

Décalages d'arbres

4.1	Généralités	94
4.1.1	Automate d'arbres	94
4.1.2	Décalages d'arbres irréductibles	99
4.1.3	Contexte, automate réduit et bloc synchronisant	101
4.1.4	Automates minimaux	103
4.1.5	Théorème de décomposition	108
	Fonctions d'éclatement et de fusion	108
	Énoncé et preuve du théorème	109
4.2	Décalages de type fini	111
4.2.1	Décalages de sommets et de transitions	111
4.2.2	Conjugaison des décalages de type fini	112
4.3	Les décalages presque de type fini (AFT)	116
4.3.1	Définition	116
4.3.2	Caractérisation des AFT par leur couverture de Shannon	119
4.4	Procédures de décision	120
4.4.1	Calcul de la couverture de Shannon	120
4.4.2	Automate des paires et graphe des paires	121
4.4.3	Localité	125
4.4.4	Automate fermant à gauche	126

Ce quatrième chapitre est consacré aux décalages d'arbres, et plus particulièrement à la classe des décalages d'arbres sofiques. La Partie 4.1 présente les outils pour étudier les décalages d'arbres sofiques : automate d'arbres et automate minimal. La Partie 4.1.5 est consacrée au théorème de décomposition, grâce auquel nous montrons que la conjugaison des décalages d'arbres de type fini est un problème décidable. La Partie 4.3 étudie la classe des décalages d'arbres presque de type fini, classe intermédiaire entre la classe des décalages d'arbres de type fini et la classe des décalages d'arbres sofiques.

4.1 Généralités

Dans cette partie nous nous intéressons aux décalages définis sur le monoïde libre à d générateurs \mathbb{M}_d . Comme \mathbb{M}_d est un monoïde libre son graphe de Cayley ne contient aucun cycle, c'est pourquoi on parle de décalage d'arbres dans cette partie. Rappelons que nous nous donnons un alphabet fini A .

Commençons par fixer les notations et le vocabulaire spécifiques aux décalages d'arbre. On considère l'alphabet $\{0, 1, \dots, d-1\}$, de sorte que chaque élément du monoïde \mathbb{M}_d peut être identifié à un mot sur $\{0, 1, \dots, d-1\}$. Un élément de $\{0, 1, \dots, d-1\}^*$, l'ensemble des mots sur l'alphabet $\{0, 1, \dots, d-1\}$, sera un *nœud*, et le nœud correspondant à l'élément neutre du monoïde, ou de manière équivalente au mot vide, sera appelé la *racine* et noté par ε . Si x est un nœud et que i est un élément de $\{0, 1, \dots, d-1\}$ alors on dit que $y = xi$ est le $i^{\text{ième}}$ fils de x , et que x est le père de y . Un *arbre* est une configuration, c'est-à-dire un élément de $A^{\{0,1,\dots,d-1\}^*}$; les arbres seront le plus souvent notés par t . Remarquons que les arbres considérés ici sont de rang fixe : tous les nœuds ont le même nombre de fils. Si t est un arbre et x un mot de $\{0, 1, \dots, d-1\}^*$, on rappelle que t_x désigne la lettre qui apparaît en position x dans l'arbre t . Le support élémentaire de taille n est formé de l'ensemble des mots sur $\{0, 1, \dots, d-1\}$ de longueur au plus n . Un bloc désigne un motif élémentaire, et si u est un bloc on note $|u|$ sa hauteur. On rappelle que si \mathbf{T} est un décalage d'arbres, son langage d'ordre n , noté $\mathcal{L}_n(\mathbf{T})$, est formé de tous les blocs élémentaires qui apparaissent dans des configurations de \mathbf{T} . Les nœuds d'un bloc de taille n correspondant à un mot de longueur n sur $\{0, 1, \dots, d-1\}$ sont appelés les *feuilles*, ce sont des nœuds sans fils.

Sans perte de généralité nous allons supposer dans toute la suite que les arbres considérés sont des arbres binaires, c'est-à-dire que nous nous restreignons au cas où $d = 2$. Dans ce cas nous parlerons de fils gauche et droit. On utilisera la notation (a, t_0, t_1) pour désigner l'arbre étiqueté par la lettre a à la racine, dont le fils gauche est t_0 et le fils droit t_1 . En particulier, un bloc de hauteur 1 sera noté (a, b, c) , où a étiquette la racine, b le fils gauche de la racine et c le fils droit de la racine.

4.1.1 Automate d'arbres

Dans cette partie nous présentons une version des automates finis d'arbres qui reconnaît des configurations infinies ou des blocs finis. Dans ce modèle le calcul remonte des branches infinies vers la racine. L'acceptation ou non d'un arbre est déterminée simplement par l'existence d'un calcul de l'automate remontant jusqu'à la racine, car tous les états sont choisis terminaux. Cette version d'auto-

mate d'arbres correspond à la version standard d'automates d'arbres montants, lorsque tous les états sont à la fois initiaux et terminaux. Pour d'autres version d'automates d'arbres le lecteur pourra se référer à [CDG⁺07].

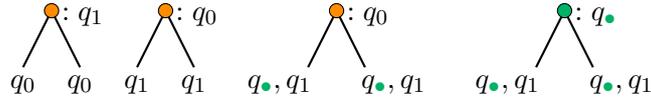
Définition 15. Un *automate fini d'arbres* est une structure $\mathcal{A} = (V, A, \Delta)$ où V est un ensemble fini d'états (ou sommets), A est un alphabet fini, et Δ est un ensemble de transitions de la forme $(q_0, q_1), a \rightarrow q$, avec $q, q_i \in V, a \in A$. Une transition $(q_0, q_1), a \rightarrow q$ est dite étiquetée par a , sortant du couple d'états (q_0, q_1) et arrivant dans l'état q .

Remarque. Dans toute la suite on omettra l'adjectif fini, et on parlera simplement d'automate d'arbres pour désigner un automate fini d'arbres.

Exemple 4.1.1. Soit l'alphabet à deux éléments $A = \{ \bullet, \circ \}$. On définit un automate \mathcal{A}_1 dont l'ensemble d'états est $Q_1 = \{q_0, q_1\}$, donné par l'ensemble de transitions suivantes :

$$\Delta_1 = \{(q_0, q_0, \bullet) \rightarrow q_1, (q_1, q_1, \circ) \rightarrow q_0\}.$$

Exemple 4.1.2. Avec l'alphabet $A = \{ \circ, \bullet \}$, toujours sur des arbres binaires, on définit un automate \mathcal{A}_2 dont l'ensemble d'états est $Q_2 = \{q_0, q_1, q_\bullet\}$, donné par l'ensemble de transitions suivantes :

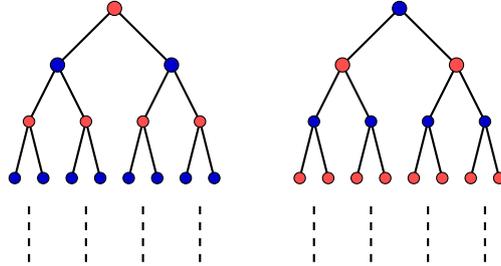


Un tel automate est dit *déterministe* si pour tout couple d'états (q_0, q_1) et pour toute lettre $a \in A$, il existe au plus une transition $(q_0, q_1), a \rightarrow q$. L'ensemble des transitions définit alors une fonction partielle $\delta : V^2 \times A \rightarrow V$. Dans ce cas, on utilisera aussi pour l'automate la notation $\mathcal{A} = (V, A, \delta)$.

Un *calcul* de l'automate $\mathcal{A} = (V, A, \Delta)$ sur un arbre t est un arbre C sur l'alphabet V tel que pour tout nœud x , s'il existe une transition $(C_{x0}, C_{x1}, t_x) \rightarrow C_x \in \Delta$. Un arbre t est *accepté* par un automate \mathcal{A} s'il existe un calcul de \mathcal{A} sur t . Il est aisé de voir que l'ensemble des arbres acceptés par un automate \mathcal{A} constitue un décalage, que l'on note $\mathbf{T}_{\mathcal{A}}$.

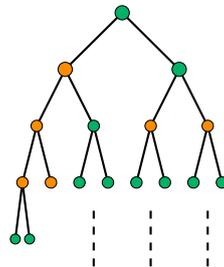
Un arbre fini est dit *complet* si chacun de ses nœuds a soit deux fils, soit aucun fils. On définit une notion de calcul d'un automate d'arbres sur un arbre fini complet de la manière suivante. Un *calcul fini* de l'automate $\mathcal{A} = (V, A, \Delta)$ sur un arbre fini complet u est un arbre fini complet C sur l'alphabet V tel que, pour tout nœud x de u , il existe une transition $(C_{x0}, C_{x1}), u_x \rightarrow C_x \in \Delta$.

Exemple 4.1.3. Les seuls arbres acceptés par l'automate de l'Exemple 4.1.1 sont les deux arbres suivants :



Sur chacun des arbres, tous les nœuds d'un même niveau partagent la même couleur et deux niveaux consécutifs ont des couleurs différentes. Le décalage d'arbres $\mathbf{T}_{\mathcal{A}_1}$ est de type fini.

Exemple 4.1.4. Les arbres acceptés par l'automate de l'Exemple 4.1.2 sont ceux qui ne contiennent aucun chemin le long duquel on trouve un nombre pair de \bullet entre deux \bullet . Le décalage d'arbres $\mathbf{T}_{\mathcal{A}_2}$ est sofique mais pas de type fini.



Soit m un entier positif. Un automate d'arbres déterministe \mathcal{A} est m -local si pour toutes paires d'arbres (t, t') contenant un même bloc b de hauteur m en position x dans t et en position x' dans t' , et pour tous calculs C sur t et C' sur t' de l'automate \mathcal{A} , on a $C_x = C'_{x'}$.

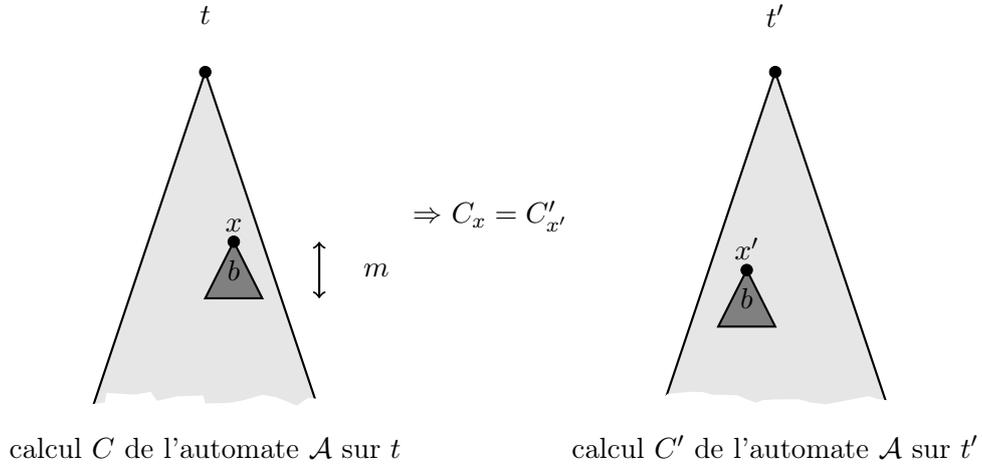


FIGURE 27: Notion de m -localité pour un automate d'arbres \mathcal{A} , illustrée sur deux calculs de \mathcal{A} sur les arbres t et t' dans lesquels un même bloc b de hauteur m apparaît.

Autrement dit une mémoire de hauteur m suffit à déterminer l'état atteint par n'importe quel calcul C de l'automate. On dit dans ce cas que le bloc b *force* l'état C_x dans l'automate \mathcal{A} . On dit simplement qu'un automate d'arbres est *local* s'il est m -local pour un entier positif m .

Si un automate d'arbres $\mathcal{A} = (Q, A, \delta)$ est déterministe, on peut définir par induction une fonction δ^k sur les blocs de hauteur k

- si $k = 0$ alors $\delta^0 = \delta$;
- si $k \geq 1$ alors $\delta^k((t_0, t_1, a)) = \delta(\delta^{k-1}(t_0), \delta^{k-1}(t_1), a)$.

Ces fonctions δ^k seront aussi appelées δ quand aucune confusion n'est possible sur la valeur de l'entier k à choisir.

Les automates d'arbres acceptent exactement les décalages d'arbres sofique, et les automates d'arbres locaux accepte exactement les décalages d'arbres de type fini. Les démonstrations de ces caractérisations sont similaires à celles des résultats correspondants pour les décalages sur \mathbb{N} ou sur \mathbb{Z} (voir [LM95] ou [Kit98]).

Proposition 4.1.1 ([AB09]). *Tout décalage d'arbres de type fini est accepté par un automate d'arbre déterministe local. Réciproquement tout décalage d'arbres accepté par un automate d'arbres déterministe local est de type fini.*

Démonstration. Soit \mathbf{T}_F un décalage d'arbres de type fini défini par l'ensemble fini de motifs interdits F . Sans perte de généralité, on peut supposer que les motifs de l'ensemble F sont tous des blocs de hauteurs m avec m entier strictement positif. On définit un automate d'arbres déterministe $\mathcal{A} = (V, A, \delta)$ dont les états sont les blocs du langage d'ordre m : $V = \mathcal{L}_m(X)$. Pour $p_i \in V, a \in A$, si le bloc $q = (a, p_0, p_1)$ de hauteur $m + 1$ est un bloc autorisé de \mathbf{T} , alors $\delta(p_0, p_1, a) = \text{tronc}(q)$, où $\text{tronc}(q)$ est le bloc de hauteur m tel que $\text{tronc}(q)_x = q_x$ pour $x \in \{0, 1\}^{\leq m-1}$. La fonction partielle δ n'est pas définie sinon. Par construction, l'automate \mathcal{A} est déterministe et m -local. Il est clair que cet automate reconnaît le décalage \mathbf{T} , ce qui prouve la première partie de la proposition.

