\gamma_q \neq \gamma$. Le mot wa peut se décomposer en

$$wa = ubva \ (u, v \in X^* \text{ et } b, a \in X)$$

de telle façon que :

$$\hat{\delta}^{(n)}(u, z_0) = (e, \omega \gamma_q) \text{ où } \omega \text{ est formé de symboles } y \in Y \text{ vérifiant}$$

$$\delta^{(n)}(b, y) = (0, e).$$

$$\hat{\delta}^{(n)}(ub, Z_0) = (e, y_{i_r} \dots y_{i_1} \gamma) \text{ ce qui implique } \delta^{(n)}(b, \gamma_q) = (1, y_{i_r} \dots y_{i_1} \gamma).$$

D'après la définition de ρ , ou bien $\delta^{(n+1)}(b, \gamma_q) = (1, e)$ et $ub \in L(A^{(n+1)})$ ou bien $\exists s \in [r]$ tel que $\delta^{(n+1)}(b, \gamma_n) = (1, y_{i_r}, \dots, y_{i_s})$. D'autre part comme le mot $ubva$ est reconnu par l'automate $A^{(n)}$, il existe une décomposition de v en $v = v_1 c v_2 d v_3$ ($v_1, v_2, v_3 \in X^*$ et $c, d \in X$) avec éventuellement $v_2 = e$ et c et d confondus telle que :

$$\begin{aligned} - \hat{\delta}^{(n)}(v_1 c, y_{i_r} \dots y_{i_{s+1}}) &= \text{ou bien } (e, e) \\ &\text{ou bien } (c, e) \end{aligned}$$

(1) Il est à noter que l'on peut éliminer dans les automates de tels symboles de pile. On pourrait alors poser : $\pi_2(\hat{\delta}^{(n)}(w, z_0)) = \gamma$ avec $x \in X \delta^{(n)}(x, \gamma) = (1, e)$.

— v_2d est le mot le plus court tel que :

$\hat{\delta}^{(n)}(v_2d, y_{i_s})$ ou bien $\hat{\delta}^{(n)}(cv_2d, y_{i_s}) =$ ou bien $(e, \bar{\omega})$ où $\bar{\omega}$ est soit vide soit
ne contenant que des y
de Y tels que $y' \in V^{(n)}$
ou bien (d, e)

Dans chacun de ces cas, $\hat{\delta}^{(n+1)}(v_1cv_2d, y_{i_r} \dots y_{i_{s+1}} y'_{i_s}) = (e, e)$.

En effet, la condition de minimalité posée sur le facteur v_2d impose qu'à l'étape précédant celle où la pile est un mot $\bar{\omega}$ soit vide soit ne contenant que des y de Y tels que $y' \in V^{(n)}$, la pile est un mot $\bar{y}\bar{\omega}_2$ où $y' \notin V^{(n)}$ et tel que $\delta^{(n)}(d, \bar{y}) =$ soit $(1, \bar{\omega}_1)$ avec $\bar{\omega}_1\bar{\omega}_2 = \bar{\omega}$, soit $(0, e)$ avec $\bar{\omega}_2 = \bar{\omega}$.

L'étape correspondante dans la lecture du mot $m=wa$ par l'automate $A^{(n+1)}$ est alors une étape dont la pile est \bar{y}' et on a $\delta^{(n+1)}(d, \bar{y}') = (1, e)$.

Par conséquent $\hat{\delta}^{(n+1)}(ubv_1cv_2d, z_0) = (e, e)$ et le mot ubv_1cv_2d facteur gauche de $m=wa=ubv_1cv_2dv_3a$ appartient à $L(A^{(n+1)})$.

Dans la suite, nous désignerons par τ la fonction qui fait correspondre à tout mot m de $L(A^{(n)})$ son facteur gauche unique dans $L(A^{(n+1)})$. D'autre part, nous désignerons par $r^{(n)}$ [resp $r^{(n+1)}$] la fonction qui associe à tout mot de $L(A^{(n)})$ [resp $L(A^{(n+1)})$] son facteur gauche unique dans

$$FR(L(A^{(n)})) \text{ [resp } FR(L(A^{(n+1)})) \text{]}.$$

Notre but est maintenant de montrer que

$$\forall m \in L(A^{(n)}) : r^{(n)}(m) = r^{(n+1)}(\tau(m))$$

ce qui implique immédiatement : $FR(L(A^{(n)})) = FR(L(A^{(n+1)}))$.

Revenons à notre mot quelconque $wa \in L(A^{(n)})$. Pour prouver que wa avait un facteur gauche dans $L(A^{(n+1)})$, nous avons distingué deux cas :

1° $wa = ubva$ avec $\hat{\delta}^{(n)}(u, z_0) = (e, \gamma_q)$ où γ_q est formé de symboles $y \in Y$ vérifiant $\delta^{(n)}(b, y) = (0, e)$.

$$\delta^{(n)}(b, \gamma_q) = (1, y_{i_r} \dots y_{i_1} \gamma)$$

$$\text{et donc } \hat{\delta}^{(n)}(ub, z_0) = (e, y_{i_r} \dots y_{i_1} \gamma)$$

$$\delta^{(n+1)}(b, \gamma_q) = (1, e)$$

$$\text{et donc } \hat{\delta}^{(n+1)}(ub, z_0) = (e, e)$$

Du fait que la règle $\delta^{(n)}(b, \gamma_q) = (1, y_{i_r} \dots y_{i_1} \gamma)$ de $A^{(n)}$ est devenue

$$\delta^{(n+1)}(b, \gamma_q) = (1, e) \text{ dans } A^{(n+1)},$$

on peut affirmer que les symboles $y'_{i_s}, \dots, y'_{i_1}$ étaient dans $V^{(n)}$ et donc que les symboles y_{i_r}, \dots, y_{i_1} vérifiaient :

$$\forall j \in [r] \forall x \in X \delta^{(n)}(x, y_{i_j}) = \text{ou bien } (1, e) \\ \text{ou bien } (0, e)$$

et que γ était tel que : $\forall x \delta^{(n)}(x, \gamma) = (1, e)$.