Soit \mathbf{T} un décalage d'arbres et $\mathcal{A} = (V, A, \delta)$ un automate d'arbre m -local qui accepte \mathbf{T} . On définit F comme l'ensemble des blocs interdits de hauteur $m + 1$ dans \mathbf{T} . Il est aisé de voir que $\mathbf{T} \subseteq \mathbf{T}_F$. Supposons maintenant que $t \in \mathbf{T}_F$. On définit un calcul de l'automate \mathcal{A} sur le bloc t de la manière suivante. Pour tout $x \in \{0, 1\}^*$, on appelle C_x l'état sur lequel termine tout calcul de l'automate \mathcal{A} sur le bloc de hauteur m enraciné en x dans l'arbre t . Soit b le bloc de hauteur $m + 1$ enraciné en x dans t . Comme $t \in \mathbf{T}_F$, b est un bloc autorisé pour \mathbf{T} , donc ce motif apparaît dans un arbre $t' \in \mathbf{T}$ en position $y \in \{0, 1\}^*$. Soit C' le calcul de l'automate \mathcal{A} sur t' . Ainsi $\delta(c_{y0}, \dots, c_{yd}, t'_y) = c_y$. Comme l'automate \mathcal{A} est m -local, $C_{xi} = C_{yi}$ pour $0 \leq i \leq d$ et $C_y = C_x$. Comme $t_x = t'_y$ il s'ensuit que $\delta(C_{x0}, \dots, C_{xd}, t_x) = C_x$. Ainsi c est un calcul de l'automate \mathcal{A} sur t et $t \in \mathbf{T}$, ce qui achève la démonstration de la proposition. \square

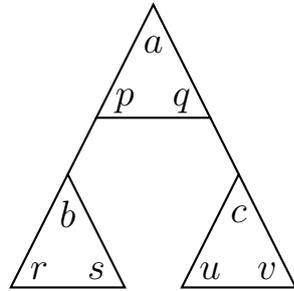
Proposition 4.1.2. *Un décalage d'arbres est sofique si et seulement s'il est reconnu par un automate d'arbres déterministe.*

Démonstration. Soit \mathbf{T} un décalage sofique sur l'alphabet A . Alors \mathbf{T} est l'image d'un décalage de type fini \mathbf{T}' par une m -fonction de bloc Φ . Soit $V = \mathcal{L}_m(\mathbf{T}')$ l'ensemble des blocs de hauteur m qui apparaissent dans \mathbf{T}' . On définit l'automate d'arbres $\mathcal{A} = (V, A, \delta)$, dont les transitions sont données par la règle suivante : $(p, q, a) \rightarrow r \Leftrightarrow a = \phi(r)$ où ϕ est la fonction locale qui définit Φ . Alors par définition cet automate est déterministe et reconnaît bien le décalage \mathbf{T} .

Réciproquement soit $\mathcal{A} = (V, A, \delta)$ un automate d'arbres déterministe qui accepte un décalage d'arbres $\mathbf{T}_{\mathcal{A}}$. On définit un nouvel alphabet

$$A' = \{(p, q, a) \in V^2 \times A \mid \exists r \in V \text{ tel que } \delta(p, q, a) = r \text{ dans } \mathcal{A}\},$$

et sur cet alphabet A' on définit le décalage d'arbres de type fini \mathbf{T}' dont les motifs autorisés sont de la forme :



avec $p = \delta(r, s, b)$ et $q = \delta(u, v, c)$

On appelle Φ la 1-fonction de bloc qui à un arbre t' sur A' associe l'arbre t sur A obtenu en ne conservant que la partie dans A de chaque lettre de A' . Alors le décalage $\Phi(\mathbf{T}')$ est un décalage d'arbre sofique comme image par une fonction de bloc d'un décalage d'arbres de type fini, et $\Phi(\mathbf{T}')$ est exactement \mathbf{T} ce qui achève la démonstration. \square

4.1.2 Décalages d'arbres irréductibles

Un *code préfixe fini et complet* de $\{0, 1\}^*$ est un ensemble préfixe (i.e. aucun mot n'est préfixe d'un autre mot) P de mots finis sur $\{0, 1\}^*$ tel que tout mot de $\{0, 1\}^*$ de longueur supérieure à celles des mots de P possède un préfixe dans P . Un décalage d'arbres \mathbf{T} est *irréductible* si pour toute paire de blocs $u, v \in \mathcal{L}(\mathbf{T})$, il existe un arbre $t \in \mathbf{T}$ et un code préfixe fini et complet $P \subset \{0, 1\}^{\geq |u|}$, tel que u est le sous-arbre de t à la racine ε , et v est un sous-arbre de t en position x pour tout $x \in P$.

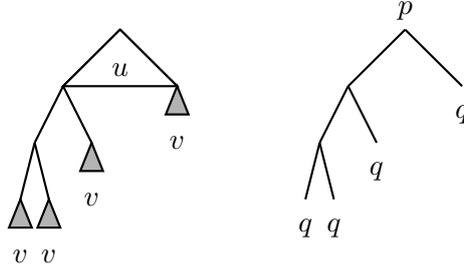


FIGURE 28: Décalage d'arbres irréductible et automate d'arbres irréductible.

Un automate d'arbres est *irréductible* si pour toute paire d'états p, q il existe un code préfixe fini et complet $P \subset \{0, 1\}^*$ et un calcul fini C de l'automate sur un motif u tel que $C_\varepsilon = p$ et $C_x = q$ pour tout $x \in P$. On dit dans ce cas qu'il existe un *hyper-chemin de q à p* étiqueté par u . Pour deux états p, q d'un automate, on dit que p est *accessible* depuis q s'il existe un hyper-chemin de q à p .

Proposition 4.1.3. *Un automate d'arbres irréductible accepte un décalage d'arbres sofique irréductible.*

Démonstration. Soit \mathcal{A} un automate d'arbres irréductible. On appelle $\mathbf{T}_{\mathcal{A}}$ le décalage d'arbres sofique accepté par l'automate \mathcal{A} . Soient u et v deux blocs dans le langage de $\mathbf{T}_{\mathcal{A}}$, et on suppose que le bloc u est de hauteur k . Alors il existe un arbre $t_u \in \mathbf{T}_{\mathcal{A}}$ dans lequel u apparaît à la racine, et un arbre $t_v \in \mathbf{T}_{\mathcal{A}}$ dans lequel v apparaît à la racine. Comme $t_u \in \mathbf{T}_{\mathcal{A}}$ il existe un calcul C de \mathcal{A} sur t_u , dans lequel les feuilles du motifs p sont étiquetées par des états p_i pour $0 \leq i \leq 2^{k-1}$. De même il existe un calcul C' de \mathcal{A} sur t_v dans lequel $C'_\varepsilon = q$. Comme \mathcal{A} est irréductible, il existe un hyper-chemin de p_i à q étiqueté par w_i , et associé à un code préfixe fini et complet $P_i \subset \{0, 1\}^{\geq |w_i|}$. On définit le code préfixe fini complet $P = \bigcup_{0 \leq i \leq 2^{k-1}} w_i P_i$. On définit l'arbre t de la manière suivante :

- le bloc u apparaît à la racine de t ;
- le bloc w_i apparaît à la $i^{\text{ème}}$ feuille de u ;
- l'arbre t_v est enraciné en chaque feuille des w_i .

Cet arbre est accepté par l'automate \mathcal{A} donc appartient bien au décalage \mathbf{T} . Ceci montre que \mathbf{T} est irréductible. \square

La réciproque de la Proposition 4.1.3 peut être démontrée en utilisant la notion de couverture de Shannon d'un décalage sofique, définie dans la Partie 4.1.4. Ainsi les notions d'irréductibilité pour les décalages et pour les automates sont directement corrélés.

4.1.3 Contexte, automate réduit et bloc synchronisant

Les automates d'arbres reconnaissent les décalages d'arbres soifiques, mais pour l'instant nous n'avons pas trouvé d'automate d'arbres canonique reconnaissant un décalage d'arbres \mathbf{T} . Dans ce but, nous définissons la notion de contexte d'un bloc fini, qui correspond à la notion de *follower set* de Lind et Marcus [LM95] pour les décalages sur \mathbb{N} ou \mathbb{Z} .

Soit \mathbf{T} un décalage d'arbre. Un *contexte* c est un motif fini dont une des feuilles joue un rôle spécial, on dira qu'elle est *marquée*. Si u est un motif, $c(u)$ est le motif obtenu en remplaçant la feuille marquée du contexte c par le motif u . Si $c(u)$ apparaît dans un motif du langage $\mathcal{L}(\mathbf{T})$, on dit alors que c est un *contexte du motif u dans \mathbf{T}* . Étant donné un bloc u , on note par $\text{cont}_{\mathbf{T}}(u)$ l'ensemble des tous les contextes de u dans \mathbf{T} . Étant donné un automate d'arbres $\mathcal{A} = (V, A, \Delta)$ qui accepte un décalage d'arbres sofique \mathbf{T} , le *contexte* d'un état $q \in V$ dans \mathcal{A} est l'ensemble des motifs u avec une feuille marquée en position x sur lesquels il existe un calcul fini C de l'automate \mathcal{A} avec $C_x = q$. On le note $\text{cont}_{\mathcal{A}}(q)$. Remarquons que le contexte d'un motif u dans \mathbf{T} est l'union des contextes des états p qui terminent un calcul de \mathcal{A} sur u . Un décalage d'arbres sofique a donc un nombre fini de contextes distincts.

Soit $\mathcal{A} = (V, A, \Delta)$ un automate d'arbre. On dit que l'automate \mathcal{A} est *réduit* si $p \neq q$ implique $\text{cont}_{\mathcal{A}}(p) \neq \text{cont}_{\mathcal{A}}(q)$ pour $p, q \in V$, autrement dit si deux états distincts ont des contextes distincts. Si $\mathcal{A} = (V, A, \Delta)$ et $\mathcal{A}' = (V', A, \Delta')$ sont deux automates d'arbres déterministes, une *réduction* de \mathcal{A} vers \mathcal{A}' est une fonction h de V vers V' telle que, pour toute lettre $a \in A$, on a $p, q, a \rightarrow r \in \Delta$ si et seulement si $(h(p), h(q), a) \rightarrow r \in \Delta'$. Si \mathcal{A}' est un automate obtenu par réduction à partir d'un automate \mathcal{A} , alors \mathcal{A} et \mathcal{A}' définissent le même décalage.

Soit \mathbf{T} un décalage d'arbres sofique déterministe accepté par un automate d'arbres $\mathcal{A} = (V, A, \Delta)$. On définit un automate d'arbres déterministe et réduit $R(\mathcal{A})$ qui accepte \mathbf{T} , et que l'on appelle *réduction* de l'automate \mathcal{A} . Les états de $R(\mathcal{A})$ sont les classes de la partition la plus grossière de l'ensemble V telle que si les états p, q appartiennent à la même classe, alors pour toute lettre $a \in A$ et pour tout état $r \in Q$, $\delta(p, r, a)$ et $\delta(q, r, a)$ (resp. $\delta(r, p, a)$ et $\delta(r, q, a)$)

appartiennent à la même classe, et la transition $\delta(p, r, a)$ (resp. $\delta(r, p, a)$) est définie si et seulement si la transition $\delta(q, r, a)$ (resp. $\delta(r, q, a)$) est définie. On note par $[p]$ la classe de l'état p . La transition $([p], [q], a \rightarrow [\delta(p, q, a)])$ est une transition de l'automate $R(\mathcal{A})$ si et seulement si $\delta(p, q, a)$ est définie. Cette définition ne dépend pas du choix du représentant de la classe et est donc bien fondée. Par définition, l'automate $R(\mathcal{A})$ est réduit, et la fonction h définie par $h(p) = [p]$ est une réduction de \mathcal{A} vers $R(\mathcal{A})$.

L'algorithme de minimisation pour les automates d'arbres déterministes qui acceptent des blocs finis, que l'on peut trouver dans [CDG⁺07, Section 1.5], peut être appliqué pour calculer une réduction d'un automate d'arbres acceptant le même décalage d'arbre.

Soit $\mathcal{A} = (V, A, \Delta)$ un automate d'arbres déterministe qui reconnaît un décalage d'arbres sofique \mathbf{T} , et soit u un bloc (resp. un motif). On dit que u est un *bloc synchronisant* (resp. un *motif synchronisant*) de l'automate \mathcal{A} s'il existe au moins un calcul fini de l'automate \mathcal{A} sur u , et si de plus tous les calculs finis de \mathcal{A} sur u terminent dans un même état $q \in Q$. On dit alors que le bloc u *synchronise* sur l'état q . Un automate d'arbres déterministe qui possède un bloc synchronisant est appelé *automate synchronisé*.

Remarque. Si u est un bloc synchronisant, alors tout motif dans lequel u apparaît à la racine est synchronisant.

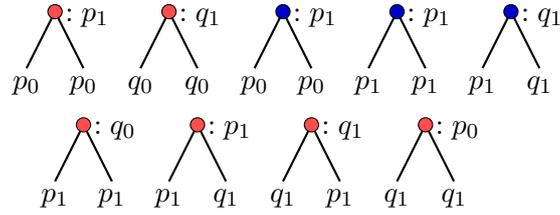
Soit $\mathcal{A} = (V, A, \Delta)$ un automate d'arbre. On définit *l'automate déterminisé de \mathcal{A}* , noté $\mathcal{D}(\mathcal{A})$, comme la partie accessible depuis l'état V dans l'automate d'arbres $(\mathfrak{P}(V), A, \delta')$, avec pour $P, Q \in \mathfrak{P}(V)$, $\delta'(P, Q, a) = \{\delta(p, q, a) \mid p \in P, q \in Q\}$ si l'ensemble est non vide, et n'est pas défini sinon.

Proposition 4.1.4. *Un automate \mathcal{A} est synchronisé si et seulement si son automate déterminisé $\mathcal{D}(\mathcal{A})$ possède un état qui est un singleton.*

La démonstration de la Proposition 4.1.4 découle directement de la définition d'un bloc synchronisant.

Proposition 4.1.5. *Il existe des automates d'arbres déterministes, irréductibles et réduits qui ne sont pas synchronisés.*

Démonstration. Soit \mathbf{T} le décalage d'arbres plein sur l'alphabet à deux éléments $A = \{\bullet, \circ\}$. Il est trivialement accepté par un automate d'arbres à un seul état. Mais il est aussi accepté par l'automate d'arbres déterministe et réduit $\mathcal{A} = (V, A, \delta)$ dont les règles de transitions sont présentées ci-dessous.

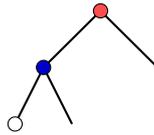


L'automate A est irréductible : il existe un hyper-chemin

$$p_0 \rightsquigarrow p_1 \rightsquigarrow q_0 \rightsquigarrow q_1 \rightsquigarrow p_0,$$

où $p \rightsquigarrow q$ désigne un hyper-chemin de l'état p vers l'état q .

L'automate A est réduit, car deux états différents ont des contextes différents. En effet, les états q_0, q_1 n'ont pas même contexte que les états p_0, p_1 , car le motif dont la racine est étiquetée par \bullet , et dont la feuille gauche est marquée est un contexte des états p_0, p_1 mais pas de q_0, q_1 . De plus, le motif suivant, dans lequel la feuille marquée est représentée par un \circ :



est un contexte de p_0, q_1 mais pas de p_1, q_0 .

L'automate $\mathcal{D}(\mathcal{A})$ contient deux états $V = \{p_0, q_0, p_1, q_1\}$ et $\{p_1, q_1\}$, mais aucun état réduit à un singleton et donc \mathcal{A} n'est pas synchronisé et la Proposition 4.1.5 est démontrée. \square

En particulier, la Proposition 4.1.5 nous apprend que, contrairement au cas des décalages sur \mathbb{N} ou \mathbb{Z} , un décalage d'arbres sofique peut être accepté par plusieurs automates d'arbre irréductibles, déterministes et réduits.

4.1.4 Automates minimaux

Comme nous venons de le voir dans la Partie 4.1.3, les décalages d'arbres différent des décalages unidimensionnels au moins par le fait qu'il existe des automates d'arbres réduits, irréductibles et déterministes qui ne possèdent pas de bloc synchronisant (voir la Proposition 4.1.5). Il est cependant possible de construire un tel automate en ajoutant la condition qu'il soit synchronisé. Dans

cette partie nous présentons deux automates minimaux, la couverture de Shannon (ou couverture de Fischer) et la couverture de Krieger, qui reposent tous les deux sur la notion de contexte d'un arbre (contexte d'un arbre fini pour la couverture de Shannon, et contexte d'un arbre infini pour la couverture de Krieger).

Soit \mathbf{T} un décalage d'arbres sofique. L'automate des contextes du décalage \mathbf{T} est l'automate d'arbres déterministe $\mathcal{C} = (V, A, \Delta)$, où V est l'ensemble des contextes non vides des blocs finis apparaissant dans le décalage \mathbf{T} . Dans le cas d'un décalage d'arbres \mathbf{T} sofique, l'ensemble V est fini, mais ce n'est pas le cas lorsque \mathbf{T} n'est pas sofique. Les transitions de l'automate \mathcal{C} sont de la forme $(\text{cont}_{\mathbf{T}}(u), \text{cont}_{\mathbf{T}}(v)), a \rightarrow \text{cont}_{\mathbf{T}}(a, u, v)$, avec $u, v \in \mathcal{L}(\mathbf{T})$.

Remarque. Il est possible de définir un automate des contextes pour un décalage d'arbre qui n'est pas sofique, mais dans ce cas cet automate possède une infinité d'états.

Proposition 4.1.6. *L'automate des contextes d'un décalage d'arbres sofique est synchronisé.*

Démonstration. Soit \mathcal{C} l'automate des contextes d'un décalage d'arbres sofique \mathbf{T} . Il existe un calcul fini C_1 de l'automate \mathcal{C} sur un certain bloc u_1 qui se termine dans l'état $\text{cont}_{\mathbf{T}}(u_1)$. Supposons que ce bloc u_1 n'est pas un bloc synchronisant de l'automate \mathcal{C} . Il existe donc un autre calcul C_2 de l'automate \mathcal{C} sur le bloc u_1 qui termine sur un état $\text{cont}_{\mathbf{T}}(u_2)$ pour un certain bloc u_1 tel que $\text{cont}_{\mathbf{T}}(u_2) \neq \text{cont}_{\mathbf{T}}(u_1)$. Ainsi $\text{cont}_{\mathbf{T}}(u_2) \subsetneq \text{cont}_{\mathbf{T}}(u_1)$, car $\text{cont}_{\mathbf{T}}(u_2) \neq \text{cont}_{\mathbf{T}}(u_1)$. Si le bloc u_2 n'est pas un bloc synchronisant de l'automate \mathcal{C} , on peut à nouveau appliquer le même raisonnement pour construire un bloc u_3 . En itérant ce procédé, nous obtenons soit un bloc synchronisant pour l'automate \mathcal{C} , soit une suite strictement décroissante de contextes. Et comme il n'existe qu'un nombre fini de contextes, cette deuxième option est à exclure, ce qui prouve l'existence d'un bloc synchronisant dans l'automate des contextes \mathcal{C} . \square

Si \mathcal{A} est un automate d'arbres, on dit qu'une composante de l'automate est *minimale* si tout hyper-chemin partant d'un état de la composante arrive aussi dans la composante (voir le chapitre [BBrEP10] pour la définition en dimension 1).

Proposition 4.1.7. *Si \mathbf{T} est un décalage d'arbres sofique irréductible, alors son automate des contextes \mathcal{C} possède une unique composante minimale irréductible \mathcal{S} , obtenue en ne conservant que les états accessibles depuis l'état $\text{cont}_{\mathbf{T}}(z)$, où z est un bloc synchronisant de \mathcal{C} . L'automate \mathcal{S} est la couverture de Shannon (aussi appelée couverture de Fischer) du décalage \mathbf{T} .*

Démonstration. Comme les états de l'automate \mathcal{S} sont les états accessibles depuis l'état $q = \text{cont}_{\mathbf{T}}(z)$, \mathcal{S} est une composante irréductible minimale. En effet, soit $p = \text{cont}_{\mathbf{T}}(u)$ un état accessible à partir de l'état q . Il existe par définition un hyper-chemin de q à p étiqueté par un motif v de \mathbf{T} , et le bloc u est synchronisant. Comme \mathbf{T} est un décalage irréductible, il existe un motif v et un code préfixe fini complet P formé de mots de hauteur au moins la hauteur de z , tel que z est enraciné à la racine de v et tel que u est un sous-arbre de v enraciné en tout nœud $x \in P$. Comme tout calcul de l'automate des contextes \mathcal{C} sur v se termine dans l'état p , il existe un hyper-chemin de p à q .

Montrons à présent que \mathcal{S} est unique. Si un état $p = \text{cont}_{\mathbf{T}}(u)$ appartient à une autre composante R irréductible et minimale de l'automate \mathcal{C} , par irréductibilité du décalage \mathbf{T} , il existe un motif w de \mathbf{T} et un code préfixe fini complet P formé de mots de hauteur au moins la hauteur de z , tel que z est enraciné à la racine de w et tel que u est un sous-arbre de w enraciné en tout nœud $x \in P$. Par définition de l'automate \mathcal{C} , il existe alors un hyper-chemin de p à q . Donc q appartient à R , ce qui implique que $R = \mathcal{S}$. Donc \mathcal{S} est l'unique composante minimale irréductible de \mathcal{C} . \square

On définit la *couverture de Shannon* (aussi appelée couverture de Fischer) d'un décalage d'arbres sofique irréductible \mathbf{T} comme l'unique composante maximale irréductible \mathcal{S} de son automate des contextes \mathcal{C} , obtenue en ne conservant que les états accessibles depuis l'état $\text{cont}_{\mathbf{T}}(z)$, où z est un bloc synchronisant de \mathcal{C} . Comme le bloc z est synchronisant et que l'automate \mathcal{C} est déterministe, tout état de cette composante est le contexte d'un bloc synchronisant. La définition de la couverture de Shannon ne dépend donc pas du choix du bloc synchronisant z .

Montrons que les états accessibles depuis $\text{cont}_{\mathbf{T}}(z)$ forment une composante irréductible, et que celle-ci est l'unique composante irréductible de l'automate \mathcal{C} . Il suffit pour cela de montrer qu'il existe un hyper-chemin depuis tout état de l'automate \mathcal{C} vers $\text{cont}_{\mathbf{T}}(z)$. Soit $p = \text{cont}_{\mathbf{T}}(u)$ un état de \mathcal{C} . Comme \mathbf{T} est irréductible, il existe un motif w de \mathbf{T} et un code préfixe fini complet P de $\{0, 1\}^{\geq \text{hauteur}(z)}$, tel que z est un sous-arbre de w enraciné en ε , et u est un sous-arbre de w enraciné en tout $x \in P$. Soit C un calcul de \mathcal{C} sur le motif w . Alors $C_\varepsilon = \text{cont}_{\mathbf{T}}(z)$, et pour toute feuille $x \in P$, $C_x = p$. Donc il existe un hyper-chemin de p vers $\text{cont}_{\mathbf{T}}(z)$. Enfin, il est facile de vérifier que l'automate \mathcal{S} accepte le décalage \mathbf{T} (cela découle directement de l'irréductibilité du décalage \mathbf{T}).

La couverture de Shannon d'un décalage sofique irréductible \mathbf{T} étant bien définie, nous montrons qu'il s'agit de l'unique automate d'arbres déterministe, irréductible et synchronisé qui accepte ce décalage \mathbf{T} .

Proposition 4.1.8. *Deux automates d'arbres réduits, déterministes, irréductibles et synchronisés acceptant le même décalage sofique irréductible sont égaux à un renommage des états près.*

Démonstration. Soient $\mathcal{A} = (V, A, \Delta)$ et $\mathcal{A}' = (V', A, \Delta')$ deux automates d'arbres réduits, déterministes, irréductibles et synchronisés acceptant le même décalage sofique irréductible \mathbf{T} . Soit $u \in \mathcal{L}_m(X)$ (resp. $v \in \mathcal{L}_m(X)$) un bloc synchronisant de l'automate \mathcal{A} (resp. \mathcal{A}'). Nous supposons que le bloc z synchronise sur l'état p dans l'automate \mathcal{A} . Comme \mathbf{T} est irréductible, il existe un motif z du décalage \mathbf{T} et un code préfixe fini complet $P \subset \{0, 1\}^{\geq m-1}$, tels que le bloc u est un sous-arbre du motif z enraciné en ε , et tels que v est un sous-arbre du motif z enraciné en toute feuille $x \in P$. Comme les automates \mathcal{A} et \mathcal{A}' sont déterministes, le motif z est un motif synchronisant pour les deux automates \mathcal{A} et \mathcal{A}' . Chaque calcul de l'automate \mathcal{A} sur le motif z termine sur l'état p , tandis que chaque calcul de l'automate \mathcal{A}' sur le motif z termine sur un état p' dans V' .

Nous définissons une bijection $\varphi : V \rightarrow V'$ de la manière suivante. Soit q un état de l'automate \mathcal{A} . Comme \mathcal{A} est irréductible, il existe un calcul C de \mathcal{A} sur un motif w , et un code préfixe P tels que $C_\varepsilon = q$ et $C_x = p$ pour tout $x \in P$. Puisque l'automate \mathcal{A} est déterministe et que le bloc z est un bloc synchronisant pour l'automate \mathcal{A} , tout calcul fini de l'automate \mathcal{A} sur w termine dans l'état q . Donc le contexte de q dans \mathcal{A} est le même que le contexte de w dans \mathbf{T} . De plus, tout calcul fini de l'automate \mathcal{A}' sur w termine dans un état q' tel que le contexte de q' dans \mathcal{A}' est le même que le contexte de w dans \mathbf{T} . Comme l'automate \mathcal{A} (resp. l'automate \mathcal{A}') est réduit, deux états avec le même contexte dans \mathcal{A} (resp. dans \mathcal{A}') sont en fait égaux. Donc si $q = \text{cont}_{\mathbf{T}}(w)$, $r = \text{cont}_{\mathbf{T}}(v)$, et si $(q, r), a \rightarrow s$ est une transition de Δ , nous avons $s = \text{cont}_{\mathbf{T}}(a, w, v)$. Dans ce cas nous définissons $\varphi(q) = q'$. Ainsi φ définit un isomorphisme entre \mathcal{A} et \mathcal{A}' , où φ est une bijection telle que $(q, r), a \rightarrow s$ est une transition de Δ si et seulement si $(\varphi(q), \varphi(r)), a \rightarrow \varphi(s)$ est une transition de Δ' . \square

Soit t un arbre infini sur l'alphabet A et c un contexte. On désigne par $c(t)$ le motif c dans lequel la feuille marquée du contexte c est remplacée par l'arbre t . Si le motif infini $c(t)$ apparaît dans un arbre du décalage \mathbf{T} , on dit que c est un *contexte de t dans \mathbf{T}* . Étant donné un arbre t sur l'alphabet A , on note par $\text{cont}_{\mathbf{T}}(t)$ l'ensemble des contextes de t dans \mathbf{T} . La *couverture de Krieger* d'un décalage d'arbres \mathbf{T} est l'automate d'arbre dont les états sont les ensembles non vides de la forme $\text{cont}_{\mathbf{T}}(t)$, et dont les transitions sont $(\text{cont}_{\mathbf{T}}(t), \text{cont}_{\mathbf{T}}(t'), a) \rightarrow \text{cont}_{\mathbf{T}}(a, t, t')$, où t, t' sont deux arbres sur l'alphabet A . Ainsi définie, la couverture de Krieger d'un décalage d'arbres \mathbf{T} est un automate d'arbres qui accepte \mathbf{T} . La couverture de Krieger diffère de l'automate des

contextes, car les contextes que l'on considère ici sont ceux d'arbres infinis, alors que les contextes de l'automate des contextes sont ceux d'arbres finis. Quel lien existe-t-il entre ces deux automates ? Et quel est le lien entre les couvertures de Shannon et de Krieger ? Nous allons montrer ces liens dans la Proposition 4.1.9, qui étend aux arbres une situation similaire pour les mots [BBrEP10].