Or il est facile de vérifier qu'alors, l'ensemble

$$\{ m \in X / \hat{\delta}^{(n)}(m, y_{i_r} \dots y_{i_1} \gamma) = (e, e) \}$$

est un code préfixe fini complet. Donc il existe un code préfixe fini complet C tel que ubC soit contenu dans $L(A^{(n)})$.

Considérons maintenant les quatre mots $ubva$, ub , $r^{(n)}(ubva)$ et $r^{(n+1)}(ub)$ qui sont ordonnés dans l'ordre préfixiel puisque tous sont facteur gauche de $ubva$.

Si nous représentons cet ordre par \leq et l'ordre strict par $<$, on a :

$$ub \leq ubva, r^{(n)}(ubva) \leq ubva, r^{(n+1)}(ub) \leq ub \leq ubva ;$$

d'autre part $r^{(n)}(ubva) < ub$ sinon il existerait un facteur gauche strict ub de $r^{(n)}(ubva)$ et un code préfixe fini complet C tels que ubC soit contenu dans $L(A^{(n)})$.

Nous allons maintenant montrer que $r^{(n)}(ubva) = r^{(n+1)}(ub)$ en montrant qu'on ne peut avoir ni $r^{(n)}(ubva) < r^{(n+1)}(ub)$ ni $r^{(n+1)}(ub) < r^{(n)}(ubva)$. Pour cela, nous utiliserons la remarque suivante : si u_1 est un facteur gauche de ub , il existe un code préfixe fini complet C_1 tel que $u_1 C_1$ soit contenu dans $L(A^{(n)})$ si et seulement si il existe un code préfixe fini complet C_2 tel que $u_1 C_2$ soit contenu dans $L(A^{(n+1)})$. Grâce à cette remarque, nous pouvons déduire du fait qu'il existe des codes préfixes finis complets C_1 et C_2 tels que

$$r^{(n)}(ubva)C_1 \text{ et } r^{(n+1)}(ub)C_2$$

soient respectivement contenus dans $L(A^{(n)})$ et $L(A^{(n+1)})$ qu'il existe des codes préfixes finis complets C_3 et C_4 tels que $r^{(n)}(ubva)C_3$ et $r^{(n+1)}(ub)C_4$ soient respectivement contenus dans $L(A^{(n+1)})$ et $L(A^{(n)})$. Donc chacune des deux hypothèses $r^{(n)}(ubva) < r^{(n+1)}(ub)$ et $r^{(n+1)}(ub) < r^{(n)}(ubva)$ viole la deuxième condition de définition de la forme réduite. Donc $r^{(n+1)}(ub) = r^{(n)}(ubva)$.

2° $wa = ubv_1 cv_2 dv_3 a$ avec $\hat{\delta}^{(n)}(u, z_0) = (e, \omega \gamma_q)$ où ω est formé de $y \in Y$

$$\hat{\delta}^{(n)}(b, \gamma_q) = (1, y_{i_r} \dots y_{i_1} \gamma)$$

et donc $\hat{\delta}^{(n)}(ub, z_0) = (e, y_{i_r} \dots y_{i_1} \gamma)$

$$\delta^{(n+1)}(b, \gamma_q) = (1, y_{i_r} \dots y'_{i_s})$$

et donc $\hat{\delta}^{(n+1)}(ub, z_0) = (e, y_{i_r} \dots y'_{i_s})$

$$\hat{\delta}^{(n)}(v_1 c, y_{i_r} \dots y_{i_{s+1}}) = \text{ou bien } (e, e) \\ \text{ou bien } (c, e)$$

$$\hat{\delta}^{(n)}(v_1 cv_2 d, y_{i_r} \dots y_{i_{s+1}} y_{i_s}) = \text{soit } (e, \bar{\omega})$$

où $\bar{\omega}$ est soit vide soit ne contenant que des $y \in T$ t.q. $y \notin V^{(n)}$ soit (d, e)

Nous avons vu qu'alors $ubv_1cv_2d = \tau(wa)$. On a donc

$$\hat{\delta}^{(n)}(ubv_1cv_2d, z_0) = (e, \bar{w}') \text{ où } \bar{w}' \text{ est un facteur droit de } \bar{w}y_{i_s-1} \dots y_{i_1}.$$

Le symbole γ vérifie $\forall x \in X \delta^{(n)}(x, \gamma) = (1, e)$ et tous les symboles y du mot $\bar{w}y_{i_s-1} \dots y_{i_1}$ vérifient $\forall x \in X \delta^{(n)}(x, y) = (1, e)$ ou $(0, e)$. Donc, l'ensemble $\{m \in X^* / \hat{\delta}^{(n)}(m, \bar{w}') = (e, e)\}$ est un code préfixe fini complet C ; par conséquent ubv_1cv_2dC est contenu dans $L(A^{(n)})$ et nous pouvons montrer que

$$r^{(n)}(ubv_1cv_2dv_3a) = r^{(n+1)}(ubv_1cv_2d)$$

par le même argument que dans le cas précédent. Ceci achève la démonstration du lemme.

Des trois lemmes précédents, nous déduisons que si k est le nombre de symboles de pile de l'automate $A^{(0)}$ initial :

$$L(A^{(k)}) = FR(L(A^{(k)})) = FR(L(A^{(0)})).$$

Nous pouvons donc énoncer :

Théorème : La forme réduite d'un langage quasi-simple est quasi-simple.

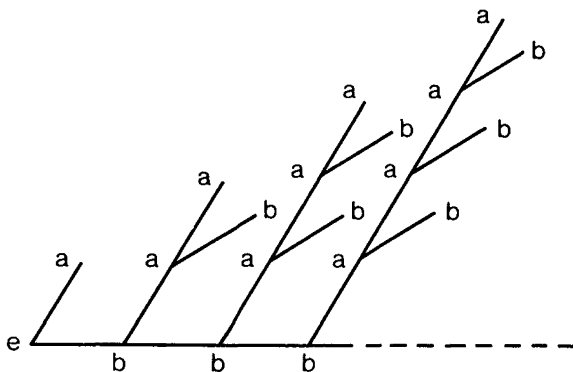
De plus, nous disposons d'un procédé effectif pour la construire.