On dit qu'un automate d'arbres $\mathcal{A} = (V, A, \Delta)$ est un *sous-automate* d'un automate d'arbres $\mathcal{A}' = (V', A, \Delta')$ sur le même alphabet A si $V \subseteq V'$ et si $\Delta = \Delta' \cap (V \times V \times A \times V)$. Un sous-automate \mathcal{A} est *minimal* si toute transition du sur-automate \mathcal{A}' partant d'un état de \mathcal{A} termine dans un état de \mathcal{A} .

Proposition 4.1.9. *La couverture de Krieger d'un décalage d'arbres sofique \mathbf{T} est, à isomorphisme près, un sous-automate de l'automate des contextes \mathcal{C} du décalage \mathbf{T} . Cet automate est réduit et synchronisé. Si le décalage \mathbf{T} est irréductible, la couverture de Krieger de \mathbf{T} a un unique sous-automate minimal et irréductible qui est la couverture de Shannon du décalage \mathbf{T} .*

Démonstration. Soit $\mathcal{K} = (V, A, \Delta)$ la couverture de Krieger d'un décalage d'arbres sofique \mathbf{T} et $\mathcal{C} = (V', A, \Delta')$ son automate des contextes. Soit t un arbre infini sur l'alphabet A et u_n son sous-arbre de hauteur n enraciné en ε . Ainsi pour tout entier $i \geq 0$, on a $\text{cont}_{\mathbf{T}}(u_{i+1}) \subseteq \text{cont}_{\mathbf{T}}(u_i)$. Comme le nombre de contextes du décalage \mathbf{T} est fini, il existe un entier strictement positif n tel que $\text{cont}_{\mathbf{T}}(u_{n+i}) = \text{cont}_{\mathbf{T}}(u_n)$ pour tout entier positif i . On définit alors $s(t) = \text{cont}_{\mathbf{T}}(u_n)$.

Montrons alors que la fonction de V dans V' qui à un contexte $\text{cont}_{\mathbf{T}}(t)$ associe $s(t)$ est bien définie, et qu'elle est de plus injective. Tout d'abord il est clair que $\text{cont}_{\mathbf{T}}(t) \subseteq s(t)$. Soit $c \in s(t) - \text{cont}_{\mathbf{T}}(t)$ un contexte dont la feuille marquée est en position x . Il existe un entier strictement positif n tel que $c(u_i) \in \mathcal{L}(\mathbf{T})$ pour tout entier $i \geq n$. Soit s un arbre infini tel que c est le sous-arbre de s enraciné en ε et t est le sous-arbre de s enraciné en x . Alors $s \in \mathbf{T}$ et $c \in \text{cont}_{\mathbf{T}}(t)$, donc $\text{cont}_{\mathbf{T}}(c) = s(t)$. En conséquence de quoi, si t et t' sont deux arbres infinis, alors $\text{cont}_{\mathbf{T}}(t) = \text{cont}_{\mathbf{T}}(t')$ si et seulement si $s(t) = s(t')$. Donc la fonction est bien définie et injective, et la couverture de Krieger est un sous-automate de l'automate des contextes.

Montrons à présent que l'automate \mathcal{K} est réduit et synchronisé. Si p est un état de \mathcal{K} égal au contexte $\text{cont}_{\mathbf{T}}(t)$ d'un arbre t sur l'alphabet A , alors $\text{cont}_{\mathcal{K}}(p) = \text{cont}_{\mathbf{T}}(t)$. L'automate \mathcal{K} est donc réduit. Soit $r_{\mathcal{A}}(t)$ le nombre d'états de l'automate \mathcal{A} sur lesquels un calcul de l'automate \mathcal{A} sur l'arbre t peut terminer. Par la Proposition 4.1.6, l'automate des contextes \mathcal{C} est synchronisé. Donc il existe un motif u tel que $r_{\mathcal{C}}(\text{cont}_{\mathbf{T}}(u)) = 1$. Il existe donc un arbre $t \in \mathbf{T}$ dont u est le sous-arbre enraciné en ε . Ainsi tous les calculs de l'automate des contextes \mathcal{C} sur l'arbre t terminent sur l'état $\text{cont}_{\mathbf{T}}(u)$. Comme

\mathcal{K} est un sous-automate de \mathcal{C} , tous les calculs de \mathcal{K} sur t terminent sur l'état $\text{cont}_{\mathbf{T}}(u)$. L'automate \mathcal{K} est donc synchronisé.

Supposons maintenant que le décalage \mathbf{T} est en plus irréductible. Soit z un bloc synchronisant de l'automate \mathcal{K} tel que tout calcul fini de \mathcal{K} sur z termine sur l'état p . Soit W l'ensemble des états accessibles depuis l'état p . Montrons que l'automate $\mathcal{K}' = (W, A, \Delta)$ est un sous-automate minimal de \mathcal{K} . Pour tout état $q \in W$, il existe un hyper-chemin de p vers q étiqueté par u . De plus, si on appelle v l'arbre fini obtenu en remplaçant chaque feuille de u par z , tout calcul de l'automate \mathcal{K} sur v termine dans l'état q . Comme \mathbf{T} est irréductible, il existe un arbre t dans \mathbf{T} et un code préfixe fini complet $P \subset \{0, 1\}^{\geq \text{hauteur}(z)}$, tel que z est le sous-arbre de t enraciné en ε , et v est le sous-arbre de t enraciné en tout $x \in P$. Il existe donc un hyper-chemin de q vers p dans l'automate \mathcal{K} . De plus, si r est un état d'un sous-automate minimal et irréductible de l'automate \mathcal{K} , sur lequel termine un calcul fini de \mathcal{K} sur un motif w , par irréductibilité du décalage \mathbf{T} , il existe un arbre t' dans \mathbf{T} et un code préfixe fini complet $Q \subset \{0, 1\}^{\geq \text{hauteur}(z)}$, tels que z est le sous-arbre de t' enraciné en ε , et que w est le sous-arbre de t' enraciné en tout $x \in Q$. Donc il existe un hyper-chemin de r vers p , et $p \in W$. Ainsi l'automate \mathcal{K}' est l'unique sous-automate minimal irréductible de l'automate \mathcal{K} , et c'est la couverture de Shannon du décalage \mathbf{T} . \square

4.1.5 Théorème de décomposition

Le théorème de décomposition pour les décalages sur \mathbb{Z} nous apprend que toute conjugaison entre deux décalages de type fini [Wil73] (ou sofiques [LM95, Exercice 7.1.10 p. 225]) peut être décomposée en une suite finie de conjugaisons élémentaires, les éclatements et les fusions. Le point clé de la démonstration de ce résultat est la possibilité de réduire la mémoire d'une fonction de bloc grâce à des éclatements d'états. Dans cette partie, nous montrons un théorème de décomposition pour les décalages d'arbres qui étend le théorème de décomposition pour les décalages d'arbres de type fini présenté dans [AB09].

Fonctions d'éclatement et de fusion

Soit \mathbf{T} un décalage d'arbres sur l'alphabet A . Pour toute lettre $a \in A$, on considère une partition \mathcal{P}_a des blocs du langage d'ordre 2, $\mathcal{L}_2(\mathbf{T})$, du décalage \mathbf{T} qui sont étiquetés par la lettre a à la racine. On notera par $[(a, b, c)]_a$ l'élément de la partition \mathcal{P}_a qui contient le bloc (a, b, c) ; il s'agit donc d'une classe d'équivalence.

Soit $\Phi : \mathbf{T} \rightarrow \mathcal{P}(\{0, 1\}^*)$ la fonction 2-bloc définie par $\phi(a, b, c) = [(a, b, c)]_a$.

On note par $\widetilde{\mathbf{T}}$ le décalage d'arbres $\Phi(\mathbf{T})$. La fonction Φ est une 2-conjugaison dont l'inverse est une fonction 1-bloc. On dit que Φ est une *fonction d'éclatement entrant* et que $\widetilde{\mathbf{T}}$, et plus généralement tout décalage d'arbres obtenu en renommant les lettres de $\widetilde{\mathbf{T}}$, est un *éclatement entrant* de \mathbf{T} . On dit aussi que Φ^{-1} est une *fonction de fusion entrante* et que \mathbf{T} est une *fusion entrante* de $\widetilde{\mathbf{T}}$. Quand \mathcal{P}_a est la partition triviale pour toute lettre a (i.e. $[(a, b, c)]_a = (a, b, c)$ pour toute lettre a), Φ est appelée *fonction d'éclatement entrant complète* et $\widetilde{\mathbf{T}}$ est l'*éclatement entrant complet* de \mathbf{T} .

Exemple 4.1.5. Soit l'alphabet à deux éléments $A = \{\bullet, \circ\}$, et on choisit comme partitions \mathcal{P}_\bullet et \mathcal{P}_\circ la partition triviale. On définit ainsi Φ la fonction d'éclatement entrant complète sur l'alphabet A .

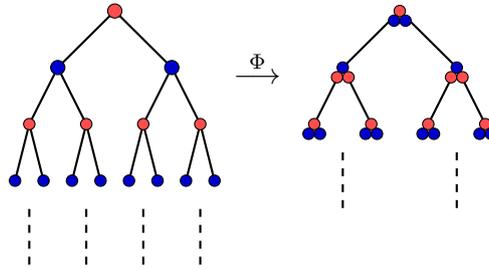


FIGURE 29: Exemple de fonction d'éclatement entrant complète Φ sur l'alphabet $A = \{\bullet, \circ\}$.

Énoncé et preuve du théorème

Théorème 4.1.10 ([AB09]). *Toute conjugaison entre deux décalages d'arbres peut être décomposée en une suite finie de fonctions d'éclatement entrants et de fonctions de fusions entrantes.*

Lemme 4.1.11. *Soit $\Phi : \mathbf{T} \rightarrow \mathbf{T}'$ une m -conjugaison entre deux décalages d'arbres \mathbf{T} et \mathbf{T}' , avec $m \geq 2$. Alors il existe une fonction d'éclatement entrant Ψ de \mathbf{T} vers $\widetilde{\mathbf{T}}$ et une $(m-1)$ -conjugaison $\widetilde{\Phi}$ de $\widetilde{\mathbf{T}}$ vers \mathbf{T}' tels que $\Phi = \widetilde{\Phi} \circ \Psi$.*

$$\begin{array}{ccc}
\mathbf{T} & \xrightarrow{\Phi} & \mathbf{T}' \\
\downarrow \Psi & & \nearrow \tilde{\Psi} \\
\tilde{\mathbf{T}} & &
\end{array}$$

Démonstration. On appelle A l'alphabet du décalage d'arbres \mathbf{T} et A' l'alphabet du décalage d'arbres \mathbf{T}' . On appelle $\varphi : \mathcal{L}_m(\mathbf{T}) \rightarrow A'$ la fonction locale qui définit la m -conjugaison Φ . Soit $\tilde{A} = \mathcal{L}_2(\mathbf{T})$ l'alphabet du décalage d'arbre \tilde{X} , et soit $\Psi_1 : \mathbf{T} \rightarrow \tilde{\mathbf{T}}$ la 2-conjugaison définie par $\psi_1(a, b, c) = (a, b, c)$. La fonction Ψ_1 est donc une fonction d'éclatement entrant complète.