Exemple de construction de forme réduite :

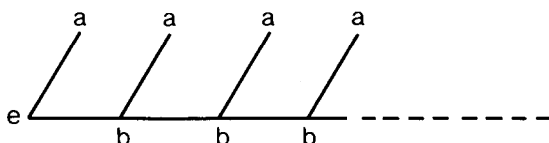
Soit l'automate $A^{(0)} = \langle \{a, b\}, \{y_1, y_2\} \{z_0, z_1\}, \delta^{(0)}, z_0 \rangle$ défini par :

$$\begin{aligned} \delta^{(0)}(a, z_0) &= (1, e) & \delta^{(0)}(b, z_0) &= (1, y_1z_1) & \delta^{(0)}(a, z_1) &= (1, e) \\ & & & & \delta^{(0)}(b, z_1) &= (1, e) \\ \delta^{(0)}(a, y_1) &= (1, e) & \delta^{(0)}(b, y_1) &= (1, y_1y_2) & \delta^{(0)}(a, y_2) &= (1, e) \\ & & & & \delta^{(0)}(b, y_2) &= (0, e). \end{aligned}$$

On a : $L(A^{(0)}) = \{b^n a^m b / 1 \leq m \leq n\} \cup \{b^n a^{n+1} / n \geq 0\}$ langage qu'il est agréable de représenter par l'arbre ci-dessous :



$FR(LA^{(0)}) = \{ b^n a / n \geq 0 \}$ qu'on peut représenter par



Nous remarquons sur cet exemple que la forme réduite d'un langage non rationnel peut être rationnelle. Si nous appliquons la réduction ρ au couple initial $(A^{(0)}, \emptyset)$, nous obtenons $A^{(1)}, V^{(1)}$ avec $V^{(1)} = \{ z_1 \}$ et

$$A^{(1)} = \langle \{ a, b \}, \{ y_1, y_2 \} \{ z_0 \} \{ y'_1, y'_2 \}, \delta^{(1)}, z_0 \rangle$$

où $\delta^{(1)}$ est définie par :

$$\begin{aligned} \delta^{(1)}(a, z_0) &= (1, e) & \delta^{(1)}(b, z_0) &= (1, y'_1) & \delta^{(1)}(a, y_1) &= (1, e) & \delta^{(1)}(b, y_1) &= (1, y_1 y_2) \\ \delta^{(1)}(a, y_2) &= (1, e) & \delta^{(1)}(b, y_2) &= (0, e) & \delta^{(1)}(a, y'_1) &= (1, e) & \delta^{(1)}(b, y'_1) &= (1, y_1 y'_2) \\ \delta^{(1)}(a, y'_2) &= (1, e) & \delta^{(1)}(b, y'_2) &= (1, e) & & & & \end{aligned}$$

Si nous appliquons une nouvelle fois la réduction ρ au couple $(A^{(1)}, V^{(1)})$, nous obtenons $(A^{(2)}, V^{(2)})$ avec $V^{(2)} = \{ z_1, y'_2 \}$ et

$$A^{(2)} = \langle \{ a, b \}, \{ z_0, y'_2 \}, \delta^{(2)}, z_0 \rangle$$

où $\delta^{(2)}$ est définie par :

$$\delta^{(2)}(a, z_0) = (1, e) \quad \delta^{(2)}(b, z_0) = (1, y'_1) \quad \delta^{(2)}(a, y'_1) = (1, e)$$

et $\delta^{(2)}(b, y'_1) = (1, y'_1)$.

$$\text{On a : } L(A^{(2)}) = \{ b^n a / n \geq 0 \} = FR(L(A^{(0)})).$$

Nous allons maintenant montrer deux propriétés des S.R.M. dont les démonstrations font appel aux langages d'interprétation et qui, parce qu'elles illustrent la différence entre les S.R.M. et leur généralisation que nous allons introduire par la suite, feront la transition avec la deuxième partie de cet article.

Définition : Étant donnés deux S.R.M. R et S , on appellera *conjonction* de R et S tout S.R.M. T dont le calcul pour une interprétation I se termine ssi les calculs de R et S se terminent.

Définition : Étant donnés deux S.R.M. R et S , on appellera *disjonction* de R et S tout S.R.M. T dont le calcul pour une interprétation I se termine ssi au moins des calculs de R et S se termine.

Lemme : Étant donnés trois S.R.M. R , S et T définis sur les mêmes alphabets de base, T est une conjonction de R et S si et seulement si :

$$FR(LI(T)) = [FR(LI(R))W^* \cap FR(LI(S))] \cup [FR(LI(R)) \cap FR(LI(S))W^*]$$

où W est l'alphabet $\{0, 1\}^k$ sur lequel sont définis les langages d'interprétation.

Théorème : Étant donnés deux S.R.M. R et S , il n'existe pas en général de conjonction de R et S dans la classe des S.R.M.

Preuve : Considérons le schéma S_1 défini par les équations :

$$\begin{aligned} xF_1 &:= \text{si } xP \text{ alors } xbF_2F_3 \text{ sinon } xF_1 \\ xF_2 &:= \text{si } xP \text{ alors } xbF_2F_3 \text{ sinon } x \\ xF_3 &:= \text{si } xP \text{ alors } xF_3 \text{ sinon } xb \end{aligned}$$

$$\text{On a } LI(S_1) = \{1^n 0^n / n \in \mathbb{N} - \{0\}\} = FR(LI(S_1)).$$

Considérons d'autre part le schéma S_2 défini par :

$$\begin{aligned} xG_1 &:= \text{si } xP \text{ alors } xbG_1 \text{ sinon } xbG_2G_3G_4 \\ xG_2 &:= \text{si } xP \text{ alors } x \text{ sinon } xbG_2G_3 \\ xG_3 &:= \text{si } xP \text{ alors } xb \text{ sinon } xG_3 \\ xG_4 &:= \text{si } xP \text{ alors } xG_4 \text{ sinon } x \end{aligned}$$

$$\text{On a } LI(S_2) = \{1^m 0^n 1^n / m \in \mathbb{N} \text{ et } n \in \mathbb{N} - \{0\}\} = FR(LI(S_2)).$$

Donc, en appliquant le lemme, si un schéma S_3 est une conjonction de S_1 et S_2 , on a $FR(LI(S_3)) = \{1^n 0^n 1^n / n \in \mathbb{N} - \{0\}\}$ qui n'est pas un langage algébrique et en particulier n'est pas quasi-simple. Il en résulte que $LI(S_3)$ n'est lui-même pas quasi-simple et que S_3 ne peut donc être un S.R.M.

Lemme : Étant donné trois S.R.M. R , S et T définis sur les mêmes alphabets de base, T est une disjonction de R et S ssi :

$$FR(LI(T)) = [FR(LI(R)) \cup FR(LI(S))] [FR(LI(R)) \cup FR(LI(S))] WW^*$$

Théorème : Étant donnés deux S.R.M. R et S , il n'existe pas en général de disjonction de R et S dans la classe des S.R.M.

Considérons le schéma S_1 utilisé dans la preuve du théorème précédent et le schéma S_3 défini par les équations :

$$\begin{aligned} xH_1 &:= \text{si } xP \text{ alors } xbH_2H_5 \text{ sinon } xH_1 \\ xH_2 &:= \text{si } xP \text{ alors } xbH_3H_4 \text{ sinon } xH_2 \\ xH_3 &:= \text{si } xP \text{ alors } xbH_3 \text{ sinon } x \\ xH_4 &:= \text{si } xP \text{ alors } xH_4 \text{ sinon } xb \\ xH_5 &:= \text{si } xP \text{ alors } xH_5 \text{ sinon } xbH_6H_7H_8 \\ xH_6 &:= \text{si } xP \text{ alors } x \text{ sinon } xbH_6H_7 \\ xH_7 &:= \text{si } xP \text{ alors } xb \text{ sinon } xH_7 \\ xH_8 &:= \text{si } xP \text{ alors } xH_8 \text{ sinon } xb \end{aligned}$$

On a $LI(S_3) = \{ 1^{2n}0^{n+m}1^m0/m, n \in \mathbb{N} - \{ 0 \} \} = FR(LI(S_3))$ et comme nous l'avons vu plus haut : $LI(S_1) = \{ 1^n0^n/n \in \mathbb{N} - \{ 0 \} \} = FR(LI(S_1))$. Donc, en appliquant le lemme, si un schéma S_4 est une disjonction de S_1 et S_3 , on a :

$FR(LI(S_4)) [\{ 1^n0^n/n \in \mathbb{N} - \{ 0 \} \} \cup \{ 1^{2n}0^{n+m}1^m0/0 < m < n \}]$. On peut montrer que ce langage n'est pas algébrique et en particulier n'est pas quasi-simple. Il en résulte que $LI(S_4)$ n'est lui-même pas quasi-simple et que par conséquent S_4 ne peut pas être un S.R.M.

DEUXIEME PARTIE : LES SCHEMAS RECURSIFS GENERALISES

Les schémas que nous allons considérer maintenant constituent une généralisation naturelle des schémas récursifs monadiques étudiés précédemment. Dans de tels schémas les tests peuvent porter non plus uniquement sur la valeur actuelle de la variable, mais aussi sur la valeur calculée de fonctions composées d'éléments de \mathcal{B} et \mathcal{F} .

Définition : Étant donnés les trois alphabets de base \mathcal{B} , \mathcal{F} et \mathcal{F} un schéma récursif monadique généralisé [2] S sur les alphabets \mathcal{B} , \mathcal{F} et \mathcal{F} est un système de 1 schémas de définition de la forme : $x_{F_i} := Si \ x\gamma_i P_{r_i}$, alors $x\alpha_i$ sinon $x\beta_i$ où

$$\forall i \in [l] \ P_{r_i} \in \mathcal{F}\alpha_i, \beta_i, \gamma_i \in (\mathcal{B} \cup \mathcal{F})^*$$

Les notions d'interprétation libre ou non libre sont définies comme dans la première partie.

Le calcul d'un schéma récursif monadique généralisé (SRMG) pour une interprétation I est une suite C de suites finies de couples

$$C_j = ((x_{j_1}, y_{j_1}), \dots, (x_{j_{n_j}}, y_{j_{n_j}}))$$

où $\forall i \in [n_j] \ x_{j_i} \in \mathcal{B}^*$ et $y_{j_i} \in (\mathcal{B} \cup \mathcal{F})^*$ définie récursivement comme suit :

(*) $C_0 = ((e, F_1))$

et si $C_j = ((x_{j_1}, y_{j_1}), \dots, (x_{j_{n_j}}, y_{j_{n_j}}))$

(*) si $y_{j_{n_j}} = F_i y'$ où $y' \in (\mathcal{B} \cup \mathcal{F})^*$ et $F_i \in \mathcal{F}$ (1)

$$\gamma_i = e \quad (2) \quad \text{et} \quad x_{j_{n_j}}(P_{r_i})_I = 1$$

alors $C_{j+1} = ((x_{j_1}, y_{j_1}), \dots, (x_{j_{n_j-1}}, y_{j_{n_j-1}}), (x_{j_{n_j}}, \alpha_i y'))$

(*) si les conditions (1) et (2) sont vérifiées et si de plus $x_{j_{n_j}}(P_{r_i})_I = 0$

alors $C_{j+1} = ((x_{j_1}, y_{j_1}), \dots, (x_{j_{n_j-1}}, y_{j_{n_j-1}}), (x_{j_{n_j}}, \beta_i y'))$.