Soit f (resp. g, h) la fonction de $\mathcal{L}_2(\mathbf{T})$ dans A qui envoie (a, b, c) sur a (resp. b, c). On définit $\tilde{\Phi} : \tilde{\mathbf{T}} \rightarrow \mathbf{T}'$ comme la $(m-1)$ -fonction de bloc telle que, pour tout bloc b de $\mathcal{L}_{m-1}(\tilde{X})$, par $\tilde{\varphi}(b) = \varphi(b')$, où b' est le bloc de hauteur m tel que $b'_x = f(b_x)$ pour tout $x \in \{0, 1\}^{<m-1}$, tel que $b'_{x_0} = g(b_x)$, et $b'_{x_1} = h(b_x)$ pour tout $x \in \{0, 1\}^{m-1}$. Ainsi on a $\Phi = \tilde{\Phi} \circ \Psi_1$. \square

Lemme 4.1.12. *Soit $\Phi : \mathbf{T} \rightarrow \mathbf{T}'$ une 1-conjugaison entre deux décalages d'arbres, telle que Φ^{-1} est une fonction m -bloc avec $m \geq 2$. Alors il existe une fonction d'éclatement entrant Ψ_1 de \mathbf{T} vers $\tilde{\mathbf{T}}$, une fonction d'éclatement entrant Ψ_2 de \mathbf{T}' vers $\tilde{\mathbf{T}}'$, et une 1-conjugaison $\tilde{\Phi}$ de $\tilde{\mathbf{T}}$ vers $\tilde{\mathbf{T}}'$ telle que $\Phi = \Psi_2^{-1} \circ \tilde{\Phi} \circ \Psi_1$ et $\tilde{\Phi}^{-1}$ est une $(m-1)$ -fonction de bloc. Le diagramme suivant est donc commutatif.*

$$\begin{array}{ccc}
\mathbf{T} & \xrightarrow{\Phi} & \mathbf{T}' \\
\downarrow \Psi_1 & & \downarrow \Psi_2 \\
\tilde{\mathbf{T}} & \xrightarrow{\tilde{\Phi}} & \tilde{\mathbf{T}}'
\end{array}$$

Démonstration. Supposons que le décalage d'arbres \mathbf{T} (resp. \mathbf{T}') est défini sur l'alphabet A (resp. A'). Soit \mathcal{P}_a la partition des blocs de hauteurs 2 étiquetés par une lettre a à la racine, tels que deux blocs (a, b, c) et (a, b', c') appartiennent à la même classe de la partition si et seulement si $\phi(b) = \phi(b')$ et $\phi(c) = \phi(c')$, où $\phi : A \rightarrow A'$ est la fonction locale qui définit Φ . Soit $\Psi_1 : \mathbf{T} \rightarrow \tilde{\mathbf{T}}$ la fonction de bloc définie localement par $\psi_1(a, b, c) = [(a, b, c)]_a$. Alors $\tilde{\mathbf{T}}$ est un éclatement entrant de \mathbf{T} . Soit $\tilde{\mathbf{T}}'$ l'éclatement entrant complet de \mathbf{T}' sur

l'alphabet $A'_2 = \mathcal{L}_2(\mathbf{T}')$. On note Ψ_2 la fonction d'éclatement entrant complète de \mathbf{T}' vers $\widetilde{\mathbf{T}}$.

On définit une 1-fonction de bloc $\widetilde{\Phi}$ de $\widetilde{\mathbf{T}}$ vers $\widetilde{\mathbf{T}'}$, définie par la fonction locale $\widetilde{\phi}[(a, b, c)]_a = (\phi(a), \phi(b), \phi(c))$. Cette fonction locale est bien définie, de par le choix de la partition \mathcal{P}_a ; ainsi $\Phi = \Psi_2^{-1} \circ \widetilde{\Phi} \circ \Psi_1$. Il reste à vérifier que $\widetilde{\Phi}^{-1} = \Psi_2^{-1} \circ \Phi^{-1} \circ \Psi_1$ est une $(m-1)$ -fonction de bloc, autrement dit que pour tout arbre $t \in \widetilde{\mathbf{T}'}$, le bloc fini b de hauteur $m-1$ enraciné en ε dans l'arbre t détermine complètement $\widetilde{\Phi}^{-1}(t)_\varepsilon$. C'est effectivement le cas, car le bloc de hauteur $m-1$ enraciné en ε dans l'arbre t détermine tous les $\Psi_2^{-1}(t)_{x0}$ et tous les $\Psi_2^{-1}(t)_{x1}$, pour toute position $x \in \{0, 1\}^{m-1}$, et il détermine donc le bloc de hauteur m enraciné en ε de l'arbre $\Psi_2^{-1}(t)$. Ainsi si $t' = (\Phi^{-1} \circ \Psi_2^{-1})(t)$, la lettre t'_ε est entièrement déterminée par le bloc b . De plus le bloc de hauteur $m-1$ enraciné en ε dans l'arbre t détermine aussi $\Psi_1(t'_\varepsilon)$. \square

Nous avons à présent tous les éléments pour démontrer le Théorème 4.1.10.

Démonstration du Théorème 4.1.10. Soit $\Phi : \mathbf{T} \rightarrow \mathbf{T}'$ une n -conjugaison entre deux décalages d'arbres, telle que Φ^{-1} est une m -fonction de bloc, avec $n, m \geq 1$. Par le Lemme 4.1.11 et le Lemme 4.1.12, il existe des fonctions d'éclatements entrants $\Psi_1, \dots, \Psi_{n+m-2}, \Delta_1, \dots, \Delta_{m-1}$, et un renommage de l'alphabet Δ tels que $\Phi = \Delta_1^{-1} \circ \Delta_2^{-1} \cdots \circ \Delta_{m-1}^{-1} \circ \Delta^{-1} \circ \Psi_{n+m-2} \circ \Psi_2 \circ \Psi_1$. Un renommage étant un cas particulier d'éclatement entrant, le théorème est démontré. \square

4.2 Décalages de type fini

4.2.1 Décalages de sommets et de transitions

Dans cette partie nous présentons un cas particulier de décalages d'arbre de type fini : les décalages de sommets (*vertex shift* dans [LM95]).

Soit \mathbf{T} un décalage d'arbres défini sur un alphabet A . On appelle *décalage des blocs d'ordre n* du décalage \mathbf{T} le décalage de type fini $\mathbf{T}^{(n)}$ défini sur l'alphabet $A^{\mathbb{S}_n}$, et qui est l'image de \mathbf{T} par la n -fonction de bloc Φ_n définie localement par $\phi(p) = p$ pour tout motif $p \in \mathcal{L}_n(\mathbf{T})$. La fonction Φ_n est en fait la composée de $n-1$ fonctions d'éclatements entrants complets successifs, et donc tout décalage \mathbf{T} est conjugué à ses décalages des blocs d'ordre supérieur $\mathbf{T}^{(n)}$.

Un *décalage de sommets* est un décalage d'arbres accepté par un automate d'arbres $\mathcal{A} = (V, V, \Delta)$, où les transitions sont de la forme $(q_0, q_1, q) \rightarrow q$. Un arbre est accepté par un tel automate si chaque nœud est étiqueté par l'état correspondant dans le calcul de l'automate. L'ensemble des calculs d'un automate d'arbres est un décalage de sommets. Afin de simplifier les notations,

nous dirons qu'un décalage de sommets est l'ensemble des calculs d'un automate sans étiquette $B = (V, \Gamma)$ dont les transitions sont des éléments de $V^2 \times V$, et sont notées par $(q_0, q_1) \rightarrow q$. Cela revient à dire qu'un décalage de sommets est accepté par un automate d'arbres sans étiquette. En particulier un décalage d'arbres de sommets est un décalage d'arbres de type fini, car un automate d'arbres sans étiquette est 1-local.

Remarque. Un décalage irréductible de type fini n'est pas toujours conjugué à l'ensemble des calculs de son automate minimal. Par exemple le décalage trivial sur un alphabet à deux lettres est accepté par un automate à un seul état.

Proposition 4.2.1. *Tout décalage d'arbres de type fini est conjugué à un décalage d'arbres de sommets.*

Démonstration. Soit $\mathbf{T} = \mathbf{T}_{\mathbb{F}}$ un décalage d'arbres de type fini, défini par l'ensemble fini de blocs interdits F tous de hauteur m , avec $m \geq 1$. Soit $\mathcal{A} = (V, A, \delta)$ l'automate d'arbres déterministe m -local dont l'ensemble d'états est $V = \mathcal{L}_m(\mathbf{T})$. Pour $p_0, p_1 \in V$ et $a \in A$, on note $\text{tronc}(b)$ le bloc de hauteur $m - 1$ tel que $\text{tronc}(b)_x = b_x$ pour tout $x \in \{0, 1\}^{\leq m-1}$, où b est le bloc (a, p_0, p_1) . On définit alors $\delta(p_0, p_1, a) = \text{tronc}(b)$ si et seulement si b est un bloc autorisé de \mathbf{T} .

L'automate d'arbre \mathcal{A} accepte \mathbf{T} , et pour tout arbre $t \in \mathbf{T}$, il existe un unique calcul de \mathcal{A} sur t . Soit \mathbf{T}' le décalage de sommets composé de tous les calculs de l'automate \mathcal{A} . On considère $\mathbf{T}^{(m)}$ le décalage des blocs d'ordre m de \mathbf{T} . Alors $\mathbf{T}^{(m)}$ est exactement \mathbf{T}' , et donc \mathbf{T} est conjugué à \mathbf{T}' . \square

Un décalage d'arbres de transitions est un décalage d'arbres accepté par un automate dont toutes les transitions ont des étiquettes différentes. Un décalage d'arbres de transitions est donc un décalage d'arbres de type fini.

Proposition 4.2.2. *Tout décalage d'arbres de type fini est conjugué à un décalage d'arbres de transitions.*

Démonstration. Si \mathbf{T} est un décalage de type fini d'ordre n , alors sa représentation d'ordre n est un décalage de transitions. \square

4.2.2 Conjugaison des décalages de type fini

Dans cette partie nous montrons, grâce au théorème de décomposition, que le problème de la conjugaison des décalages d'arbres de type fini est décidable. Pour cela nous commençons par montrer que deux fusions entrantes commutent.

Proposition 4.2.3. *Supposons que \mathbf{T}_1 est un décalage de sommets, et que \mathbf{T}_2 , \mathbf{T}_3 sont des décalages de sommets obtenus par fusion entrante de \mathbf{T}_1 . Alors il existe un décalage de sommets \mathbf{T}_4 que l'on peut obtenir à la fois comme fusion entrante de \mathbf{T}_2 et de \mathbf{T}_3 .*

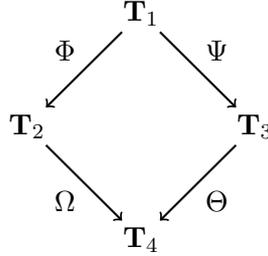


FIGURE 30: Les fusions entrantes commutent. Si les décalages de sommets \mathbf{T}_2 et \mathbf{T}_3 sont des fusions entrantes de \mathbf{T}_1 , il existe un décalage d'arbres de sommets \mathbf{T}_4 qui est une fusion entrante commune à \mathbf{T}_2 et \mathbf{T}_3 .

Dans la Figure 30, les fonctions Φ et Ψ sont des fusions entrantes. Il s'ensuit par la Proposition 4.2.3 que les fonctions Ω et Θ sont aussi des fusions entrantes.

Il est possible de traduire les fonctions d'éclatements entrants et de fusions sortantes sur un décalage de sommets par une transformation sur un automate d'arbres sans étiquette qui reconnaît ce-dernier.

Soit \mathbf{T} un décalage de sommets accepté par un automate d'arbres sans étiquette $\mathcal{A} = (V, \Delta)$, et Φ une fonction d'éclatement entrant de \mathbf{T} vers $\tilde{\mathbf{T}}$ définie par des partitions des ensembles de transitions Δ_r partant de l'état r pour tout sommet $r \in V$. On note par \tilde{V} l'alphabet du décalage $\tilde{\mathbf{T}}$. Il s'agit de l'union des ensembles $\{[(r, p, q)]_r, \mid (p, q) \rightarrow r \in \Delta\}$. Les sommets $[(r, p, q)]_r$ sont appelés *sommets éclatés* du sommet r . On dit que les sommets $[(r, p, q)]_r$ sont *fusionnés* en le sommet r . Le décalage de sommets $\tilde{\mathbf{T}}$ est accepté par l'automate $\tilde{\mathcal{A}} = (\tilde{V}, \tilde{\Delta})$, défini par $([(r, p, q)]_r, [(r', p', q')]_{r'}) \rightarrow [(s, r, r')]_s \in \tilde{\Delta}$ si et seulement si $(r, r') \rightarrow s \in \Delta$.

Pour plus de lisibilité, les sommets de $\tilde{\Delta}$ correspondants à une partition de l'ensemble Δ_r sont notés $r^1 \dots, r^{\ell(r)}$. Une fusion entrante d'un décalage de sommets \mathbf{T} accepté par un automate \mathcal{A} est un décalage de sommets $\tilde{\mathbf{T}}$ accepté par un automate $\tilde{\mathcal{A}}$ tel que $\tilde{\mathbf{T}}$ est un éclatement entrant de \mathbf{T} . Les sommets $r^1 \dots, r^{\ell(r)}$ de l'automate $\tilde{\mathcal{A}}$ sont fusionnés en le sommet r . Remarquons que lorsque $(p', q') \rightarrow r^i \in \tilde{\Delta}$, alors $(p', q') \rightarrow r^j \notin \tilde{\Delta}$, pour tous $p', q' \in \tilde{V}$ et tous $1 \leq i \neq j \leq \ell(r)$. Ceci implique en particulier que $(r^i, p') \rightarrow q' \in \tilde{\Delta}$ si

et seulement si $(r^j, p') \rightarrow q' \in \widetilde{\Delta}$, et $(p', r^i) \rightarrow q' \in \widetilde{\Delta}$ pour toute transition $(p', p^j) \rightarrow q' \in \widetilde{\Delta}$, pour tous sommets $p', q' \in \widetilde{V}$ et pour tous $1 \leq i, j \leq \ell(r)$. De manière informelle, si un sommet r est éclaté en sommets $r^1 \dots, r^{\ell(r)}$, les transitions entrantes sur r sont réparties entre les sommets r^i , alors que les transitions sortantes de r sont dupliquées.

Démonstration. Soit $\mathcal{A}_i = (V_i, \Delta_i)$ un automate d'arbres sans étiquette qui reconnaît le décalage de sommets \mathbf{T}_i pour $i = 1, 2$ et 3 . Supposons tout d'abord qu'il existe une fusion entrante $\Phi : \mathbf{T}_1 \rightarrow \mathbf{T}_2$ et une fusion entrante $\Psi : \mathbf{T}_1 \rightarrow \mathbf{T}_3$. Les états $p^1, \dots, p^{l(p)}$ de V_1 sont fusionnés en un état p de V_2 . Par définition d'une fusion entrante, cela implique que si $(q, r) \rightarrow p^i \in \Delta_1$, alors $(q, r) \rightarrow p^j \notin \Delta_1$ pour tous états $q, r \in V_1$ et tous $1 \leq i \neq j \leq l(p)$. Cela implique aussi que $(p^i, q) \rightarrow r \in \Delta_1$ si et seulement si $(p^j, q) \rightarrow r \in \Delta_1$, et $(q, p^i) \rightarrow r \in \Delta_1$ si et seulement si $(q, p^j) \rightarrow r \in \Delta_1$ pour tous états $q, r \in V_1$ et tous $1 \leq i, j \leq l(p)$. Nous supposons aussi que par ailleurs les états $q^1, \dots, q^{l(q)}$ de V_1 sont fusionnés en un état q de V_3 .

Le cas le plus simple est celui où les états $p^1, \dots, p^{l(p)}$ et $q^1, \dots, q^{l(q)}$ sont tous distincts. Il suffit de définir le décalage \mathbf{T}_4 comme la fusion entrante de \mathbf{T}_2 obtenue en fusionnant les états $p, q^1, \dots, q^{l(q)}$ en un état q . Il s'agit aussi de la fusion entrante de \mathbf{T}_3 obtenue en fusionnant les états $q, p^1, \dots, p^{l(p)}$ en un état q . Supposons à présent que $p^1 = q^1, \dots, p^l = q^l$ pour un entier $1 \leq l \leq \min(l(p), l(q))$. Cela implique que, pour tous $1 \leq i \leq l(p)$, $1 \leq j \leq l(q)$, on a $(p^i, q) \rightarrow r \in \Delta_n$ si et seulement si $(p^j, q) \rightarrow r \in \Delta_n$, et $(q, p^i) \rightarrow r \in \Delta_n$ si et seulement si $(q, p^j) \rightarrow r \in \Delta_n$ pour $n = 1$ et $n = 2$. Nous définissons le décalage \mathbf{T}_4 comme la fusion entrante de \mathbf{T}_2 obtenue en fusionnant les états $p, q^{l+1}, \dots, q^{l(q)}$ en un état p . Il s'agit aussi de la fusion entrante de \mathbf{T}_3 obtenue en fusionnant les états $q, p^{l+1}, \dots, p^{l(p)}$ en un état p . Ainsi, si les fonctions Φ et Ψ sont des fusions entrantes, il en est de même des fonctions Ω et Θ . \square

Grâce à la Proposition 4.2.3, il nous est possible de définir la *fusion minimale* d'un décalage sofique \mathbf{T} , comme le décalage d'arbres de sommets défini par l'automate d'arbres $\mathcal{A} = (V, \Delta)$ avec le plus petit nombre de sommets, cet automate \mathcal{A} étant obtenu par des fusions entrantes successives appliquées à un automate reconnaissant le décalage \mathbf{T} . Cette fusion minimale est bien unique, car si un décalage \mathbf{T} possédait deux fusions minimales \mathbf{T}_1 et \mathbf{T}_2 , la Proposition 4.2.3 nous permet de construire une fusion entrante commune \mathbf{T}_3 , qui par minimalité de \mathbf{T}_1 et \mathbf{T}_2 est telle que $\mathbf{T}_1 = \mathbf{T}_2 = \mathbf{T}_3$.

Théorème 4.2.4. *Soient \mathbf{T}_1 et \mathbf{T}_2 deux décalages d'arbres de type fini. Alors savoir si \mathbf{T}_1 et \mathbf{T}_2 sont conjugués est un problème décidable.*

Démonstration. La Proposition 4.2.1 permet de supposer que les décalages \mathbf{T}_1 et \mathbf{T}_2 sont des décalages de sommets. Par le Théorème 4.1.10, il existe une suite de fonctions d'éclatements entrants et de fonctions de fusions entrantes transformant \mathbf{T}_1 en \mathbf{T}_2 .

Supposons dans un premier temps que cette suite se décompose en une suite de fonctions d'éclatements entrants de \mathbf{T}_1 vers \mathbf{T} , suivie (à un renommage près des sommets de \mathbf{T}) d'une suite de fonctions de fusions entrantes de \mathbf{T} vers \mathbf{T}_2 . Ce cas particulier est illustré dans la Figure 31. La Proposition 4.2.3 nous assure l'existence d'un décalage de sommets à la confluence de deux flèches en pointillé dans la Figure 31. Ainsi les décalages \mathbf{T}_1 et \mathbf{T}_2 ont une fusion commune, et donc la même fusion minimale. Réciproquement si \mathbf{T}_1 et \mathbf{T}_2 ont la même fusion minimale, il existe une suite de fonctions d'éclatements entrants et de fonctions de fusions entrantes de \mathbf{T}_1 vers \mathbf{T}_2 .

Supposons maintenant qu'il existe une suite quelconque d'éclatements entrants et de fusions entrantes de \mathbf{T}_1 vers \mathbf{T}_2 . Cette suite peut se décomposer en plusieurs suites de la forme précédente, et nous arrivons à la même conclusion par transitivité. \square

Le schéma de la preuve nous donne un algorithme pour calculer, si elle existe, une conjugaison entre deux décalages d'arbres de type fini \mathbf{T}_1 et \mathbf{T}_2 . La Proposition 4.2.1 nous permet de nous ramener à des décalages de sommets : $\mathbf{Y}_1 = \psi_1(\mathbf{T}_1)$ et $\mathbf{Y}_2 = \psi_2(\mathbf{T}_2)$, où ψ_i est une suite de fonctions d'éclatements entrants et \mathbf{Y}_i un décalage d'arbres de sommets. Par la Proposition 4.2.3, on se ramène à leur fusion minimale : $\phi_1(\mathbf{Y}_1) = \tilde{\mathbf{Y}}_1$ et $\phi_2(\mathbf{Y}_2) = \tilde{\mathbf{Y}}_2$ où ψ_i est une suite de fonctions de fusions entrantes et $\tilde{\mathbf{Y}}_i$ est une fusion minimale. Il nous suffit alors de comparer les $\tilde{\mathbf{Y}}_i$: s'ils sont différents alors les décalages \mathbf{T}_1 et \mathbf{T}_2 ne sont pas conjugués, et s'ils sont égaux nous avons une conjugaison $\mathbf{T}_1 = \psi_1^{-1} \circ \phi_1^{-1} \circ \phi_2 \circ \psi_2(\mathbf{T}_2)$. Cet algorithme fonctionne en temps exponentiel, car la comparaison des fusions minimales $\tilde{\mathbf{Y}}_1$ et $\tilde{\mathbf{Y}}_2$ revient à un problème d'isomorphisme de graphes.

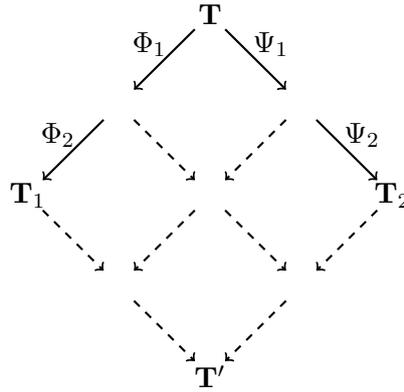


FIGURE 31: Une suite de fonctions d'éclatements entrants de \mathbf{T}_1 vers \mathbf{T} suivie (à un renommage près de l'alphabet de \mathbf{T}), par une suite de fonctions de fusions entrantes de \mathbf{T} vers \mathbf{T}_2 . Chaque flèche représente une fonction de fusion entrante. Les décalages d'arbres \mathbf{T}_1 et \mathbf{T}_2 ont la même fusion minimale \mathbf{T}' .

4.3 Les décalages presque de type fini (AFT)

La notion de décalage d'arbres presque de type fini étend celle connue pour les décalage sur \mathbb{Z} , initialement introduite dans [Mar85]. Pour les décalages sur \mathbb{Z} , elle constitue une classe intéressante notamment en théorie des codes [MK88]. Cette classe de décalages contient la classe des SFT irréductibles, et est strictement incluse dans la classe des décalages sofiqes irréductibles.

4.3.1 Définition

Soient \mathbf{T}, \mathbf{T}' deux décalages d'arbre. Une fonction de bloc $\Phi : X \rightarrow Y$ est *fermante à gauche* s'il existe deux entiers positifs m, a (appelés mémoire et anticipation) tels que, si $\Phi(s) = s'$ et $\Phi(t) = t'$ avec $s'_x = t'_x$ pour tout $x \in \{0, 1\}^i$ avec $0 \leq i \leq (a + m)$, et $s_x = t_x$ pour tout $x \in \{0, 1\}^i$ avec $0 \leq i \leq m - 1$, alors $s_x = t_x$ pour tout $x \in \{0, 1\}^m$.

Un automate d'arbres $\mathcal{A} = (V, A, \Delta)$ est *fermant à gauche* si deux calculs de l'automate \mathcal{A} sur un même arbre qui terminent dans un même état sont les mêmes. Cela revient à dire qu'il existe un entier positif a tel que deux calculs finis C, C' de l'automate \mathcal{A} sur un même bloc $u \in \mathcal{L}_a(\mathbf{T})$ tels que $C_\varepsilon = C'_\varepsilon$ vérifient $C_x = C'_x$ pour $x \in \{0, 1\}$. La notion d'automate d'arbres fermant à

gauche correspond à celle d'automate co-déterministe avec délai fini pour les mots (voir par exemple [Sak09]).

Une fonction de bloc $\Phi : \mathbf{T} \rightarrow \mathbf{T}'$ est *fermante à droite* s'il existe des entiers positifs m, a (appelés mémoire et anticipation) tels que, lorsque $\Phi(s) = s'$, $\Phi(t) = t'$ avec $s'_x = t'_x$ pour tout $x \in \{0, 1\}^i$ avec $0 \leq i \leq (a + m)$, et $s_x = t_x$ pour tout $x \in \{0, 1\}^i$ avec $a + 1 \leq i \leq (a + m)$, alors $s_x = t_x$ pour tout $x \in \{0, 1\}^a$. La fonction Φ est *résoluble à droite* si c'est une fonction 1-bloc fermante à droite avec paramètres $m = 1$ et $a = 0$.

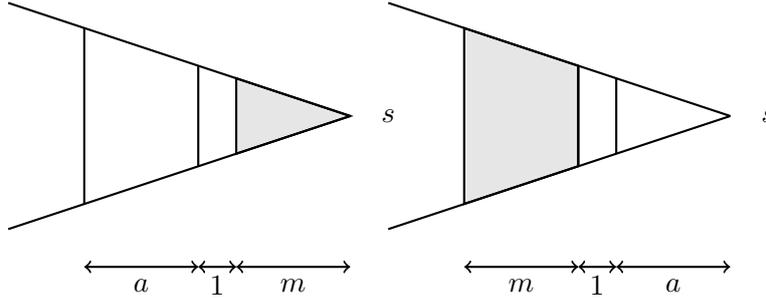


FIGURE 32: Les notions d'automate fermant à gauche et à droite. Les arbres sont ici dessinés horizontalement pour mettre en évidence la provenance des deux notions.

Soit Φ une 1-fonction de bloc de \mathbf{T} dans \mathbf{T}' , définie par une fonction locale ϕ . On dit que le bloc z est *résoluble pour Φ* s'il existe un entier m tel que, si u et v sont deux blocs de hauteur m qui apparaissent dans \mathbf{T} qui vérifient $\phi(u) = \phi(v)$, alors $u_\varepsilon = v_\varepsilon$. Cette notion correspond à celle de *resolving block* définie dans [Mar85].

Définition 16. Un décalage d'arbres sofique est *presque de type fini* (on parlera dans la suite de décalage AFT, de l'anglais *almost of finite type*) si c'est l'image d'un décalage d'arbres irréductible de type fini par une fonction de bloc qui est à la fois résoluble à droite, fermante à gauche, et qui possède un bloc résoluble.

Il est facile de vérifier que la composée d'une conjugaison avec une fonction de bloc fermante à gauche est encore fermante à gauche. De même, une conjugaison préserve la propriété de posséder un bloc résoluble pour une fonction de bloc.

La proposition suivante montre que le caractère AFT d'un décalage d'arbres est un invariant de conjugaison.

Proposition 4.3.1 ([AB10]). *Soient \mathbf{T} et \mathbf{T}' deux décalages d'arbres irréductible conjugués. Alors \mathbf{T} est AFT si et seulement si \mathbf{T}' est AFT.*

Démonstration. Soit \mathbf{T} un décalage d'arbres AFT. Soit Θ une 1-fonction de bloc résoluble à droite, fermante à gauche et qui possède un bloc résoluble, d'un décalage de type fini \mathbf{X} vers \mathbf{T} . Soit Φ une conjugaison entre \mathbf{T} et \mathbf{T}' . On suppose que Φ est une m -conjugaison dont l'inverse est une n -conjugaison. Soit $\mathbf{X}^{(n)}$ (resp. $\mathbf{T}^{(n)}$) le décalage des blocs d'ordre n de \mathbf{X} (resp. \mathbf{T}), et τ (resp. τ') la conjugaison naturelle entre \mathbf{X} et $\mathbf{X}^{(n)}$ (resp. entre \mathbf{T} et $\mathbf{T}^{(n)}$). Le décalage d'arbres $\mathbf{X}^{(n)}$ est de type fini. On peut étendre la fonction de bloc Θ en une 1-fonction de bloc Θ' de $\mathbf{X}^{(n)}$ vers $\mathbf{T}^{(n)}$, de sorte que Θ' est elle aussi une 1-fonction de bloc résoluble à droite, fermante à gauche et avec un bloc résoluble. Soit $\Phi' = \Phi \circ \tau'^{(-1)}$, qui est donc une conjugaison, et $\Psi = \Phi' \circ \Theta'$. La fonction de bloc Ψ est une fonction 1-bloc résoluble à droite. Elle est aussi fermante à gauche et possède aussi un bloc résoluble, donc le décalage d'arbres \mathbf{T}' est AFT. \square

$$\begin{array}{ccccc}
\mathbf{X} & \xrightarrow{\tau} & \mathbf{X}^{(n)} & & \\
\Theta \downarrow & & \Theta' \downarrow & \searrow \Psi & \\
\mathbf{T} & \xrightarrow{\tau'} & \mathbf{T}^{(n)} & \xrightarrow{\Phi'} & \mathbf{T}' \\
& & \underbrace{\hspace{10em}}_{\Phi} & &
\end{array}$$

On dit qu'un automate d'arbres est *presque de type fini* (on dira aussi que l'automate d'arbres est AFT) s'il est déterministe, irréductible, fermant à gauche et synchronisé.