(*) si $y_{j_{n_j}} = b_i y'$ avec $y' \in (\mathcal{B} \cup \mathcal{F})^*$

alors $C_{j+1} = ((x_{j_1}, y_{j_1}), \dots, (x_{j_{n_j-1}}, y_{j_{n_j-1}}), (x_{j_{n_j}}, b_i y'))$

(*) si $y_{j_{n_j}} = e$ et $n_j > 1$ (3), $y_{j_{n_j-1}} = F_i y'$ (4) et $x_{j_{n_j}}(P_{r_i})_I = 1$

alors $C_{j+1} = ((x_{j_1}, y_{j_1}), \dots, (x_{j_{n_j-1}}, \alpha_i y'))$

- (*) si les conditions (3) et (4) sont vérifiées et si de plus $x_{j_n_j}(P_{r_i}) = 0$
 alors $C_{j+1} = ((x_{j_i}, y_{j_i}), \dots, (x_{j_{n_j-1}}, \beta_j y'))$
 (*) si $y_{j_n_j} = e$ et $n_j = 1$ alors $C_{j+1} = C_j$.

On dit que le calcul C se termine s'il existe un entier j tel que $y_{j_n_j} = e$ et $n_j = 1$. On note alors $\text{Val}(S, I) = x_{j_i}$.

Nous avons vu qu'à toute interprétation libre I d'un schéma récursif monadique on pouvait faire correspondre une arborescence h -aire infinie ($h = \text{Card}(\mathcal{B})$) dont les sommets sont étiquetés par des éléments de $\{0, 1\}^k$. D'autre part nous avons remarqué qu'au calcul d'un schéma récursif monadique ordinaire correspond un chemin et un seul dans cette arborescence (l'ensemble des mots testés est totalement ordonné par l'ordre préfixiel); de plus, lorsque le calcul se termine, ce chemin est fini, ce qui nous permettait de définir un mot d'interprétation. Nous allons voir que les schémas récursifs généralisés n'ont pas cette bonne propriété.

Soit le schéma S défini par le système

$$\begin{aligned} xF_1 &:= \text{Si } xP \text{ alors } x \text{ sinon } xG_1 \\ xG_1 &:= \text{Si } xaF_1P \text{ alors } xH_1 \text{ sinon } xH_1 \\ xH_1 &:= \text{Si } xbF_1P \text{ alors } x \text{ sinon } x \end{aligned}$$

et soit d'autre part une interprétation libre I vérifiant :

$$eP_I = aP_I = 0 \quad \text{et} \quad a^2P_I = abP_I = bP_I = 1.$$

Le calcul de S pour l'interprétation I est :

$$\begin{aligned} C_0 &= ((e, F_1)) \rightarrow C_1 = ((e, G_1)) \rightarrow C_2 = ((e, G_1), (e, aF_1)) \rightarrow C_3 = ((e, G_1), (a, F_1)) \rightarrow \\ C_4 &= ((e, G_1), (a, G_1)) \rightarrow C_5 = ((e, G_1), (a, G_1), (a, aF_1)) \rightarrow C_6 = ((e, G_1), (a, G_1), (a^2, F_1)) \rightarrow \\ C_7 &= ((e, G_1), (a, G_1), (a^2, e)) \rightarrow C_8 = ((e, G_1), (a, H_1)) \rightarrow C_9 = ((e, G_1), (a, H_1), (a, bF_1)) \rightarrow \\ C_{10} &= ((e, G_1), (a, H_1), (ab, F_1)) \rightarrow C_{11} = ((e, G_1), (a, H_1), (ab, e)) \rightarrow C_{12} = ((e, G_1), (a, e)) \rightarrow \\ C_{13} &= ((e, H_1)) \rightarrow C_{14} = ((e, H_1), (e, bF_1)) \rightarrow C_{15} = ((e, H_1), (b, F_1)) \rightarrow C_{16} = ((e, H_1), (b, e)) \rightarrow \\ C_{17} &= ((e, e)). \end{aligned}$$

Les valeurs testées au cours du calcul sont donc situées sur trois chemins différents : (e, a, a^2) , (e, a, ab) et (e, b) .

A la partie d'une interprétation explorée au cours d'un calcul d'un SRMG correspond lorsque ce calcul se termine, un arbre fini valué. Nous sommes donc amenés à définir pour ces schémas la notion de forêt d'interprétation correspondant à celle de langage d'interprétation pour les schémas récursifs monadiques ordinaires.

Nous désignerons par A_I l'arborescence h -aire valuée associée à l'interprétation libre I .

Définitions : (1) L'arbre d'interprétation $A_{S,I}$ associé à S et I est un sous-graphe de A_I défini de la façon suivante :

(*) si le calcul de $\langle S, I \rangle$ ne se termine pas, $A_{S,I}$ n'est pas défini

(*) si le calcul de $\langle S, I \rangle$ est fini, nous désignerons par $T_{S,I}$ l'ensemble des valeurs testées et les sommets de $A_{S,I}$ seront ceux désignés par les éléments de $FG(T_{S,I})$ (ensemble des facteurs gauches des éléments de $T_{S,I}$).

(2) Nous appellerons *forêt d'interprétation* d'un schéma récursif monadique généralisé S l'ensemble : $FI(S) = \{ A_{S,I} / \text{le calcul de } \langle S, I \rangle \text{ se termine} \}$.

Si l'on désire comparer la puissance des deux classes de schémas récursifs monadiques au moyen de l'outil que représente le langage d'interprétation, on est amené à considérer un codage des arbres valués sur $\{0, 1\}^k$ par des mots sur le monoïde libre engendré par $\{0, 1\}^k \cup \{ ", ", "(", ")" \}$. Notre but n'est pas ici de détailler ce codage (voir [2]), d'autant plus que l'augmentation de puissance apportée par la possibilité d'effectuer des calculs annexes avant les tests peut être mesurée en considérant les schémas récursifs généralisés normaux pour lesquels la notion de langage d'interprétation est bien définie.