Proposition 4.3.2 ([AB10]). *Un décalage d'arbres est AFT si et seulement si il est accepté par un automate d'arbres AFT.*

Démonstration. Soit \mathbf{T} un décalage d'arbres AFT sur un alphabet A . Soit $\Phi : \mathbf{X} \rightarrow \mathbf{T}$ une fonction de bloc surjective entre un décalage d'arbres de type fini irréductible \mathbf{X} et le décalage \mathbf{T} , qui est donc résoluble à droite, fermante à gauche et qui a un bloc résoluble.

Sans perte de généralité (en remplaçant \mathbf{X} par un de ses décalages des blocs d'ordre supérieur), on peut supposer que \mathbf{X} est un décalage de transitions et que Φ est une fonction 1-bloc. Encore une fois en remplaçant le décalage \mathbf{X} par une de ses décalage des blocs d'ordre supérieur, on peut suppose que Φ est fermante à gauche avec paramètres $a' = 0, m' = 1$.

Soit $\mathcal{A} = (V, E, \Delta)$ un automate d'arbres irréductible et de transitions qui accepte le décalage \mathbf{X} . Soit $\mathcal{A}' = (V, A, \Delta')$ l'automate dont les transitions sont

$(p, q, \phi(e), r) \in \Delta'$ si et seulement si $(p, q, e, r) \in \Delta$. Comme Φ est une fonction 1-bloc résoluble à droite, \mathcal{A}' est un automate déterministe.

On montre que l'automate d'arbres \mathcal{A}' possède un bloc synchronisant. Comme Φ possède un bloc résoluble z , c'est un bloc synchronisant pour \mathcal{A}' . On émonde l'automate \mathcal{A}' en ne gardant que les états accessibles depuis p_z . Comme le décalage \mathbf{T} est irréductible, l'automate \mathcal{A}' reste irréductible et accepte toujours \mathbf{T} .

Montrons à présent que l'automate d'arbres \mathcal{A}' est fermant à gauche. Si ce n'était pas le cas, il existerait deux calculs distincts C et C' de l'automate \mathcal{A}' sur un même arbre t terminant dans un même état p . Soit $(p, q, e, r) \in \Delta$ une transition sortant du couple d'états (p, q) pour des états $p, r \in V$ (s'il n'existe pas de transition de ce type, alors comme l'automate \mathcal{A} est irréductible il existe une transition sortant du couple d'états (q, p) et la suite du raisonnement est la même). On obtient ainsi deux calculs distincts (r, C, D) et (r, C', D) de l'automate \mathcal{A}' sur le même arbre $u = (\phi(e), t, t')$ et qui terminent dans l'état r . Ces deux calculs distincts de \mathcal{A}' sont aussi des calculs distincts de l'automate \mathcal{A} sur deux arbres s, s' du décalage \mathbf{T} donnés par $s_\varepsilon = s'_\varepsilon = e$ et $\Phi(s) = \Phi(s')$. Comme Φ est fermante à gauche avec paramètres $a' = 0, m' = 1$, on en déduit que $s = s'$ et donc $C = C'$.

Réciproquement, on suppose que \mathbf{T} est accepté par un automate d'arbres AFT $\mathcal{A} = (V, A, \Delta)$. Soit \mathbf{X} le décalage de sommets accepté par $\mathcal{A}' = (V, \Delta, \Delta')$, dont les transitions sont de la forme $(p, q, (p, q, a, r)) \rightarrow r$ pour $(p, q, a, r) \in \Delta$. Soit Φ la 1-fonction de bloc de \mathbf{X} sur \mathbf{T} définie par $\phi(p, q, a, r) = a$. Cette fonction de bloc est résoluble à droite puisque l'automate \mathcal{A} est déterministe. Elle est fermante à gauche car \mathcal{A} l'est. Enfin, comme l'automate \mathcal{A} est synchronisé, la fonction Φ possède un bloc résoluble. Ceci montre que le décalage \mathbf{T} est AFT. \square

4.3.2 Caractérisation des AFT par leur couverture de Shannon

Corollaire 4.3.3 ([AB10]). *Un décalage d'arbres sofique irréductible est AFT si et seulement si sa couverture de Shannon est AFT.*

Démonstration. Soit \mathbf{T} un décalage d'arbres irréductible et sofique. Par la Proposition 4.1.8, tout décalage sofique irréductible est accepté par un automate d'arbres irréductible, déterministe et synchronisé \mathcal{S} , qui est en fait la couverture de Shannon du décalage.

Supposons que le décalage \mathbf{T} est AFT. Par la Proposition 4.3.2, on sait que \mathcal{S} est égal à la minimisation d'un automate AFT \mathcal{A} . Montrons que \mathcal{S} est fermant à gauche. Si ce n'est pas le cas, il existe un arbre $t \in \mathbf{T}$ et deux calculs distincts

de \mathcal{S} sur t terminant dans le même état. En conséquence, il existe deux calculs différents de \mathcal{A} sur l'arbre t qui terminent dans deux états p, p' ayant le même contexte.

Soit z un bloc synchronisant pour l'automate \mathcal{A} , qui synchronise sur l'état q_z . Comme \mathcal{A} est irréductible, il existe un hyper-chemin de p vers q étiqueté par un motif w , et tel que z est enraciné en ε . Soit P le code préfixe fini complet qui définit cet hyper-chemin. Alors il existe un calcul C de l'automate \mathcal{A} sur un arbre s tel que, pour tout nœud $x \in P$, l'arbre t est enraciné en x dans s , $C_x = p$ et le motif z est enraciné en ε dans s . Comme p et p' ont même contexte, il existe aussi un calcul C' de l'automate \mathcal{A} sur l'arbre s tel que, pour un certain nœud $x \in P$, on a $C'_x = p'$. Comme le bloc z est synchronisant, on en déduit que $C_\varepsilon = C'_\varepsilon$. On obtient ainsi deux calculs distincts de l'automate \mathcal{A} sur un même arbre s terminant tous deux dans l'état q , ce qui contredit le caractère fermant à gauche de \mathcal{A} . Ceci montre que la couverture de Shannon d'un décalage AFT est fermante à gauche.

Réciproquement supposons que \mathcal{S} est AFT. Alors par la Proposition 4.3.2 le décalage \mathbf{T} est aussi AFT. \square

4.4 Procédures de décision

4.4.1 Calcul de la couverture de Shannon

Soit $\mathcal{A} = (V, A, \delta)$ un automate d'arbres. On rappelle que $\mathcal{D}(\mathcal{A})$ désigne son automate déterminisé.

Proposition 4.4.1. *Il est possible de vérifier en temps polynomial si un automate d'arbres est irréductible.*

Démonstration. Soit $\mathcal{A} = (V, A, \delta)$ un automate d'arbres déterministe. Pour tout état $p \in V$, on définit l'ensemble des P_i , pour $i \geq 0$, par induction de la manière suivante :

- $P_0 = \{p\}$;
- $P_1 = \{q \in V \mid \exists a \in A \text{ tel que } \delta(p, p, a) = q\}$;
- pour tout entier n strictement positif,

$$P_{n+1} = \{q \in V \mid \exists a \in A, p_1, p_2 \in P_n \cup \{p\} \text{ tels que } \delta(p_1, p_2, a) = q\} \cup P_n.$$

La suite d'ensembles de sommets $(P_n)_{n \in \mathbb{N}}$ est par définition une suite croissante, et bornée par l'ensemble fini V . Il existe donc un entier n_0 tel que pour tout

$n \geq n_0$, on a $P_n = P_{n_0}$. Par construction, il existe un hyper-chemin de p à q si et seulement si $q \in P_{n_0}$, et donc l'automate \mathcal{A} est irréductible si et seulement si pour tout état p , l'ensemble P_{n_0} défini précédemment coïncide avec l'ensemble des états V . Cet algorithme est polynomial en le nombre d'états $|V|$ de l'automate car pour chaque état p , le calcul de P_{n_0} s'effectue en au plus $|V|$ étapes, et pour un état p donné le calcul de chaque P_i est polynomial en $|V|$. \square

Proposition 4.4.2. *On peut décider si un automate d'arbres est synchronisé.*

Démonstration. Soit \mathcal{A} un automate d'arbres $\mathcal{D}(A)$ son déterminisé. Alors l'automate \mathcal{A} est synchronisé si et seulement si l'ensemble des états de son déterminisé $\mathcal{D}(A)$ contient un singleton. La complexité en temps et en espace de cet algorithme est exponentiel en le nombre d'états $|V|$ de l'automate \mathcal{A} . \square

Proposition 4.4.3. *Soit $\mathcal{A} = (V, A, \delta)$ un automate d'arbres déterministe qui accepte un décalage sofique irréductible \mathbf{T} . Alors la couverture de Shannon de \mathbf{T} est calculable à partir de \mathcal{A} .*

Démonstration. Soit $\mathcal{D}(\mathcal{A}) = (\mathfrak{P}(V), A, \delta')$ et R un état minimal de $\mathcal{D}(\mathcal{A})$ au sens de l'inclusion. Soit u l'étiquette d'un hyper-chemin entre V et R . Alors u est un motif synchronisant pour $\mathcal{D}(\mathcal{A})$. En effet par minimalité de R , tout calcul fini de $\mathcal{D}(\mathcal{A})$ sur u termine dans l'état R . Comme \mathbf{T} est irréductible, en ne conservant dans $\mathcal{D}(\mathcal{A})$ que les états accessibles depuis R on obtient un automate d'arbres déterministe, irréductible et synchronisé qui accepte le décalage \mathbf{T} . De la Proposition 4.1.8 on déduit que cette opération de réduction nous donne la couverture de Shannon de \mathbf{T} . \square

4.4.2 Automate des paires et graphe des paires

Sur des mots finis, l'automate des paires est utilisé pour vérifier des propriétés sur les paires de chemins de l'automate (voir [Sak09, p. 647]). Nous étendons cette notion d'automate des paires, ainsi que celle de graphes des paires, aux arbres afin de vérifier certaines propriétés des automates d'arbre.

Étant donné un automate d'arbres $\mathcal{A} = (V, A, \Delta)$, on définit *l'automate des paires* de \mathcal{A} , que l'on notera $\mathcal{A} \times \mathcal{A} = ((V^2 \times V^2) \cup V^2, A, \Delta')$, comme l'automate déterministe dont les transitions sont $((p, p'), (q, q')), a \rightarrow (r, r')$ si et seulement si $(p, q), a \rightarrow r$ et $(p', q'), a \rightarrow r'$ sont des transitions de \mathcal{A} . Un *état diagonal* de l'automate des paires $\mathcal{A} \times \mathcal{A}$ est un état (p, p) pour $p \in V$, ou un état $((p, p), (q, q))$ pour $p, q \in V$.

Une représentation de l'automate des paires est donnée par le graphe des paires. Soit $\mathcal{A} = (V, A, \Delta)$ un automate d'arbres. Le *graphe des paires* $G_{\mathcal{A}} =$

(V_G, E_G) de l'automate \mathcal{A} , où $V_G \subseteq (V^2 \times V^2) \cup V^2$ est l'ensemble des sommets et $E_G \subseteq V_G \times \{0, 1\} \times A \times V_G$ est l'ensemble des arêtes. L'ensemble E_G est défini de la manière suivante (pour plus de lisibilité, une arête étiquetée par 1 sera représentée par un trait plein \longrightarrow , et est appelée arête pleine, tandis qu'une arête étiquetée par 0 est représentée par un trait en pointillé \dashrightarrow , et est appelée arête pointillée). Pour toute paire de transitions étiquetées par la même lettre $(p, q), a \rightarrow r$ et $(p', q'), a \rightarrow r'$ et pour toute paire (s, s') d'états de l'automate \mathcal{A} , l'ensemble d'arêtes E_G contient les arêtes suivantes :

$$\begin{aligned} ((p, p'), (q, q')) &\dashrightarrow^a ((r, r'), (s, s')), \\ ((p, p'), (q, q')) &\xrightarrow{a} ((s, s'), (r, r')), \\ ((p, p'), (q, q')) &\dashrightarrow^a (r, r'), \\ ((p, p'), (q, q')) &\xrightarrow{a} (r, r'). \end{aligned}$$

Les étiquettes par des lettres de l'alphabet A portées par les arêtes de G peuvent être retirées afin de réduire la complexité du graphe.

Un sommet du graphe $G_{\mathcal{A}}$ est *utile* s'il possède au moins une arête pleine entrante et une arête pointillée entrante. On ne conserve que la *partie essentielle* du graphe des paires, que l'on obtient en supprimant les sommets qui ne sont pas utiles, ainsi que les arêtes arrivant sur ces sommets ou partant de ces sommets.