Définition : Un schéma récursif généralisé est dit *normal* si pour toute interprétation libre I les valeurs testées au cours du calcul de $\langle S, I \rangle$ sont totalement ordonnées par l'ordre préfixiel.

Pour étudier la classe des langages d'interprétation des schémas normaux (ces langages sont bien définis d'après la définition même d'un schéma normal), il suffit par ailleurs d'étudier celle des langages d'interprétation des schémas à un seul symbole de fonction de base; en effet, tous ces schémas sont évidemment normaux et réciproquement, étant donné un schéma normal S_1 quelconque, le schéma S_2 obtenu en identifiant dans S_1 tous les symboles de fonction de base a même langage d'interprétation que S_1 .

Dans la première partie, nous avons vu qu'étant donnés deux schémas récursifs monadiques (non généralisés) S_1 et S_2 on ne pouvait pas en général en construire un troisième S_3 qui converge pour les seules interprétations qui font converger à la fois S_1 et S_2 . Par contre, on a le

Théorème : Étant donnés deux schémas récursifs monadiques quelconques S_1 et S_2 , on peut construire un schéma récursif monadique généralisé S_3 , conjonction de S_1 et S_2 .

Soit en effet deux schémas S_1 et S_2 (généralisés ou non) définis sur les alphabets \mathcal{B} , \mathcal{F} et sur respectivement $\{G_1, \dots, G_m\}$ et $\{H_1, \dots, H_n\}$. Soit le schéma S_3 défini sur les alphabets \mathcal{B} , \mathcal{F} et $\{F_1, G_1, \dots, G_m, H_1, \dots, H_n\}$ (où F_1 est le symbole initial) par le système de $m + n + 1$ schémas de définition :

$$\left\{ \begin{array}{l} xF_1 := Si xH_1P_1 \text{ alors } xG_1 \text{ sinon } xG_1 \\ \text{et les } m + n \text{ schémas de définition de } S_1 \text{ et } S_2. \end{array} \right.$$

On vérifie aisément que S_3 converge pour une interprétation I si et seulement si les deux schémas S_1 et S_2 convergent pour cette interprétation. Remarquons que l'on a alors $\text{Val}(S_3, I) = \text{Val}(S_1, I)$. Dans le cas particulier où les schémas S_1 et S_2 sont des schémas à un seul symbole de fonction de base tels qu'au cours de leurs calculs pour les interprétations libres le plus long mot testé soit le résultat de ces calculs, on a :

$$\text{LI}(S_3) = (\text{LI}(S_1)W^* \cap \text{LI}(S_2)W^*) \setminus (\text{LI}(S_1)W^* \cap \text{LI}(S_2)W^*)W^+$$

où : $W = \{ 0, 1 \}^{\text{Card}(S)}$ et nous noterons $\text{LI}(S_3) = \text{LI}(S_1) \cap_I \text{LI}(S_2)$.

La fermeture de la classe des langages d'interprétation de ces schémas par l'opération \cap_I explique l'existence de langages context-sensitive dans cette classe. Soient les deux schémas rékursifs monadiques S_1 et S_2 :

$$\begin{array}{l}
 \begin{array}{l}
 \text{S}_1 \\
 \\
 \\
 \\
 \\
 \end{array}
 \begin{array}{l}
 \text{xG}_1 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xG}_2 \text{G}_4 \quad \underline{\text{sinon}} \text{ xG}_1 \\
 \text{xG}_2 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xG}_2 \text{G}_3 \quad \underline{\text{sinon}} \text{ x} \\
 \text{xG}_3 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xG}_3 \quad \underline{\text{sinon}} \text{ xb} \\
 \text{xG}_4 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xG}_5 \quad \underline{\text{sinon}} \text{ xG}_4 \\
 \text{xG}_5 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xG}_5 \quad \underline{\text{sinon}} \text{ x}
 \end{array}
 \\
 \\
 \begin{array}{l}
 \text{et S}_2 \\
 \\
 \\
 \\
 \\
 \end{array}
 \begin{array}{l}
 \text{xH}_1 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xH}_2 \quad \underline{\text{sinon}} \text{ xH}_1 \\
 \text{xH}_2 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xH}_2 \quad \underline{\text{sinon}} \text{ xH}_3 \\
 \text{xH}_3 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xH}_3 \quad \underline{\text{sinon}} \text{ xH}_4 \text{H}_5 \\
 \text{xH}_4 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ x} \quad \underline{\text{sinon}} \text{ xH}_4 \text{H}_5 \\
 \text{xH}_5 := \underline{\text{Si}} \text{ xP } \underline{\text{alors}} \text{ xb} \quad \underline{\text{sinon}} \text{ xH}_5
 \end{array}
 \end{array}$$

On a : $\text{LI}(S_1) = \{ 1^m 0^n 1^m / m, n \in \mathbb{N}^+ \}$
 et $\text{LI}(S_2) = \{ 1^n 0^n 1^n / m, n \in \mathbb{N}^+ \}$.

Si l'on considère alors la conjonction S_3 obtenue par la construction précédente, on obtient un schéma tel que :

$$\text{LI}(S_3) = \text{LI}(S_1) \cap_I \text{LI}(S_2) = \{ 1^n 0^n 1^n / n \in \mathbb{N}^+ \},$$

langage context-sensitive non algébrique. En fait, on montre le

Théorème : Les langages d'interprétation des schémas rékursifs monadiques généralisés normaux sont des langages context-sensitive.

Nous renvoyons le lecteur à [2] pour une démonstration détaillée de ce théorème qui consiste

- (1) à associer à tout schéma généralisé S à un symbole de base b un automate à pile A_S , déterministe avec marqueur, fonctionnant dans les deux sens et tel que $L(A_S) = \text{LI}(S)$;
- (2) à montrer qu'au cours de tout calcul de A_S , la longueur de la pile reste

bornée par une fonction linéaire de la longueur de l'entrée et par conséquent, il existe un automate à mémoire linéairement bornée MLB_S tel que

$$L(MLB_S) = L(A_S) = LI(S).$$

En fait, ce résultat s'étend au cas des schémas rékursifs monadiques généralisés quelconques en confondant l'ensemble des termes du magma libre représentant l'ensemble des arbres h -aires valués avec les mots correspondants du monoïde libre engendré par l'alphabet servant à les écrire et on montre de façon similaire que les forêts d'interprétation sont context-sensitive.