Proposition 4.4.4. *On peut calculer la partie essentielle du graphe des paires d'un automate d'arbres \mathcal{A} en temps linéaire en $\mathcal{O}(|E_G| + |V_G|)$.*

Démonstration. On suppose que l'automate d'arbres $\mathcal{A} = (A, V, \delta)$ est déterministe. Le cardinal de $|V_G|$ est alors un $\mathcal{O}(|V|^4)$ et le cardinal de $|E_G|$ est un $\mathcal{O}(|V|^6)$. On appelle *0-prédécesseur* un prédécesseur d'un sommet par une flèche pointillée (d'étiquette 0), et *1-prédécesseur* un prédécesseur d'un sommet par une flèche pleine (d'étiquette 1). On définit de façon similaire les notions de 0-successeur et de 1-successeur. On construit une file d'attente *nodeQueue* de sommets qui doivent être retirés du graphe des paires. Dès qu'un état est retiré de la file d'attente, on retire les arêtes sortant de cet état.

PARTIE ESSENTIELLE (graphe des paires $G_{\mathcal{A}}$)

```

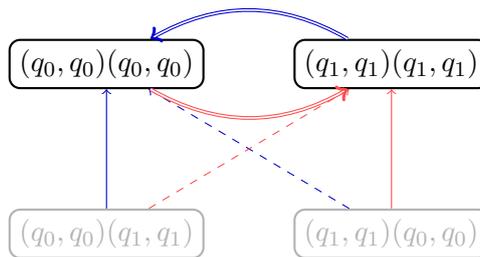
1  Soit  $n_0(u)$  le nombre de 0-prédécesseurs du sommets  $u \in V_G$ .
2  Soit  $n_1(u)$  le nombre de 1-prédécesseurs du sommets  $u \in V_G$ .
3   $vertexQueue \leftarrow$  liste des sommets  $u$  tels que  $n_0(u) = 0$  ou  $n_1(u) = 0$ .
4  Marquer les sommets contenus dans  $vertexQueue$ 
5  while  $vertexQueue$  est non-vide
6      do retirer le sommet  $u$  de  $vertexQueue$ 
7      for tout 0-successeur  $v$  de  $u$ 
8          do diminuer  $n_0(v)$ 
9          if  $n_0(v) = 0$  et  $v$  n'est pas marqué
10             then ajouter  $v$  à  $vertexQueue$ 
11                 marquer  $v$ 
12      for tout 1-successeur  $v$  de  $u$ 
13          do diminuer  $n_1(v)$ 
14          if  $n_1(v) = 0$  et  $v$  n'est pas marqué
15             then ajouter  $v$  à  $vertexQueue$ 
16                 marquer  $v$ 
17  return les états non marqués

```

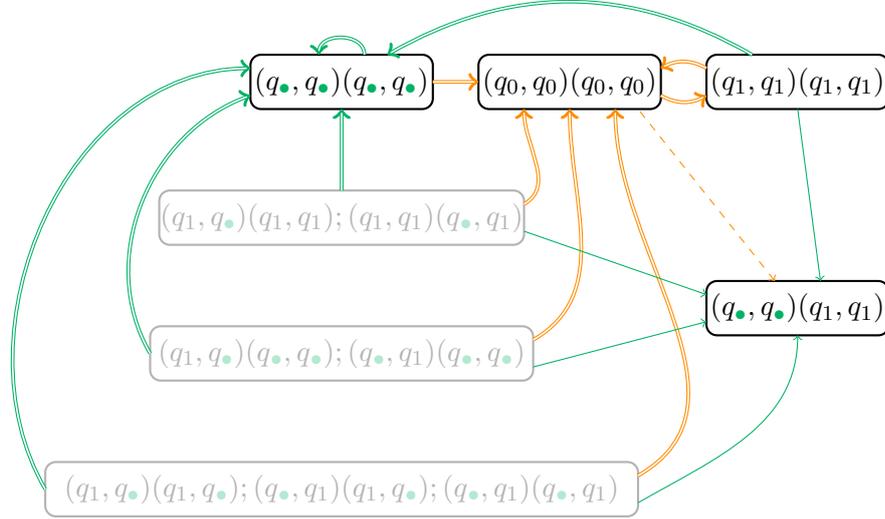
□

Un sommet $((p, q), (r, s))$ (resp. (p, q)) du graphe des paires G est *non diagonal* si $p \neq q$ ou bien $r \neq s$ (resp. $p \neq q$). Dans la suite on parlera du graphe des paires pour désigner sa partie essentielle.

Exemple 4.4.1. Le graphe des paires de l'automate d'arbres de l'Exemple 4.1.1 est représenté ci-dessous. Les sommets utiles sont cerclés de noirs, les autres de gris. Les couleurs des flèches correspondent à la lettre qu'elle porte comme étiquette. Une double flèche remplace une flèche pleine et une flèche pointillée portant la même étiquette. Pour plus de lisibilité, le sommet (p, q) est confondu avec le sommet $((p, q), (p, q))$.



Exemple 4.4.2. Le graphe des paires de l'automate d'arbres de l'Exemple 4.1.2 est représenté ci-dessous, avec les mêmes conventions que pour l'exemple précédent.



La proposition suivante, qui découle directement de la définition du graphe des paires, énonce un résultat qui sera utilisé pour montrer que le graphe des paires permet de décider certaines propriétés sur les décalages sofiques, comme la localité (voir la Proposition 4.4.7) ou la fermeture à gauche (voir la Proposition 4.4.9).

Proposition 4.4.5. *Soit $\mathcal{A} = (V, A, \Delta)$ un automate d'arbres. Un sommet $((p, p'), (q, q'))$ est un sommet du graphe des paires $G_{\mathcal{A}}$ (en ne conservant que sa partie essentielle) si et seulement s'il existe un arbre s avec deux calculs de \mathcal{A} sur s , l'un terminant sur p et l'autre sur p' , ainsi qu'un arbre t avec deux calculs de \mathcal{A} sur t , l'un terminant sur q et l'autre sur q' .*

Démonstration. Ceci provient du fait qu'on n'a conservé que la partie essentielle du graphe des paires, et que donc les sommets non utiles ont été supprimés. De plus, s'il existe une arête $((r, r'), (s, s')) \xrightarrow{0,a} ((p, p'), (q, q'))$ (ou une arête $(r, r') \xrightarrow{0,a} ((p, p'), (q, q'))$) dans le graphe des paires $G_{\mathcal{A}}$, alors il existe deux calculs E et E' de l'automate \mathcal{A} tels que $\sigma^0(E) = C$, $\sigma^0(E') = C'$, $\sigma^1(E) = D$, $\sigma^1(E') = D'$ (σ étant l'action naturelle par translation de $\{0, 1\}^*$ sur les arbres), le calcul E termine dans l'état r et le calcul E' termine dans l'état r' . \square

Remarque. Il existe une arête $((r, r'), (s, s')) \xrightarrow{0,a} ((p, p'), (q, q'))$ (ou une arête $(r, r') \xrightarrow{0,a} ((p, p'), (q, q'))$) dans le graphe des paires $G_{\mathcal{A}}$ si et seulement s'il existe une lettre $a \in A$ et deux transitions $(p, q) \xrightarrow{a} r$ et $(p', q') \xrightarrow{a} r'$ (ou $(p, q) \xrightarrow{a} s$ et $(p', q') \xrightarrow{a} s'$) dans l'automate \mathcal{A}

4.4.3 Localité

Proposition 4.4.6. *Un automate d'arbres est local si et seulement s'il existe un calcul dans son automate des paires qui termine sur un état non diagonal.*

Démonstration. Par définition de l'automate des paires $\mathcal{A} \times \mathcal{A}$, l'existence dans $\mathcal{A} \times \mathcal{A}$ d'un calcul se terminant sur un état (p, q) avec $p \neq q$ implique l'existence dans \mathcal{A} de deux calculs distincts sur un même arbre. Réciproquement, s'il existe deux calculs distincts de \mathcal{A} sur un même arbre t , ces deux calculs diffèrent en un certain nœud $x \in \{0, 1\}^*$. D'où l'existence de deux calculs sur le sous-arbre de t en position x qui se terminent dans deux états distincts. \square

Proposition 4.4.7. *Un automate d'arbres \mathcal{A} n'est pas local si et seulement si la partie essentielle de son graphe des paires contient un état non diagonal (p, q) avec $p \neq q$.*

Démonstration. Soit $G_{\mathcal{A}}$ le graphe des paires d'un automate d'arbres \mathcal{A} . Si \mathcal{A} n'est pas local, alors il existe deux calculs de l'automate \mathcal{A} sur un même arbre qui terminent dans les états r et r' respectivement, avec $r \neq r'$. Par la Proposition 4.4.5, c'est équivalent au fait que le sommet (r, r') soit dans la partie essentielle du graphe des paires.

Réciproquement, si $G_{\mathcal{A}}$ contient un sommet non diagonal $((r, r'), (s, s'))$, avec $r \neq r'$ et $s \neq s'$ (ou (r, r') avec $r \neq r'$), alors il existe une lettre $a \in A$ et une arête $((p, p'), (q, q')) \xrightarrow{0,a} ((r, r'), (s, s'))$ ou une arête $((p, p'), (q, q')) \xrightarrow{0,a} (r, r')$. Par la Proposition 4.4.5 et la remarque qui la suit, il existe deux calculs de \mathcal{A} sur un même arbre t , l'un terminant sur l'état r et l'autre terminant sur l'état r' . \square

Pour un automate d'arbres déterministe $\mathcal{A} = (A, V, \delta)$, le graphe des paires $G_{\mathcal{A}}$ possède $\mathcal{O}(|V|^4)$ sommets et $\mathcal{O}(|V|^6)$ arêtes. Il est donc possible de vérifier en temps $\mathcal{O}(|V|^6)$ si un automate d'arbres est local.

Exemple 4.4.3. L'automate d'arbres \mathcal{A}_1 de l'Exemple 4.1.1 est local car la partie essentielle de son graphe des paires ne contient que des états diagonaux. L'automate d'arbres \mathcal{A}_1 de l'Exemple 4.1.2 n'est pas local car la partie essentielle de son graphe des paires contient l'état non diagonal $(q_{\bullet}, q_{\bullet})(q_1, q_1)$.

4.4.4 Automate fermant à gauche

Proposition 4.4.8. *Un automate d'arbres n'est pas fermant à gauche si et seulement s'il existe un calcul de son automate des paires terminant dans un état diagonal et passant par un état non diagonal.*

Démonstration. Soit \mathcal{A} un automate d'arbres déterministe. Par définition de l'automate des paires, l'existence d'un calcul dans $\mathcal{A} \times \mathcal{A}$ terminant dans un état diagonal (p, p) et passant par un état (r, s) avec $r \neq s$ est équivalent à l'existence de deux calculs distincts de l'automate \mathcal{A} sur un même arbre et qui terminent sur un même sommet. \square

Proposition 4.4.9. *Un automate d'arbres n'est pas fermant à gauche si et seulement s'il existe un chemin dans son graphe des paires (dont on n'a conservé que la partie essentielle) partant d'un sommet non diagonal et arrivant sur un sommet (p, p) .*

Démonstration. Soit $G_{\mathcal{A}}$ le graphe des paires d'un automate d'arbres \mathcal{A} . Si \mathcal{A} n'est pas fermant à gauche, alors il existe deux calculs distincts C et C' de l'automate sur un même arbre t terminant dans un même état p . Soit x un nœud de l'arbre t tel que $C_x \neq C'_x$. Alors il existe dans le graphe des paires $G_{\mathcal{A}}$ un chemin étiqueté par (x, w) partant d'un sommet $((C_x, C'_x), (s, s'))$ ou un sommet $((s, s'), (C_x, C'_x))$ (selon la dernière lettre 0 ou 1 du nœud x) et terminant sur un sommet (p, p) , où w est l'étiquette du chemin de la racine de l'arbre t au nœud x .

Réciproquement, supposons qu'il existe un chemin dans le graphe des paires $G_{\mathcal{A}}$ entre un sommet $((p, p'), (q, q'))$, avec $p \neq p'$ ou $q \neq q'$, et un sommet (r, r) . Cela signifie qu'il existe un arbre t et deux calculs C et C' de l'automate \mathcal{A} sur cet arbre t qui terminent sur l'état p , et tels qu'il existe des nœuds $x, y \in \{0, 1\}^*$ avec $C_x = p$, $C'_x = q$ et $C_y = q$, $C'_y = q'$. Cela implique que les deux calculs C et C' sont distincts, et donc l'automate \mathcal{A} n'est pas fermant à gauche. \square

Proposition 4.4.10. *Soit \mathbf{T} un décalage d'arbres irréductible accepté par un automate d'arbre \mathcal{A} . On peut décider si \mathbf{T} est un AFT.*

Démonstration. On commence par calculer la couverture de Shannon \mathcal{S} du décalage d'arbre \mathbf{T} , comme indiqué dans la Proposition 4.4.3. Il suffit ensuite de vérifier qu'elle est AFT. Cela revient à vérifier qu'elle est fermante à gauche, car par définition la couverture de Shannon d'un décalage d'arbres est au automate d'arbres déterministe, irréductible et synchronisé, ce qui est possible d'après la Proposition 4.4.9. Le nombre de sommets du graphe des paires $G_{\mathcal{S}}$ est un

$\mathcal{O}(|V|^4)$ et son nombre d'arêtes est un $\mathcal{O}(|V|^6)$. La propriété de la Proposition 4.4.9 peut être vérifiée en temps linéaire en la taille du graphe des paires G_S . On peut donc vérifier en temps polynomial si la couverture de Shannon d'un décalage d'arbres sofique irréductible est AFT. \square

Remarque. L'algorithme proposé par Seidl [Sei89] pour vérifier que le degré d'ambiguïté d'un automate d'arbres est fini possède une complexité du même ordre (cube du nombre de transitions de l'automate).