La nature des langages d'interprétation des SRMG a naturellement une influence sur la nature des langages valeur de ces schémas. On sait que :

Proposition [3] : La classe des langages valeur des schémas rékursifs monadiques (non généralisés) est exactement la classe des langages algébriques.

Par contre, on a

Proposition : Il existe un schéma rékursif monadique généralisé dont le langage valeur est context-sensitive non algébrique.

Soit en effet le schéma S défini par les schémas de définition suivants :

$$\begin{aligned} xF_1 &:= \underline{\text{Si}} \ xF_2P \ \underline{\text{alors}} \ xF_4 \ \underline{\text{sinon}} \ xF_1 \\ xF_2 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xaF_2F_3 \ \underline{\text{sinon}} \ x \\ xF_3 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xF_3 \ \underline{\text{sinon}} \ x \\ xF_4 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xaF_4 \ \underline{\text{sinon}} \ xF_5 \\ xF_5 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ x \ \underline{\text{sinon}} \ xbF_5c \end{aligned}$$

a pour langage valeur $LV(S) = \{ a^n b^n c^n / n > 0 \}$, langage context-sensitive non algébrique; par suite ce schéma n'a pas de schéma non généralisé équivalent.

Nous allons maintenant montrer que les problèmes de l'équivalence forte et de l'équivalence d'interprétation, sur lesquels on ne peut rien conclure dans le cas des schémas non généralisés est indécidable si l'on considère des schémas généralisés.

Théorème : L'équivalence forte et l'équivalence d'interprétation sont indécidables dans la classe des schémas rékursifs monadiques généralisés.

Preuve : Le principe de la preuve est de coder un problème de Post en celui de l'équivalence de deux schémas.

Soit sur l'alphabet $\Sigma = \{ a, b \}$ un problème de correspondance de Post (PCP) $\pi = ((\varepsilon_1, \eta_1), \dots, (\varepsilon_s, \eta_s))$ où $\forall i \in [s], \varepsilon_i, \eta_i \in \Sigma^+$. Soit $\Sigma' = \{ a, b, c \}$ et ι et ρ les codages suivants : $\iota : \Sigma' \rightarrow X^2$ où $X = \{ 0, 1 \}$ est défini par $a\iota = 00, b\iota = 01$ et $c\iota = 10$ et se prolonge en un morphisme de $(\Sigma')^*$ dans $(X^2)^*$ et $\rho : [s] \rightarrow (X^2)^*$ est défini par $i\rho = (00)^{i-1}01$ pour tout i dans $[s]$.

On considère par ailleurs les morphismes g et \hat{g} définis comme suit :

$g : \Sigma \rightarrow \{ F_{00}, F_{01} \}$ par $ag = F_{00}$ et $bg = F_{01}$; on étend g en un morphisme de Σ^* sur $\{ F_{00}, F_{01} \}^*$;

$\hat{g} : \Sigma^* \rightarrow \{ F_{00}, F_{01}, F'_{00}, F'_{01} \}^*$ tel que

$$\begin{aligned} \forall w \in \Sigma^* \quad w\hat{g} &= (w'g)F'_{00} \quad \text{si } w = w'a \\ &= (w'g)F'_{01} \quad \text{si } w = w'b \end{aligned}$$

Considérons alors le schéma S_1 défini sur les alphabets : $\mathcal{B} = \{ b \}$ et $\mathcal{F} = \{ P, Q \}$ et $\mathcal{F}_1 = \{ F_0, F_1, F'_1, F_2, F'_2, \dots, F_s, F'_s, X, X', X'', F_{00}, F'_{00}, F_{01}, F'_{01} \}$ où F_0 est le symbole de variable fonctionnelle initiale dont les schémas de définition sont :

xF_0	:= <u>Si</u> xP <u>alors</u>	$xX''X$	<u>sinon</u> xF'_1
xF_1	:= <u>Si</u> xP <u>alors</u>	xX''	<u>sinon</u> xF'_1
xF'_1	:= <u>Si</u> xQ <u>alors</u>	$xbF_1(\epsilon_1g)$	<u>sinon</u> xbF_2
xF_2	:= <u>Si</u> xP <u>alors</u>	xF_2	<u>sinon</u> xF'_2
xF'_2	:= <u>Si</u> xQ <u>alors</u>	$xbF_1(\epsilon_2g)$	<u>sinon</u> xbF_3
.....			
.....			
.....			
xF_s	:= <u>Si</u> xP <u>alors</u>	xF_s	<u>sinon</u> xF'_s
xF'_s	:= <u>Si</u> xQ <u>alors</u>	$xbF_1(\epsilon_s g)$	<u>sinon</u> xF'_s
xx	:= <u>Si</u> xP <u>alors</u>	xx'	<u>sinon</u> xx
xx'	:= <u>Si</u> xQ <u>alors</u>	xx'	<u>sinon</u> x
xx''	:= <u>Si</u> xQ <u>alors</u>	xx''	<u>sinon</u> xb
xF_{00}	:= <u>Si</u> xP <u>alors</u>	xF_{00}	<u>sinon</u> xF'_{00}
xF'_{00}	:= <u>Si</u> xQ <u>alors</u>	xF'_{00}	<u>sinon</u> xb
xF_{01}	:= <u>Si</u> xP <u>alors</u>	xF_{01}	<u>sinon</u> xF'_{01}
xF'_{01}	:= <u>Si</u> xQ <u>alors</u>	xb	<u>sinon</u> xF'_{01}

Il est facile de vérifier que le langage d'interprétation de ce schéma est le langage $LI(S_1) = \{ i_1\rho \dots (i_t\rho)(c_1)(\epsilon_{i_t}) \dots (\epsilon_{i_1}t)(c_1)/t \geq 0 \text{ et } i_1, \dots, i_t \in [s] \}$.

Soit par ailleurs le schéma S_2 défini sur les alphabets $\mathcal{B} = \{ b \}$, $\mathcal{F} = \{ P, Q \}$ et

$$\mathcal{F}_2 = \{ F_1, F'_1, F''_1, F'''_1, \dots, F_s, F'_s, F''_s, F'''_s, F_{00}, F'_{00}, F''_{00}, F'''_{00},$$

$$F_{01}, F'_{01}, F''_{01}, F'''_{01}, T, E, X, X', X'' \}.$$

où le symbole initial est F_1 et dont les schémas de définition sont :

$$\begin{aligned}
 xF_1 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX''X \ \underline{\text{sinon}} \ xF'_1 \\
 xF'_1 &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbF''_1(\eta_1\hat{g}) \ \underline{\text{sinon}} \ xbF_2 \\
 xF_2 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xF_2 \ \underline{\text{sinon}} \ xF_2 \\
 xF'_2 &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbF''_1(\eta_2\hat{g}) \ \underline{\text{sinon}} \ xbF_3 \\
 &\dots\dots\dots \\
 xF_s &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xF_s \ \underline{\text{sinon}} \ xbF'_s \\
 xF'_s &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbF''_1(\eta_s\hat{g}) \ \underline{\text{sinon}} \ xF'_s \\
 xF''_1 &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX'' \ \underline{\text{sinon}} \ xF''_1 \\
 xF''_1 &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbF''_1(\eta_1g) \ \underline{\text{sinon}} \ xbF''_2 \\
 xF'' &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xF''_2 \ \underline{\text{sinon}} \ xF''_2 \\
 xF''_2 &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbF''_1(\eta_2g) \ \underline{\text{sinon}} \ xbF''_3 \\
 &\dots\dots\dots \\
 xF''_s &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xF''_s \ \underline{\text{sinon}} \ xF''_s \\
 xF''_s &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbF''_1(\eta_sg) \ \underline{\text{sinon}} \ xF''_s \\
 xF''_{00} &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ xF''_{00} \\
 xF''_{00} &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbT \ \underline{\text{sinon}} \ xb
 \end{aligned}$$

$$\begin{aligned}
 xF_{01} &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ xF'_{01} \\
 xF'_{01} &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xb \ \underline{\text{sinon}} \ xbT \\
 xF''_{00} &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ xF''_{00} \\
 xF''_{00} &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbT \ \underline{\text{sinon}} \ xbE \\
 xF''_{01} &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ xF''_{01} \\
 xF''_{01} &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xbE \ \underline{\text{sinon}} \ xbT \\
 xT &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ xbT \\
 xE &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xE \ \underline{\text{sinon}} \ xbT \\
 xX &:= \underline{\text{Si}} \ xP \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ xX \\
 xX' &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xX' \ \underline{\text{sinon}} \ x \\
 xX'' &:= \underline{\text{Si}} \ xQ \ \underline{\text{alors}} \ xX'' \ \underline{\text{sinon}} \ xb
 \end{aligned}$$

On vérifie que le langage d'interprétation du schéma S_2 est le langage $LI(S_2) = \{ (i_1\rho) \dots (i_t\rho)(c_1)(w_1)(c_1)/i_1, \dots, i_t \in [s], t \geq 0 \text{ et } w \neq \eta_{i_1} \dots \eta_{i_t} \}$.

Les deux schémas S_1 et S_2 ainsi construits sont deux schémas sur un alphabet réduit à un seul élément et tels qu'au cours de tout calcul qui se termine le plus long mot testé est le résultat du calcul. Si l'on applique alors aux deux schémas S_1 et S_2 la construction de la conjonction que nous avons donnée précédemment, on obtient un schéma S_3 qui s'arrête pour une interprétation libre I si et seulement si les deux schémas S_1 et S_2 s'arrêtent tous les deux et tel qu'alors : $\text{Val}(S_1, I) = \text{Val}(S_3, I)$.

On sait par ailleurs que : $LI(S_3) = LI(S_1) \cap_I LI(S_2)$, c'est-à-dire :

$$LI(S_3) = \{ (i_1\rho) \dots (i_t\rho)(c1)(w1)(c1)/w = \varepsilon_{i_t} \dots \varepsilon_{i_1} \text{ et } w \neq \eta_{i_t} \dots \eta_{i_1} \}.$$

Dans ces conditions les deux schémas S_1 et S_2 pour lesquels $S_1 \equiv_I S_3$ ssi $LI(S_1) = LI(S_3)$ sont I -équivalents si et seulement si le PCP initial n'a pas de solution : en effet $LI(S_1) = LI(S_3) \Leftrightarrow$ il n'existe pas de suite d'entiers i_t, \dots, i_1 vérifiant $\varepsilon_{i_t} \dots \varepsilon_{i_1} = \eta_{i_t} \dots \eta_{i_1}$ i.e. ssi le PCP π n'a pas de solution.

Les deux schémas S_1 et S_3 sont I -équivalents si et seulement si ils sont fortement équivalents et par conséquent l'équivalence forte des schémas récursifs généralisés est indécidable.

BIBLIOGRAPHIE

- [1] ASHCROFT E., MANNA Z. et PNUELI A., *Decidable Properties of Monadic Functional Schemas*, in *Theory of Machines and Computation*, pp. 3-17, Kohavi et Paz ed. Academic Press (1971).
- [2] COUSINEAU F. G. et RIFFLET J.-M., *Schémas de Programme : Problèmes d'équivalence et complexité*, Thèse de Troisième Cycle, Université Paris 7 (février 1974).
- [3] GARLAND S. J. et LUCKHAM D. C., *Program Schemes, Recursive Schemes and Formal Languages*, *J. Comput. System Sci.* 7 (1973), n° 2, pp. 119-160.